

# PROPIETARISPUNTCAT

Plataforma de comunicació per a comunitats de veïns i administradors de finques

Autor: Jaime Aparicio Barrera

Tutor: Maria José Pedraza Gutiérrez

Professor: Jordi Ustrell Garrigos

Grau en Multimèdia

Enginyeria Web

Juny del 2021

## Crèdits/Copyright



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/)

## FITXA DEL TREBALL FINAL

<b>Títol del treball:</b>	<i>Propietarispuntcat</i>
<b>Nom de l'autor:</b>	<i>Jaime Aparicio Barrera</i>
<b>Nom del col·laborador/a docent:</b>	<i>Jordi Ustrell Garrigos</i>
<b>Nom del PRA:</b>	<i>Ferran Adell Español</i>
<b>Data de lliurament (mm/aaaa):</b>	<i>06/2021</i>
<b>Titulació o programa:</b>	<i>Grau en Multimèdia</i>
<b>Àrea del Treball Final:</b>	<i>Enginyeria web</i>
<b>Idioma del treball:</b>	<i>Català</i>
<b>Paraules clau</b>	<i>Comunitat de veïns, administrador de finques, tauler virtual, informació, propietaris, documentació</i>
<b>Resum del Treball (màxim 250 paraules):</b>	
<p>La idea principal del projecte sorgeix a partir de la experiència personal de comprovar que, en moltes comunitats de veïns, hi ha una dificultat important per a accedir a documentació actualitzada d'interès general de la finca. Aquest problema s'accentua degut al propi funcionament les comunitats. Quan el president es fa responsable i portaveu de la comunitat davant de l'administrador de finques, sempre pot provocar que no hi hagi la suficient agilitat per resoldre un problema comunitari en cas d'urgència, ja que aquest no té accés immediat a la informació general de la comunitat.</p> <p>Si bé sabem que hi ha administradors de finques que fan servir sistemes interns de gestió de comunitats que, en part, ajuden a solucionar part d'aquests problemes, aquesta plataforma no està enfocada per ells, sinó que pretén convertir-se en una plataforma dirigida específicament a les comunitats de veïns. En aquesta plataforma podran trobar tota aquella documentació referida a la seva finca com els serveis contractats, telèfons d'interès i d'urgències, pressupostos de treballs a realitzar, llistat de reunions i actes).</p> <p>La implementació d'aquesta plataforma requereix crear una base de dades MySQL per a guardar tots els contactes, així com per a penjar tota la documentació disponible. D'aquesta manera, a través d'una API REST, la qual fa servir el protocol HTTP<sup>1</sup>, s'implementaran totes les comunicacions de dades que es facin entre el FrontEnd i el Backend. El disseny de la pàgina és responsiu per adaptar-se a qualsevol mida de pantalla.</p>	

<sup>1</sup> HTTP (HyperText Transfer Protocol). Protocol de comunicació web que permet transferència d'informació.

**Abstract (in English, 250 words or less):**

The main idea of this project came from my own experience and after realising how in many neighbourhood communities exist a significant difficulty accessing to updated property's general interest documentation. This problem becomes bigger because the own communities work process. When the president becomes responsible and spokesman for the community in front the estate administrator it can create an agility lack for solving urgent community problem an others, it's generally caused because he hasn't have all the updated community general information.

Although we know that there are estate administrators using his internal community management software which help them to solve some of these problems, this platform it's not focus to them, it's focus to the another specially target, the neighbourhood communities.

In that platform they can find all the documentation related to their property like suppliers, contracts, telephone numbers of interest and emergencies, budgets, list of meetings and events.

The implementation of this platform requires the creation of a MySQL database to store all users and all the available documentation. Requests from the FrontEnd to the BackEnd are handle through an REST API, using HTTP. The web design is responsive to suit any screen size.

## Cita

“Quant més dur treballo, més sort tinc”

Gary Player

# Agraïments

A la meva parella Marga, que ha estat la persona qui més m'ha animat i recolzat per continuar els meus estudis i ha tingut una paciència infinita en els moments de més esforç.

# Notacions i Convencions

Tipus de text	Tipografia	Mida	Format
<b>Títol 1</b>	Arial	20pt	Negreta
<b>Títol 2</b>	Arial	13pt	Negreta
<b>Títol 3</b>	Arial	12pt	Negreta
Contingut	Arial	10pt	Normal
Figures	Arial	9pt	Normal
Codi	Consolas	10,5pt	Normal

# Índex

<b>1. Introducció</b>	<b>13</b>
1.1. Introducció/Prefaci	13
1.2. Descripció/Definició	14
1.3. Objectius generals	15
1.3.1. Objectius principals	15
1.3.2. Objectius secundaris	15
1.4. Metodologia i procés de treball	16
1.5. Planificació	18
1.5.1. Diagrama de Gantt	19
1.6. Pressupost	21
1.7. Estructura de la resta del document	22
<b>2. Anàlisi de mercat</b>	<b>23</b>
2.1. Públic objectiu i perfils d'usuari	23
2.1.1. Anàlisi del públic objectiu	23
2.2. Competència/Antecedents	25
2.2.1. Colindar app	25
2.2.2. Fynkus	27
2.2.3. Conclusions	29
2.3. Anàlisi DAFO	30
<b>3. Proposta</b>	<b>31</b>
3.1. Definició d'objectius/especificacions del producte	31
3.1.1. Tipus de rols d'usuari	31
3.2. Model de negoci	32
<b>4. Disseny</b>	<b>33</b>
4.1. Arquitectura general de l'aplicació	33
4.2. Arquitectura de la informació	34
4.2.1. Base de dades	34
4.2.2. Arbre de Navegació	38
4.3. Llenguatges de programació i APIs utilitzats	40



---

4.3.1. API's: Geocoding.....	40
4.3.2. API REST.....	41
4.3.3. EndPoints.....	42
4.3.4. FrontEnd .....	64
<b>4.4. Disseny gràfic .....</b>	<b>64</b>
4.4.1. Estils .....	64
4.4.2. Usabilitat/UX.....	66
<b>5. Implementació .....</b>	<b>68</b>
5.1. Requisits d'instal·lació .....	68
<b>6. Demostració.....</b>	<b>71</b>
6.1. Prototips .....	71
6.1.1. Prototips Lo-Fi.....	71
6.1.2. Prototips Hi-Fi .....	76
<b>7. Conclusions i línies de futur .....</b>	<b>79</b>
7.1. Conclusions .....	79
7.2. Línies de futur .....	80
<b>Bibliografia.....</b>	<b>81</b>
<b>Annexos.....</b>	<b>82</b>

# Figures i taules

## Índex de figures

Figura 1: Cicle de vida desenvolupament en cascada .....	16
Figura 2: Cicle de vida desenvolupament en cascada del projecte .....	17
Figura 3: PAC1, PAC2. Diagrama de Gantt .....	19
Figura 4: PAC3. Diagrama de Gantt .....	19
Figura 5: PAC4. Diagrama de Gantt .....	19
Figura 6: PAC5. Diagrama de Gantt .....	20
Figura 7: Colindar. Pantalla inici.....	25
Figura 8: Colindar. Creació de Publicacions .....	26
Figura 9: Colindar. Reserves d'instal·lacions .....	26
Figura 10: Fynkus. Com funciona .....	27
Figura 11: Fynkus. Pantalla inicial.....	28
Figura 12: Fynkus. Pantalla inicial.....	28
Figura 13: Anàlisi DAFO .....	30
Figura 14: Arquitectura general de l'aplicació .....	33
Figura 15: Diagrama Base de Dades .....	34
Figura 16: Arbre de navegació. Mapa General .....	38
Figura 17: Arbre de navegació. Inici de sessió .....	39
Figura 18: Arbre de navegació. Comunitat.....	39
Figura 19: Arbre de navegació. Documentació, Reformes, Proveïdors, Juntes i Tauler Anuncis .....	40
Figura 20: NodeJS: Exemple de creació d'un servidor .....	41
Figura 21: Express: Exemple d'ús.....	41
Figura 22: Logotips .....	64
Figura 23: Colors Corporatius .....	65
Figura 24: família Tipogràfica Logotip .....	65
Figura 25: Paleta de colors .....	65
Figura 26: Família Tipogràfica.....	66
Figura 27: Estils: Botons .....	66
Figura 28: Panell control XAMPP.....	68
Figura 29: Posada en funcionament de l'API en local.....	69
Figura 28: Wireframes: Inici Sessió, Comunitat i registre.....	71
Figura 29: Wireframes: Comunitat, Menú i Perfil Usuari .....	72
Figura 30: Wireframes: Telèfons interès i Documentació.....	73
Figura 31: Wireframes: Proveïdors .....	74
Figura 32: Wireframes: Reformes .....	75
Figura 33: Wireframes: Juntes i Tauler d'Anuncis.....	75
Figura 34: Prototips Hi-Fi (1).....	76
Figura 35: Prototips Hi-Fi (2).....	77
Figura 36: Prototips Hi-Fi (3).....	78

Figura 36: Annex C: Arbre de Navegació.....	83
Figura 37: Annex C: Esquema E-R Base de Dades.....	84
Figura 38: Annex C: Creació Taules Base de Dades.....	84
Figura 39: Annex C: Creació Prototips Hi-Fi versió mòbil .....	85
Figura 40: Annex C: Creació Prototips Hi-Fi versió web .....	85

## Índex de taules

Taula 1: Planificació del projecte.....	18
Taula 2: Pressupost .....	21
Taula 3: Comparativa aplicacions .....	29
Taula 4: Base de dades: Taula d'Entitats.....	36
Taula 5: Base de dades: Taula de Relacions.....	37
Taula 6: EndPoint: POST login .....	42
Taula 7: EndPoint: POST user.....	43
Taula 8: EndPoint: GET user/id.....	43
Taula 9: EndPoint: PUT changeRole/id/cif .....	43
Taula 10: EndPoint: PUT user/id.....	44
Taula 11: EndPoint: DELETE user/id .....	44
Taula 12: EndPoint: PUT password .....	45
Taula 13: EndPoint: GET communities/id .....	45
Taula 14: EndPoint: GET joinCommunity/cif .....	45
Taula 15: EndPoint: POST joinCommunity/cif.....	46
Taula 16: EndPoint: PUT joinCommunity .....	46
Taula 17: EndPoint: DELETE joinCommunity/user/cif.....	47
Taula 18: EndPoint: GET pendingAccess/id .....	47
Taula 19: EndPoint: GET requestAccessCommunity/id .....	47
Taula 20: EndPoint: GET community/address .....	48
Taula 21: EndPoint: POST community/id.....	48
Taula 22: EndPoint: PUT community/id .....	49
Taula 23: EndPoint: DELETE community/id.....	49
Taula 24: EndPoint: GET phone/cif.....	50
Taula 25: EndPoint: GET phone/cif/string .....	50
Taula 26: EndPoint: POST phone/cif .....	50
Taula 27: EndPoint: PUT phone/cif.....	51
Taula 28: EndPoint: DELETE phone/id .....	51
Taula 29: EndPoint: GET provider/cif.....	52
Taula 30: EndPoint: GET provider/cif/string .....	52
Taula 31: EndPoint: POST provider/cif .....	53
Taula 32: EndPoint: PUT provider/cif .....	53
Taula 33: EndPoint: DELETE provider/cif .....	53
Taula 34: EndPoint: PUT provider/provider/cif .....	54
Taula 35: EndPoint: GET profession.....	54
Taula 36: EndPoint: POST profession .....	55

---

Taula 37: EndPoint: GET documents/cif/documentType.....	55
Taula 38: EndPoint: GET documents/cif/documentType/string .....	56
Taula 39: EndPoint: POST documents/cif .....	56
Taula 40: EndPoint: PUT documents/cif .....	56
Taula 41: EndPoint: DELETE documents/cif.....	57
Taula 42: EndPoint: GET refurbishment /cif .....	57
Taula 43: EndPoint: GET refurbishment /cif/string .....	58
Taula 44: EndPoint: POST refurbishment /cif .....	58
Taula 45: EndPoint: PUT refurbishment /cif .....	59
Taula 46: EndPoint: DELETE refurbishment /cif .....	59
Taula 47: EndPoint: GET workstate .....	59
Taula 48: EndPoint: GET advertisement/cif .....	60
Taula 49: EndPoint: GET advertisement/cif/string.....	60
Taula 50: EndPoint: POST advertisement .....	61
Taula 51: EndPoint: PUT advertisement/cif .....	61
Taula 52: EndPoint: DELETE advertisement/cif.....	61
Taula 53: EndPoint: GET meeting/cif .....	62
Taula 54: EndPoint: GET meeting/cif/string .....	62
Taula 55: EndPoint: POST Meeting .....	63
Taula 56: EndPoint: PUT meeting/cif .....	63
Taula 57: EndPoint: DELETE meeting/cif .....	64

# 1.Introducció

## 1.1. Introducció/Prefaci

Fa uns pocs anys em vaig trobar en la situació que em va tocar ser president d'escala de la meva comunitat, només un parell de mesos després d'haver entrat a viure al pis. L'accés al càrrec venia acompanyat d'una carpeta molt gran de diferent documentació que gairebé ningú s'havia mirat en moltíssims anys i, per tant, cada any es passava aquesta carpeta de manera automàtica de president a president. Al poc temps, vaig haver d'afrontar una urgència a l'escala i no vaig saber trobar dintre de tota aquesta carpeta una cosa tan bàsica com el segur de la comunitat, amb l'afegit que, al ser cap de setmana, l'administrador de finques no estava localitzable havent de buscar-me la vida per solucionar el problema. Aquest és només un exemple viscut, però no l'únic, de com la falta d'organització i disposició de documentació d'interès en una escala de veïns, pot esdevenir un problema.

Si bé, n'hi ha molts administradors de finques que han aprofitat les noves tecnologies per actualitzar-se i millorar la gestió i la comunicació de les seves comunitats, aquests sistemes són productes destinats més als administradors de finques, els quals poden treballar amb productes diferents i incompatibles, creant un perjudici per als propis veïns.

Així doncs, he volgut aprofitar l'oportunitat que em dona aquest treball i tots els coneixements assolits durant tots aquests anys d'estudis, per crear una plataforma web que estigui dirigida i destinada als propis veïns per tal d'oferir un espai virtual a la pròpia comunitat.

## 1.2. Descripció/Definició

Aquest treball pretén crear una aplicació web que serveixi a una comunitat de veïns per tenir un control més directe i precís sobre tot allò que envolta la gestió de la seva finca, que en cap cas té la pretensió de substituir la feina d'un administrador de finques, sinó més aviat oferir una eina complementaria més enfocada a les necessitats dels veïns.

Actualment existeixen administradors de finques que fan servir serveis de gestió online i que ofereixen solucions completes tant per a ells com per als propietaris ja que poden tenir accés a una zona del portal virtual on poden fer tota mena de consultes referides a l'estat de la seva comunitat. El problema d'aquests productes és que són eines enfocades a la part d'administració de finques i provoca que els veïns no tinguin el control sobre la seva comunitat, fet que pot provocar situacions com que davant d'un eventual canvi d'administrador per part dels veïns, provoqui pèrdua d'informació per possibles incompatibilitats de sistemes de gestió del nou administrador.

La solució proposada consisteix en oferir un entorn virtual on cada veí es podrà registrar per tenir accés a una sèrie de serveis que permetran tenir coneixement de tot allò que envolta el dia a dia de la comunitat. Entre aquests serveis es troba la gestió documental de la finca, serveis contractats, la consulta de telèfons d'interès i d'urgències, gestió de pressupostos i treballs pendents, llistat de pròximes reunions, secció de contacte. Per tant, aquest és un espai propi de la comunitat que pot ser gestionat per la mateixa comunitat, si així es decideix, a través del president de la comunitat (o d'un altre responsable), o delegar la responsabilitat al propi administrador de finques.

Amb aquesta aplicació també es té en compte que és possible que hi hagin propietaris que no disposin de connexió a internet o que no es trobin còmodes davant de l'ús de noves tecnologies i, per tant, aquest sistema no es incompatible amb altres mètodes organitzatius. Un altre aspecte que es vol tenir en compte es que el resultat final ha de tenir un disseny suficientment senzill com per ser usat per un públic objectiu adult amb un rang d'edat ampli, a banda que sigui adaptable a diferents tipus de pantalles.

## 1.3. Objectius generals

### 1.3.1. Objectius principals

Objectius de l'aplicació:

- Recopilar diferent tipus de documentació d'una comunitat de veïns
- Facilitar llistat de telèfons d'interès i professionals en diversos àmbits
- Oferir informació actualitzat sobre l'estat general d'una finca
- Tenir accés directe a pressupostos de treballs a realitzar
- Fer de tauler d'anuncis virtual
- Servir d'eina comunicativa entre veïns i administrador de finques

Objectius per a l'usuari:

- Tenir un accés més directe a tota la informació necessària de la seva comunitat
- Tenir un històric de tot allò que hagi succeït a la comunitat
- Poder trobar solucions a problemes de manera més ràpida i senzilla
- Reduir la dependència amb tercers per resoldre dubtes

Objectius personals de l'autor del TF:

- Poder aplicar i veure el grau de coneixements assolits durant els meus estudis
- Ser capaç de fer un projecte complet, des del seu disseny fins a la seva implementació final
- Fer servir i aprendre tecnologies, com NodeJS<sup>2</sup> i Angular<sup>3</sup>, que no he fet servir al llarg del meu Grau

### 1.3.2. Objectius secundaris

Objectius addicionals que enriqueixen el TF.

- Explorar possibles solucions d'una situació en la qual m'he trobat en la meva experiència personal
- Oferir solucions que redueixen la dependència de les comunitats respecte els administradors de finques
- Disseny adaptat a les diferents persones i als diferents dispositius

---

<sup>2</sup> NodeJS: <https://nodejs.org/es/>

<sup>3</sup> Angular: <https://angular.io>

## 1.4. Metodologia i procés de treball

Aquest és un projecte que s'ha realitzat des de zero i aquest fet ha influït significativament en quina era la millor metodologia de treball, ja que s'ha hagut d'anar pas a pas. En aquest supòsit, es va desestimar des d'un inici aplicar cap metodologia àgil ja que calia completar certes tasques abans d'implementar unes altres, de manera que, una metodologia seqüencial resultava més adient. A més, el fet d'establir una sèrie de fases em va ajudar a tenir un coneixement més exhaustiu de les necessitats del projecte, la qual cosa em va servir per poder establir unes bases sòlides sobre les quals poder avançar, amb el consegüent estalvi de temps al no haver de fer nous replantejaments (encara que sí revisions del treball realitzat en etapes anteriors).

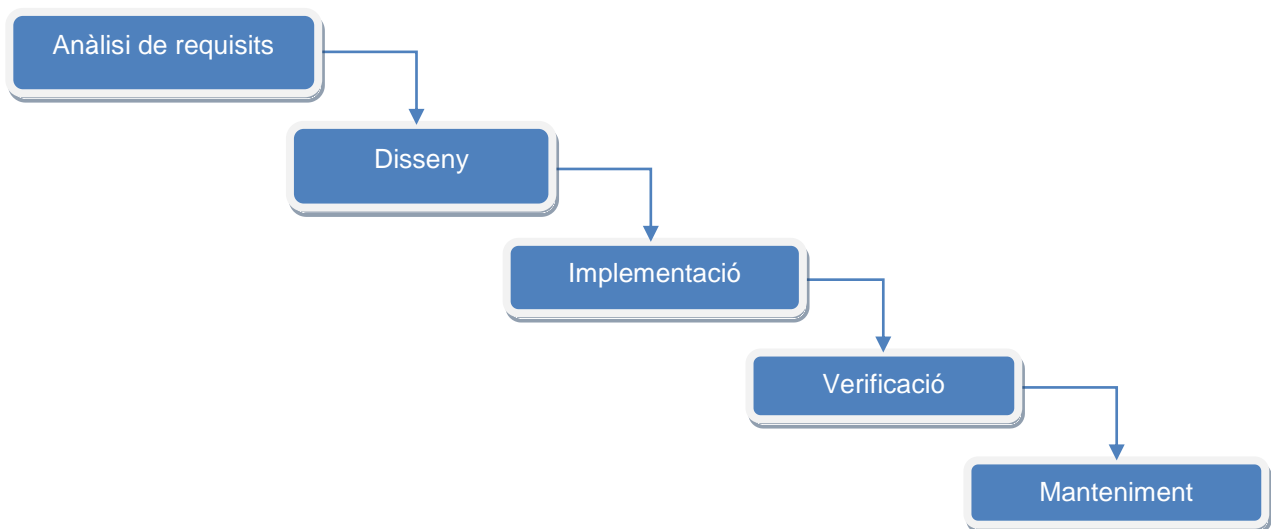


Figura 1: Cicle de vida desenvolupament en cascada

**Anàlisi de requisits:** l'etapa inicial i la més important ja que es va analitzar les necessitats del sistema i va quedar definit l'abast del projecte. En aquesta fase va quedar determinat quines funcionalitats s'havien d'incloure dintre del treball i quines no eren necessàries. També es va realitzar un estudi de mercat per analitzar les opcions existents de la competència i detectar encerts i mancances va ajudar a determinar aquestes funcionalitats.

**Disseny del sistema:** en aquesta fase es va definir com seria el producte final. Un cop recollits tots els requisits del sistema, calia plasmar-los en un disseny clar, entenedor i accessible. Tenir clar tots els conceptes de disseny del projecte facilitava poder centrar-se en el propi codi durant la següent fase d'implementació.



**Implementació:** és la etapa en la qual es va programar tot el codi que donaria lloc al producte final. Aquesta fase constava de crear una base de dades (MySQL), codificar una API REST al BackEnd (NodeJS), que va permetre atendre totes les peticions de comunicació realitzades des del FrontEnd (Angular), el qual es la part on es va desenvolupar la interfície que interactua amb l'usuari.

**Verificació:** aquesta fase va servir per trobar possibles errors de codi en el sistema desenvolupat.

**Manteniment:** aquesta fase està relacionada amb la revisió i adaptació continua del sistema per part de l'usuari final i, per tant, queda fora de l'àmbit d'aquest projecte.

D'acord amb aquesta metodologia, es va poder establir que cada PAC que s'havia de realitzar a aquest projecte, havia de correspondre a una fase diferent del cicle de vida en cascada:

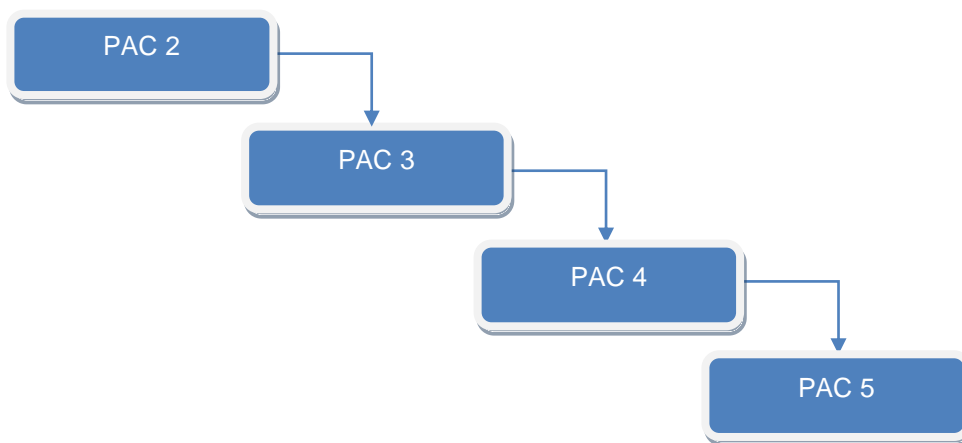


Figura 2: Cicle de vida desenvolupament en cascada del projecte

## 1.5. Planificació

La planificació del projecte es va desenvolupar en cinc fases, les quals van correspondre amb cada una de les cinc PAC's previstes en la assignatura i les seves dates claus. El conjunt de tasques també va venir derivat del conjunt de feines a realitzar en cada entrega, de tal manera que es pot veure reflectit en la següent taula:

Activitat	Duració	Data inici	Data Final
<b>PAC 1</b>	<b>5</b>	<b>17/02/2021</b>	<b>21/02/2021</b>
Planificació	1	17/02/2021	17/02/2021
Redacció document	5	17/02/2021	21/02/2021
<b>PAC 2</b>	<b>14</b>	<b>22/02/2021</b>	<b>07/03/2021</b>
Definició de projecte	4	22/02/2021	25/02/2021
Objectius generals	2	26/02/2021	27/02/2021
Metodologia i procés de treball	3	28/02/2021	02/03/2021
Planificació inicial	2	03/03/2021	04/03/2021
Pressupost	1	05/03/2021	05/03/2021
Estructura del document	2	06/03/2021	07/03/2021
<b>PAC 3</b>	<b>28</b>	<b>08/03/2021</b>	<b>04/04/2021</b>
Estudi de mercat	3	08/03/2021	10/03/2021
Proposta	3	11/03/2021	13/03/2021
Disseny de l'aplicació	15	14/03/2021	28/03/2021
Prototips	7	29/03/2021	04/04/2021
<b>PAC 4</b>	<b>35</b>	<b>05/04/2021</b>	<b>09/05/2021</b>
Disseny de la base de dades	7	05/04/2021	11/04/2021
Desenvolupament del BackEnd	7	12/04/2021	18/04/2021
Desenvolupament del FrontEnd	14	19/04/2021	02/05/2021
Correcció d'errors	4	03/05/2021	06/05/2021
Documentació i justificació del desenvolupament	3	07/05/2021	09/05/2021
<b>PAC 5</b>	<b>26</b>	<b>10/05/2021</b>	<b>07/06/2021</b>
Finalització del projecte	16	10/05/2021	25/05/2021
Finalització de la memòria	7	26/05/2021	01/06/2021
Elaboració d'una presentació del projecte	3	02/06/2021	04/06/2021
Elaboració d'una presentació del projecte en format vídeo	3	05/06/2021	07/06/2021
<b>Defensa virtual</b>	<b>5</b>	<b>14/06/2021</b>	<b>18/06/2021</b>
Defensa virtual	5	14/06/2021	18/06/2021

Taula 1: Planificació del projecte

### 1.5.1. Diagrama de Gantt

La taula anterior representa la previsió inicial del temps de desenvolupament estimat per a la realització de totes les tasques. Aquesta taula es pot veure de manera més visual i detallada mitjançant un diagrama de Gantt de cada fase, com es pot veure a continuació:

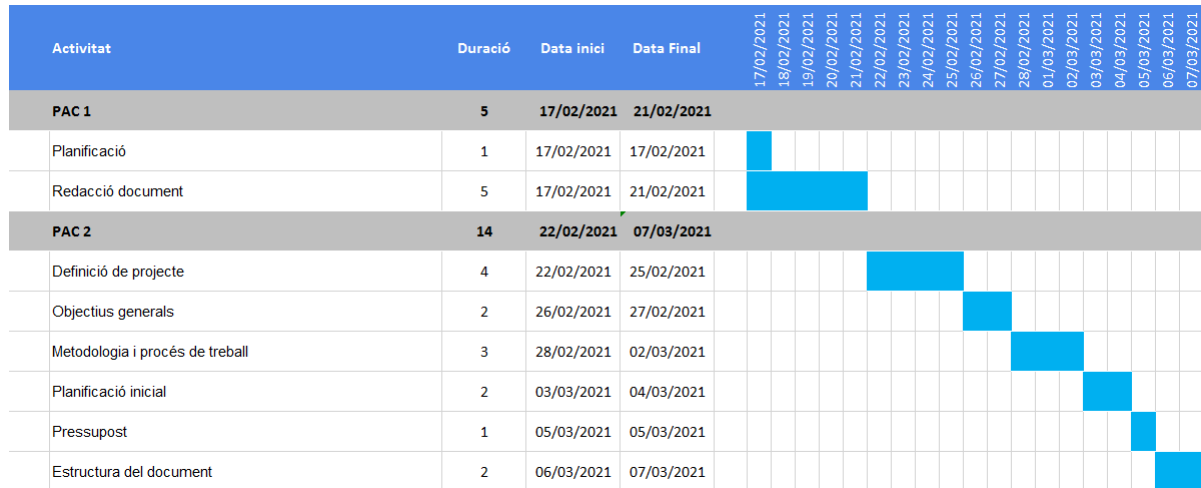


Figura 3: PAC1, PAC2. Diagrama de Gantt

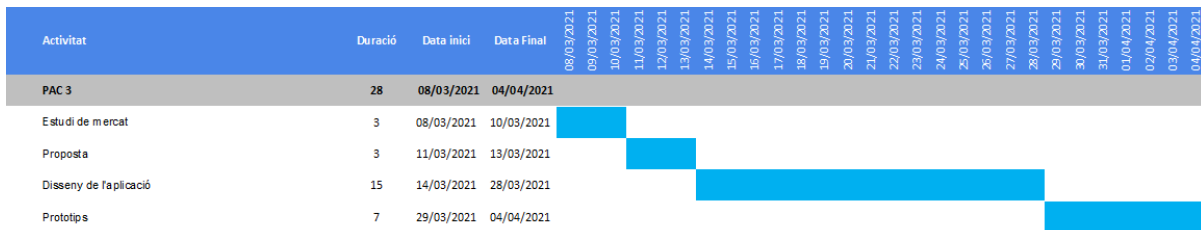


Figura 4: PAC3. Diagrama de Gantt

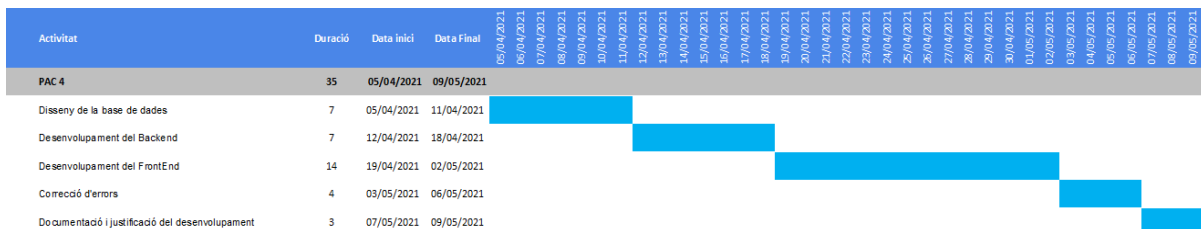


Figura 5: PAC4. Diagrama de Gantt

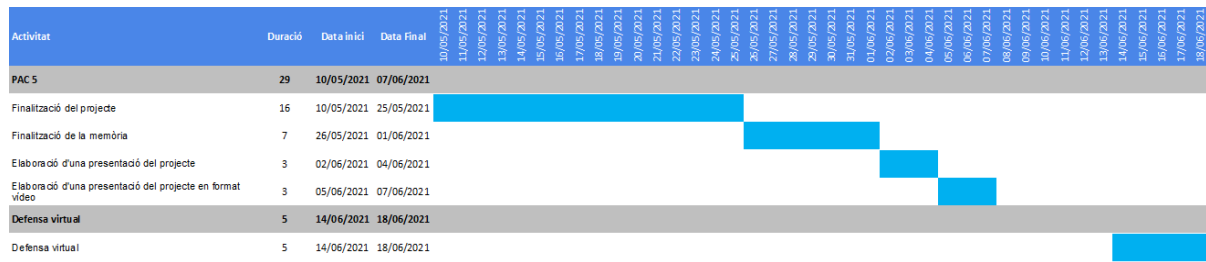


Figura 6: PAC5. Diagrama de Gantt

## 1.6. Pressupost

La planificació del projecte es va fer tenint en compte que l'equip tècnic per desenvolupar-ho constava únicament d'una sola persona. A partir d'aquí, encara que hi havia diferents tipus de tasques que pertanyien a àmbits diferents, es podia considerar que aquest projecte realitzat per una persona freelance, podia tenir un preu mig de 30€/h.

Treballs	Hores	Preu/h	Preu
<b>Anàlisi</b>			
Anàlisi de mercat	8,00	30,00	240,00
Definició requeriments tècnics	20,00	30,00	600,00
<b>Disseny</b>			
Definició arquitectura y usabilitat	15,00	30,00	450,00
Proposta gràfica	30,00	30,00	900,00
Creació de continguts	10,00	30,00	300,00
<b>Desenvolupament</b>			
Disseny i desenvolupament BBDD	20,00	30,00	600,00
Disseny i desenvolupament BackEnd	35,00	30,00	1050,00
Disseny i desenvolupament FrontEnd	60,00	30,00	1800,00
<b>Publicació</b>			
Publicació web	8,00	30,00	240,00
<b>TOTAL</b>			<b>6180,00</b>

Taula 2: Pressupost

## 1.7. Estructura de la resta del document

Aquesta memòria consta de set capítols en total. Cadascun d'aquests capítols pretén documentar cadascuna de les fases en les quals s'ha fonamentat aquest projecte.

Al **Capítol 2. Anàlisi de mercat** s'explica detalladament quin és el públic objectiu d'aquest projecte per tal de tenir clar des d'un primer moment quins requisits de disseny ha de tenir aquesta plataforma. De la mateixa forma, també s'inclou un estudi de plataformes similars de la competència, analitzant els seus punts forts així com les seves mancances.

Al **Capítol 3. Proposta** s'exposa la solució que presentem i els fets diferencials d'aquesta solució respecte a la competència.

Al **Capítol 4. Disseny** s'inclou totes les propostes gràfiques de les interfícies que mostren de manera visual com ha de quedar el producte final. Això també inclou definir la arquitectura de l'aplicació, el seu sistema de navegació, estils, etc.

Al **Capítol 5. Implementació** es fa una anàlisi del procés de desenvolupament i s'expliquen algunes de les funcionalitats del codi.

Al **Capítol 6. Demostració** s'inclou una explicació del funcionament de l'aplicació

Per últim, al **Capítol 7. Conclusions** es fan una sèrie de reflexions a mode de cloenda del treball per tal d'analitzar els objectius assolits, així com tot el procés de treball al llarg del projecte. També es fan suggeriments per analitzar les possibilitats d'ampliacions i millores en un futur.

## 2. Anàlisi de mercat

La pretensió d'aquest estudi és doble ja que, per un costat, s'identifica quin és el perfil objectiu teòric al qual està dirigida aquesta aplicació, informació necessària per establir els requisits i realitzar un disseny adequat. Per l'altra banda, analitzar com són els productes que actualment existeixen al mercat, quines són les seves fortaleses i quines les seves debilitats, de tal manera que, aquest anàlisi serveixi per evitar els errors d'altres persones i aprofitar els encerts.

### 2.1. Públic objectiu i perfils d'usuari

El servei que s'ofereix a través d'aquesta plataforma està dirigit sobretot a totes aquelles persones adultes que siguin propietàries d'immobles. Tanmateix, dintre del grup de persones adultes es pot especificar un subgrup prioritari, que serien aquelles persones entre 30 i 55 anys, donat que les persones joves tenen més problemes d'accés de compra d'un pis i que les persones més grans són més reticents o tenen més dificultat a l'ús d'eines en línia o sistemes diferents als coneguts.

Tanmateix, donat que en l'administració d'una comunitat intervenen també els administradors de finques, aquests es converteixen en stakeholders de l'aplicació i per tant, seran un perfil a tenir en compte en l'aplicació.

#### 2.1.1. Anàlisi del públic objectiu

Per fer un anàlisi del públic objectiu, presentem diverses targetes de diferents tipus de perfils d'usuari amb un escenari principal que hauran son representatius de la realitat i que s'hauran de tenir en compte alhora de la definició del lloc web:

##### Persona A

- **Nom:** Dídac Samper
- **Edat:** 46 anys
- **Professió:** Arquitecte
- **Tipus d'usuari:** President escala
- **Escenari principal:** vol tenir accés i estudiar els diferents pressupostos de treballs que s'han de fer al coberta de l'edifici. També vol saber en quin estat econòmic es troba la comunitat per fer front a aquestes despeses
- **Pors:** no té clar que els treballs a realitzar s'ajustin a la viabilitat econòmica de la escala
- **Solució habitual:** demanar-ho a l'administrador de finques.

**Persona B**

- **Nom:** Laura Ribas
- **Edat:** 52 anys
- **Professió:** Administrativa
- **Tipus d'usuari:** Propietària
- **Escenari principal:** ha trobat una fuita a la sala d'aigües al tornar a casa un dissabte a la tarda, i el president de l'escala no hi es.
- **Pors:** no té clar quina es la companyia d'assegurances de la finca ni els seu numero de pòlissa i no sap a qui trucar
- **Solució habitual:** cerca de proveïdors en Google

**Persona C**

- **Nom:** Gloria Perez
- **Edat:** 65 anys
- **Professió:** Jubilada
- **Tipus d'usuari:** Propietària
- **Escenari principal:** va fer una proposta que va ser acceptada en una reunió de veïns fa temps i que no s'ha executat mai, però ja no conserva l'acte d'aquella reunió
- **Pors:** no poder arribar a demostrar que la seva proposta ja va ser acceptada
- **Solució habitual:** mirar d'aconseguir aquest acte a través del president de la escala

**Persona D**

- **Nom:** Xavi Sans
- **Edat:** 49 anys
- **Professió:** Administrador de finques
- **Tipus d'usuari:** Administrador de finques
- **Escenari principal:** vol oferir més transparència a les diferents comunitats de veïns que gestiona
- **Pors:** no vol comprometre el seu sistema de gestió interna habitual i no vol fer una inversió en altres sistemes
- **Solució habitual:** desestimar la opció de donar el valor de la informació als seus clients



## 2.2. Competència/Antecedents

### 2.2.1. Colindar app

Colindar app<sup>4</sup> és una aplicació multi plataforma que permet l'administració de finques a través d'aquest sistema online. Un cop completat el registre dintre del sistema, accedim a un menú on hem completarem les nostres dades indicant la adreça, el pis i crearem la comunitat. A partir d'aquí, accedim a la pantalla d'inici.

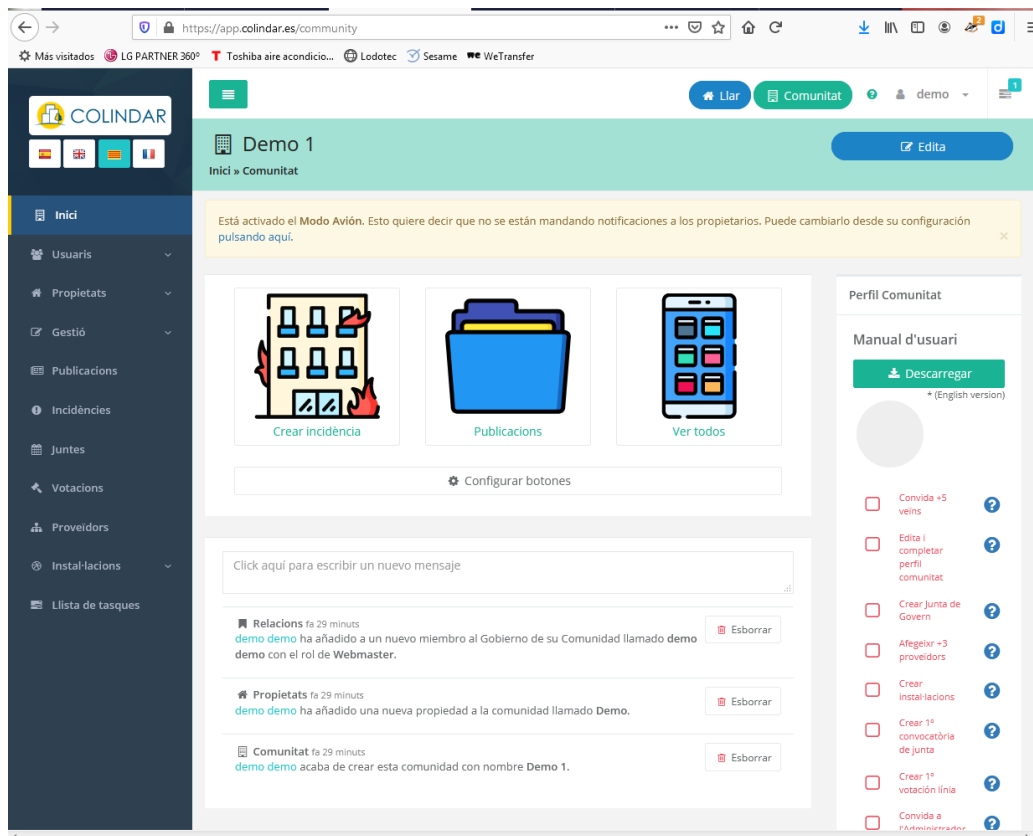


Figura 7: Colindar. Pantalla inici

A simple vista, el menú de navegació sembla senzill i amb bastants possibilitats, però quan comences a navegar observes que la corba d'aprenentatge no es tan fàcil com es podia imaginar i cal fer una investigació més a fons de totes les possibilitats que ofereix aquesta plataforma. Com podem observar, tenim una secció d'usuaris des d'on gestionar i contactar altres usuaris. Una secció de propietats per llistar els diferents veïns de la finca. Una secció de Gestió que es molt confusa però que sembla que serveix per administrar i configurar algunes opcions del sistema. També tenim una secció de Publicacions, una d'Incidències i un altre de Juntas que permet la creació d'aquests esdeveniments a l'usuari gràcies a un formulari.

<sup>4</sup> Colindar app. <https://colindar.com/>

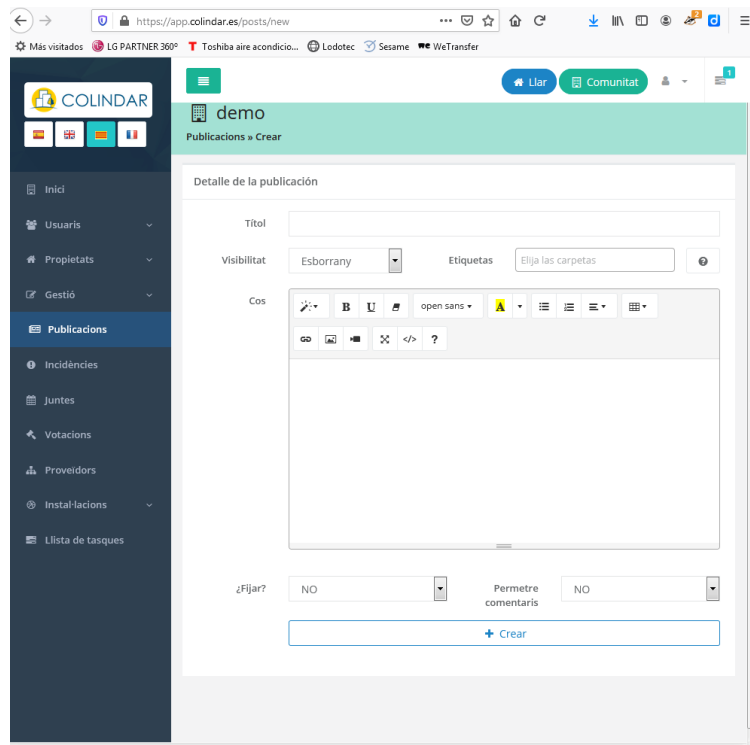


Figura 8: Colindar. Creació de Publicacions

Possiblement, aquesta aplicació té dues funcionalitats que destaquen sobre la resta, com són la possibilitat de fer votacions online i la segona és la gestió d'instal·lacions esportives, si n'hi ha, per poder fer reserves.

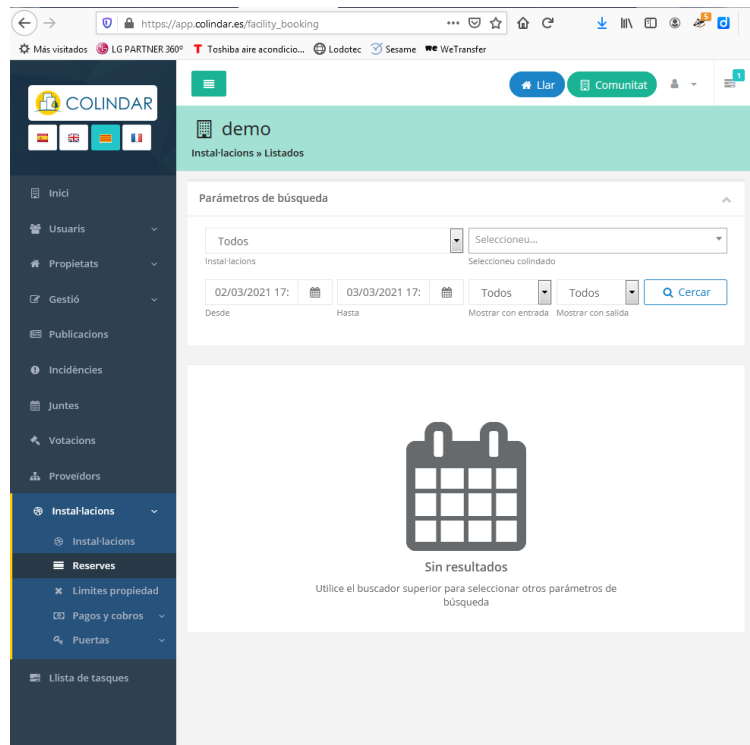


Figura 9: Colindar. Reservas d'instal·lacions

### Punts forts

- Disseny modern i visualment atractiu
- Té multi llenguatge
- La creació d'esdeveniments es fa a través de formularis que contempnen la majoria de les necessitats
- Té la opció de contactar per WhatsApp (prèvia subscripció de pagament)
- Té un tauler d'accés ràpid per veure els últims esdeveniments creats

### Punts dèbils

- La taxonomia en algunes ocasions es ambigua i poc entenedora
- Hi ha espais que costa d'entendre com funcionen i els formularis no són sempre les millors opcions
- Els formularis podrien ser més fàcils per a l'usuari
- El sistema de navegació no es del tot apropiat i es podria haver optimitzat millor els submenús
- El sistema ofereix unes funcionalitats limitades i que no tenen en compte totes les necessitats d'una comunitat de veïns
- El llistat de publicacions s'ordena per només per data. No es poden pujar arxius
- Si la comunitat no té instal·lacions, l'aplicació es una mica limitada

## 2.2.2. Fynkus

Fynkus<sup>5</sup> és una aplicació dissenyada per a telèfon mòbil que es pot descarregar tant de Google Play com de l'APP Store. Tal i com ells mateixos ens indiquen a la seva web, es l'administrador l'encarregat de registrar la comunitat de veïns i introduir els propietaris a través del correu electrònic.



Figura 10: Fynkus. Com funciona

<sup>5</sup> Fynkus. <https://www.fynkus.es/simulacion-propietario/>

Un cop registrat, es pot observar un menú bastant clar i amb diferents opcions on podem començar a operar

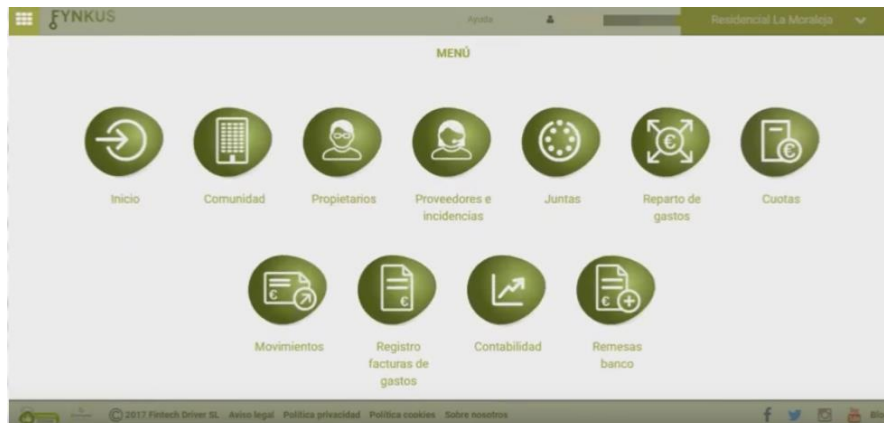


Figura 11: Fynkus. Pantalla inicial

Com es pot observar, aquesta aplicació destinada sobretot a la gestió de l'administració de finques, on el gruix principal d'accions es per poder gestionar de manera autònoma tota la comptabilitat. Tanmateix, el sistema també permet la creació de juntes de veïns, així com la creació d'incidències amb la conseqüent assignació d'un proveïdor per a la seva solució. També es poden donar d'alta i visualitzar contractes de manera que es fàcil fer el seguiment d'aquells serveis contractats.

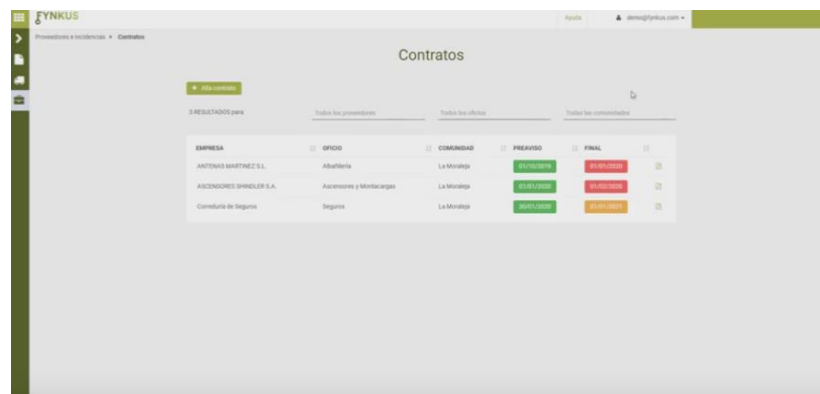


Figura 12: Fynkus. Pantalla inicial

### Punts forts

- El disseny és molt simple i clarificador amb la qual cosa saps perfectament el que estàs fent a cada moment
- La taxonomia i el sistema de navegació estan molt ben pensats i no donen marge a l'error
- Hi ha moltes funcionalitats que permeten tenir un control molt exhaustiu de tot allò relacionat amb una comunitat de veïns
- Deixa pujar i baixar diferents arxius
- Permet portar un control financer detallat




Punts dèbils

- És una pàgina web més dirigit a aquelles comunitats que no tenen administrador de finques i es volen autogestionar la comunitat o segones residències
- Requereix que tots els usuaris tinguin bon domini d'eines en línia
- Aquest sistema de gestió no es el més adient quan hi ha administrador de finques als quals, els hi pot suposar un increment important de la carrega de treball
- No té opció d'exportar/importar dades

**2.2.3. Conclusions**

Existeixen al mercat diferents opcions per a que una comunitat de propietaris prengui el control de la seva activitat, i podem trobar des de aplicacions senzilles que donen certa autonomia als propietaris, fins a solucions més complexes que aporten una gestió completa i, si es vol, independent. D'aquí podem extreure la conclusió que hi ha una gran majoria de comunitats que són gestionades per administradors de finques i cal cercar solucions per a què les comunitats de veïns tinguin un portal informatiu propi i que sigui assumible (es a dir, que no suposi un increment del volum de feina) per a la part de l'administrador de finques.

Funcionalitats	Colindar	Fynkus	Propietarispuntcat
Permet autonomia de gestió a l'usuari			
Ordenació de documents			
Llistat d'esdeveniments i proveïdors			
Llistat de contractes			
Servei de missatgeria			
Telèfons habituals			
Tauler anuncis			
Multi llenguatge			
Gestió comptabilitat			
Facilitat d'ús per part dels propietaris			
Facilitat d'ús per part dels administradors			
Compatibilitat amb altres sistemes			

 No
  Sí, amb dificultats
  Sí

Taula 3: Comparativa aplicacions

### 2.3. Anàlisi DAFO

A continuació, es presenta una anàlisi de les característiques internes del projecte, tenint en compte l'estudi realitzat anteriorment sobre altres possibilitats existents al mercat.

Aquest anàlisi es mostra a través d'una matriu de fortaleces i debilitats (anàlisi intern) en front a les oportunitats i amenaces que genera l'entorn extern.

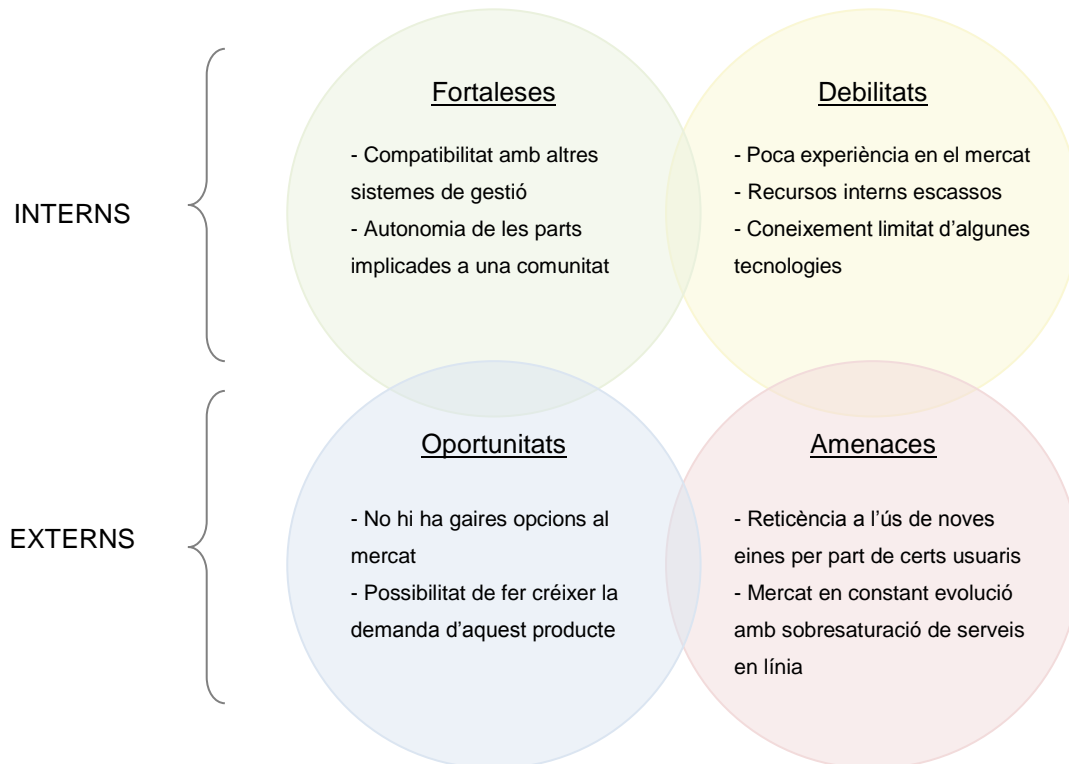


Figura 13: Anàlisi DAFO

## 3.Proposta

### 3.1. Definició d'objectius/especificacions del producte

La posada en marxa d'aquest projecte té com a principal objectiu fer d'enllaç comunicatiu entre una comunitat de propietaris i l'administrador de finques. Aquesta aplicació ha d'estar sobretot al servei dels propietaris, però ha de permetre incorporar als gestors de finques sense que els hi suposi una incomoditat ni incompatibilitats amb la seva forma interna de treballar. Per aconseguir-ho, l'aplicació haurà de recollir els següents serveis:

- Facilitar el màxim de dades de l'administrador
- Llistar el màxim de veïns possibles i quins ocupen els diferents càrrecs
- Recopilar i ordenar diferent tipus d'informació:
  - Actes
  - Serveis contractats
  - Pressupostos
  - Factures
  - Pòlisses
  - Butlletins i comunicats
  - Resums econòmics
  - Altres
- Facilitar llistat de telèfons d'interès i d'emergència
- Gestió de proveïdors
- Juntes de veïns
- Tauler d'anuncis
- Configuració del perfil d'usuari
- Treballs pendents

#### 3.1.1. Tipus de rols d'usuari

Tots els usuaris han de passar per un procés de registre a l'aplicació. Inicialment, hi ha una persona que és l'encarregada de registrar la comunitat i que té el rol d'**administrador**, el qual pot ser qualsevol persona de la comunitat i que es l'encarregat d'afegir veïns a la seva comunitat. L'administrador de finques té el rol de **gestor**, mentre que els rols de **president** i el **secretari** de la comunitat tenen les mateixes funcionalitats que el gestor. També existeix el rol de **proveïdor**, el qual es específic d'aquest grup, i només per identificar-los i sense cap funcionalitat en el sistema. Per últim, existeix el rol de **propietari** per a la resta d'usuaris.

Funcions rol administrador, gestor, president, secretari

- Donar d'alta / baixa nous veïns
- Canviar perfils d'usuaris
- Pujar / baixar / eliminar documents
- Realitzar notificacions
- Crear esdeveniments
- Afegir / eliminar proveïdors

Funcions rol propietari

- Lectura d'esdeveniments i documentació
- Baixar documents

### **3.2. Model de negoci**

La monetització d'aquesta aplicació no forma part dels objectius d'aquest projecte, tot i així, en un anàlisi de la seva viabilitat econòmica podem tenir en compte diferents opcions.

Despeses

El cost del projecte està directament associat al temps de mà d'obra necessària per al seu desenvolupament. Per a la posada en marxa de l'aplicació s'haurà de fer una inversió econòmica relativa al cost d'un allotjament web que sigui adient a les necessitats d'una pàgina com aquesta.

Encara que no estigui contemplat en aquesta memòria, es podrien afegir més despeses en el cas de realitzar campanyes de màrqueting o posicionament web.

Beneficis econòmics

Com he comentat anteriorment, aquest projecte no té com a prioritat la obtenció de beneficis econòmics, però de totes maneres, es podria monetitzar per diferents formes:

- Subscripcions. Es permetria la subscripció gratuïta per a comunitats amb menys de cinc veïns registrats. A partir d'aquí, s'establiria diversos nivells de pagament en funció del numero de persones de la comunitat.
- Pagaments per obtenir rellevància i visualització de proveïdors.
- Publicació d'anuncis personalitzats.



## 4. Disseny

### 4.1. Arquitectura general de l'aplicació

L'aplicació consta de de tres serveis diferenciats:

#### Base de dades

On s'emmagatzema tota la informació de caràcter general en forma de taules relacionades. Per aquest propòsit es fa servir un servei de bases de dades relacional.

#### API REST

On s'implementa una API REST, de manera que fa servir el protocol HTTP per atendre totes les comunicacions provinents del FrontEnd, que bàsicament consisteixen en peticions CRUD<sup>6</sup>, que s'implementen gràcies als mètodes GET, POST, PUT i DELETE.

#### FrontEnd

On s'implementa les funcionalitats del costat client, que permeten la interacció amb l'usuari. Una vegada que detecta qualsevol esdeveniment per part de l'usuari, connecta amb la API REST per a que aquesta atengui la petició de comunicació de la manera apropiada.

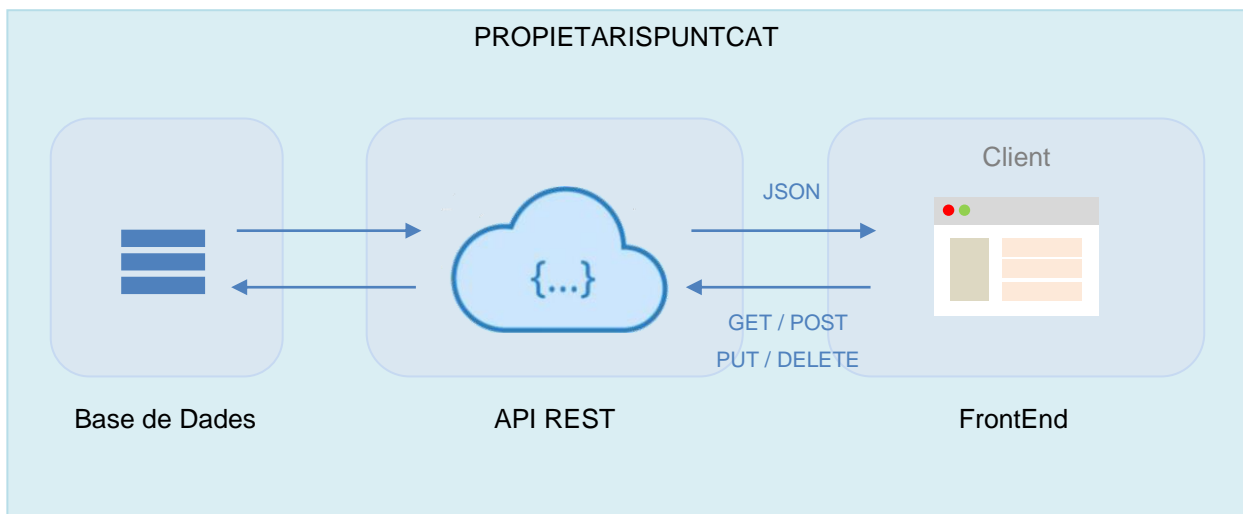


Figura 14: Arquitectura general de l'aplicació

<sup>6</sup> CRUD. Acrònim que fa referència a les funcions principals que s'han d'implementar en bases de dades: Create, Read, Update, Delete (Crear, Llegir, Actualitzar i Eliminar).

## 4.2. Arquitectura de la informació

### 4.2.1. Base de dades

Una de les parts principals del sistema és la seva base de dades que es on es guarden totes les dades recollides. Aquesta s'ha desenvolupat amb una plataforma un sistema de gestió de dades relacional, multifilar i multiusuari, com és MySQL, el qual es un programari lliure (té llicència GPL) i és compatible amb aplicacions de tota mena (web, d'escriptori o d'altres).

Gràcies a les consultes que permet el llenguatge SQL, podem atendre totes les peticions CRUD que provenen des del FrontEnd, a través de la API REST.

Per tal de disseny de la base de dades, en primer terme, es van identificar les diferents entitats que requerien les diferents funcionalitats del sistema, així com les seves relacions, de manera que es va poder realitzar el següent diagrama que mostra les diferents taules contingudes i les diferents dades que s'emmagatzemen:

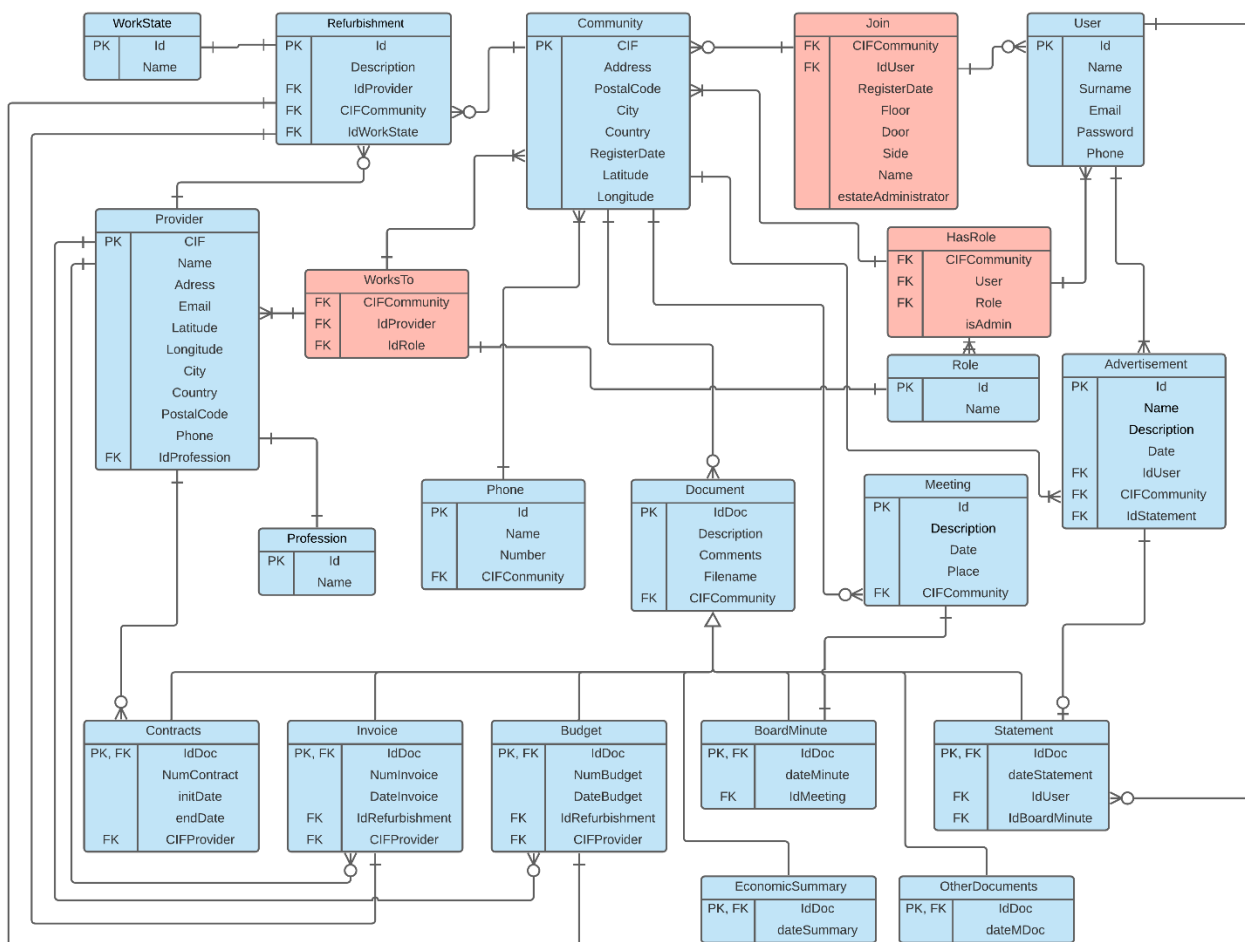


Figura 15: Diagrama Base de Dades

En aquest diagrama es mostren les següents taules:

Taula Entitats	Descripció
<b>User</b> (Usuari)	Permet guardar totes les dades necessàries de l'usuari: ID (PK <sup>7</sup> ), Nom, Cognom, Correu Electrònic, Contrasenya i Telèfon
<b>Community</b> (Comunitat)	Taula on es guarden cadascuna de les diferents comunitats creades. Conté les dades següents: CIF (PK), Adreça, Codi Postal, Ciutat, País, Data de registre, Latitud i Longitud
<b>Role</b> (Rol)	Taula que emmagatzema els diferents rols que poden tenir els diferents usuaris de l'aplicació (Administrador, Gestor, President, Secretari, Proveïdor, Propietari), amb les següents dades: ID (PK), Nom
<b>Provider</b> (Proveïdor)	Taula on es guarden cadascuna de les diferents proveïdors donats d'alta. Conté amb les dades següents: CIF (PK), Nom, Adreça, Correu Electrònic, Codi Postal, Ciutat, País, Data de registre, Latitud i Longitud , Id Professi6 (FK <sup>8</sup> )
<b>Profession</b> (Professi6)	Taula on es guarden cadascun dels diferents oficis que pot tenir un proveïdor. Conté els atributs següents: ID (PK), Nom
<b>Phone</b> (Telèfon)	Taula on es guarden els diferents telèfons que es mostren a la pàgina de telèfons d'interès. Conté els atributs següents: ID (PK), Nom, Número
<b>Document</b> (Document)	Taula, que fa de superclasse <sup>9</sup> d'altres taules, i que conté els atributs següents: IdDoc (PK), Descripci6, Comentaris, Nom Arxiu, CIF Comunitat (FK)
<b>Contract</b> (Contracte)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus Contracte. Conté els atributs següents: IdDoc (PK, FK), Numero de Contracte, Data Inici, Data Venciment, CIF Proveïdor (FK)
<b>Invoice</b> (Factura)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus <i>Factura</i> . Conté els atributs següents:

<sup>7</sup> PK. Primary Key (Clau Primària) serveix per identificar de manera unívoca una fila de cada taula de manera que no hi puguin haver dues files duplicades.

<sup>8</sup> FK. Foreign Key (Clau Forana) és l'atribut que relaciona una fila d'una taula amb la clau primària (PK) d'un altre.

<sup>9</sup> Ens podem trobar amb entitats que tenen una relació de generalitzaci6/especialitzaci6, també conegudes com superclasse/subclasse de manera que la superclasse conté les característiques comunes a totes les entitats i les subclasses contenen les específiques.

	IdDoc (PK, FK), Numero Factura, Data Factura, CIF Proveïdor (FK), Id Reforma (FK)
<b>Budget</b> (Pressupost)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus <i>Pressupost</i> . Conté els atributs següents: IdDoc (PK, FK), Numero Pressupost, Data Pressupost, CIF Proveïdor (FK), Id Reforma (FK)
<b>BoardMinute</b> (Acta)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus <i>Acta</i> . Conté els atributs següents: IdDoc (PK, FK), Data, ID Junta (FK)
<b>EconomicSummary</b> (Resum econòmic)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus <i>Econòmics</i> . Conté els atributs següents: IdDoc (PK, FK), Data
<b>Statement</b> (Comunicat)	Taula, subclasse de la classe <i>Document</i> , on es guarden els documents tipus <i>Comunicat</i> . Conté els atributs següents: IdDoc (PK, FK), Data, ID Usuari (FK), ID Acta (FK)
<b>otherDocument</b> (Altres documents)	Taula, subclasse de la classe <i>Document</i> , on es guarden altres documents no relacionats amb les anteriors taules. Conté els atributs següents: IdDoc (PK, FK), Data
<b>Refurbishment</b> (Reforma)	Taula on es guarden cadascun dels diferents treballs que s'han fet o estan pendents a la comunitat. Conté els atributs següents: ID (PK), Descripció, ID Proveïdor (FK), CIF Comunitat (FK), ID Estat Treballs(FK)
<b>WorkState</b> (Estat treballs)	Taula on es guarden l'estat dels treballs relacionats amb la comunitat. Conté els atributs següents: ID (PK), Nom
<b>Meeting</b> (Junta)	Taula on es guarden cadascuna de les diferents juntes de veïns creades. Conté els atributs següents: ID (PK), Descripció, Data, Lloc, CIF Comunitat (FK)
<b>Advertisement</b> (Tauler Anuncis)	Taula on es guarden els missatges del tauler d'anuncis. Conté els atributs següents: ID (PK), Nom, Descripció, ID Usuari(FK), ID Comunitat (FK), CIF Comunitat (FK)

Taula 4: Base de dades: Taula d'Entitats

Taula Relacions	Descripció
<b>Join</b>	Taula que relaciona la comunitat amb l'usuari. També conté dades complementaries de l'usuari (pis, porta i escala). Conté els atributs següents: ID Usuari(FK), ID Comunitat (FK), Nom, Data Registre, Pis, Porta, Escala, Administrador Finques
<b>HasRole</b>	Taula que relaciona la comunitat, l'usuari i el rol que té a dintre d'aquesta. Conté els atributs següents: ID Usuari(FK), ID Comunitat (FK), ID Rol (FK), Administrador Web
<b>WorksTo</b>	Taula que relaciona la comunitat amb els proveïdors i el rol que tenen a dintre d'aquesta. Conté els atributs següents: ID Proveïdor(FK), ID Comunitat (FK), ID Rol (FK)

Taula 5: Base de dades: Taula de Relacions

## 4.2.2. Arbre de Navegació

En les següents il·lustracions, es detalla el mapa general de l'arbre de navegació de l'aplicació (per veure-ho de manera més detallada, s'adjunten sub-diagrames de les principals parts):

### Arbre de navegació. Mapa General

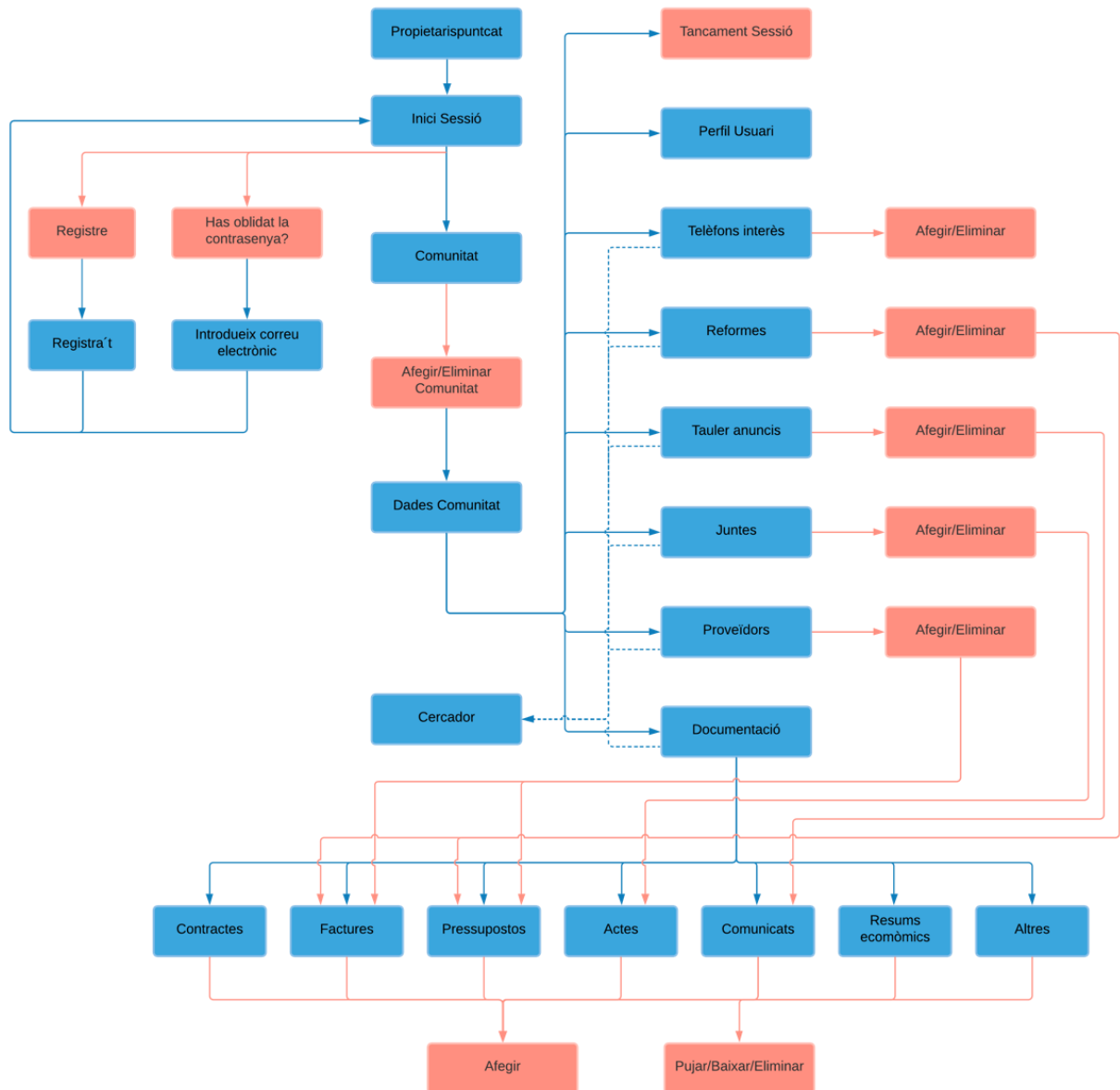


Figura 16: Arbre de navegació. Mapa General

Arbre de navegació. Inici de sessió

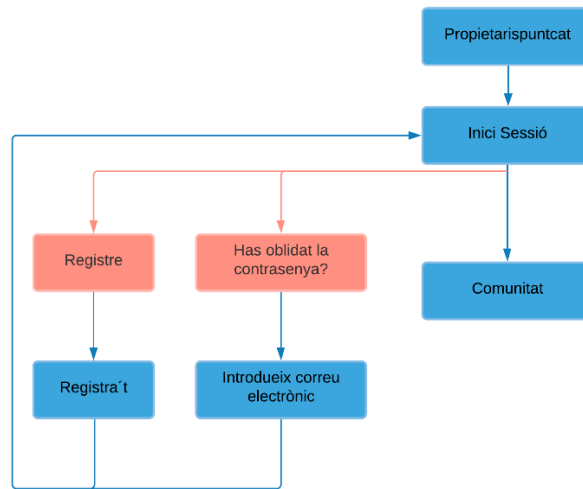


Figura 17: Arbre de navegació. Inici de sessió

Arbre de navegació. Comunitat (pàgina d'inici):



Figura 18: Arbre de navegació. Comunitat

Arbre de navegació. Documentació, Reformes, Proveïdors, Juntes i Tauler d'Anuncis:

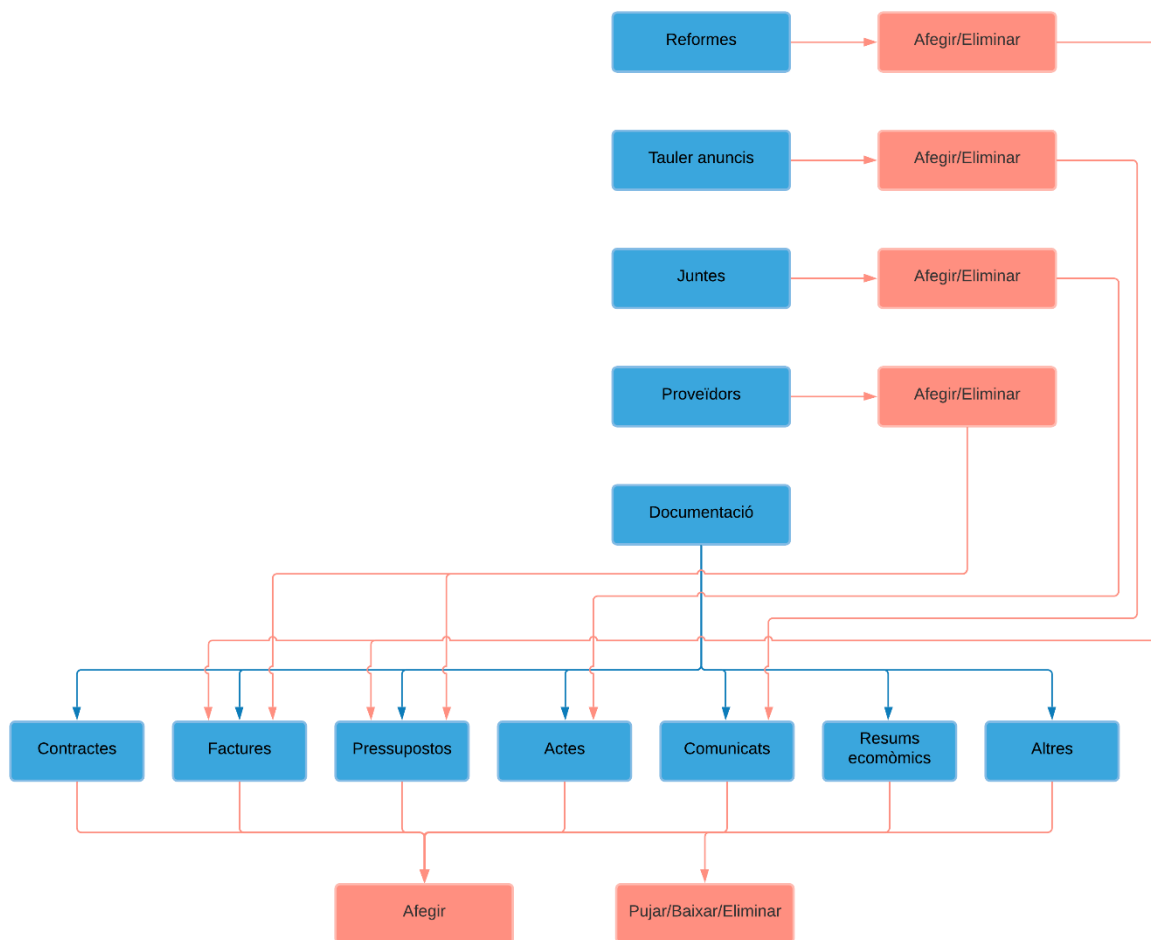


Figura 19: Arbre de navegació. Documentació, Reformes, Proveïdors, Juntes i Tauler Anuncis

## 4.3. Llenguatges de programació i APIs utilitzats

### 4.3.1. API's: Geocoding

L'aplicació necessita fer servir API's de tercers per tal de poder mostrar correctament tots els serveis oferts. En primer lloc, per obtenir les coordenades de les diferents adreces que es guarden a la base de dades es fa servir la API Geocoding<sup>10</sup> de Google. En el moment que es fa la petició des del servidor, s'obtenen un conjunt de dades en format JSON<sup>11</sup> entre els quals ens trobem els camps Latitud i Longitud que són enviats automàticament al camp en concret de la taula de la base de dades.

<sup>10</sup> Geocoding. <https://developers.google.com/maps/documentation/geocoding/overview>

<sup>11</sup> JSON. <https://en.wikipedia.org/wiki/JSON>



### 4.3.2. API REST

L'API REST està desenvolupat amb NodeJS, que és un entorn de programació basat en JavaScript i el qual proporciona una arquitectura orientada a esdeveniments, amb entrades/sortides asíncrones per tal de minimitzar el temps d'execució del sistema i maximitzar l'escalabilitat.

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Figura 20: NodeJS: Exemple de creació d'un servidor

Encara que NodeJS permet la creació de servidors, s'ha fet servir Express<sup>12</sup>, un entorn de treball de NodeJS, ja que aquest proveeix d'un conjunt de funcionalitats que permeten crear aplicacions web de manera flexible, robusta i de manera fàcil i transparent.

```
const express = require('express')
const app = express()
const port = 3000

app.get('/', (req, res) => {
  res.send('Hello World!')
})

app.listen(port, () => {
  console.log(`Example app listening at http://localhost:${port}`)
})
```

Figura 21: Express: Exemple d'ús

Altre part important de l'API REST, ha estat afegir protecció a les contrasenyes establertes per cada usuari. Mitjançant BCrypt<sup>13</sup>, s'ha afegit una capa de seguretat per la qual totes les contrasenyes s'han encriptat abans de ser guardades a la base de dades. A banda, s'ha fet servir JSON WEB Token<sup>14</sup> de manera que es realitza una operació criptogràfica sobre la contrasenya per la qual, el resultat (token) serveix per verificar la identificació de l'usuari. Cada token està compost per tres parts, *header*,

<sup>12</sup> Express. <http://expressjs.com/>

<sup>13</sup> BCrypt. <https://www.npmjs.com/package/bcrypt>

<sup>14</sup> JSON WEB Token. <https://www.npmjs.com/package/jsonwebtoken>

*payload*, *signature*, separades per un punt. Amb el *header* es defineix el tipus d'algoritme que s'empra durant la operació criptogràfica i el tipus de token. El *payload* conté la informació que es vol protegir. A la *signature* es codifica de nou el *header* i el *payload* juntament amb una clau secreta de manera que es verifica i garanteix que el missatge no ha canviat en cap moment.

### 4.3.3. EndPoints

A continuació, es detallen els diferents endPoints<sup>15</sup> que s'han fet servir en aquesta aplicació:

/login			
Descripció	Permet fer l'autenticació de l'usuari i crea un token per verificar la identitat de l'usuari durant la sessió.		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	-	-	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Dades usuari Token	
Error	400	Missatge: "Error en la validació de l'usuari"	
Error	401	Missatge: "No hem trobat cap usuari amb aquestes credencials"	

Taula 6: EndPoint: POST login

/user			
Descripció	Permet fer el registre d'un usuari		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	-	-	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "T'has enregistra't correctament"	
Error	400	Missatge: "Aquest usuari ja existeix"	

<sup>15</sup> EndPoint: <https://www.paloaltonetworks.com/cyberpedia/what-is-an-endpoint>

Error	500	Missatge: "Hi ha hagut algun error i no s'ha pogut crear l'usuari"
-------	-----	--

Taula 7: EndPoint: POST user

/user/:id			
Descripció	Permet trobar un usuari concret pel seu id		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Usuari	
Error	404	Missatge: "L'usuari no existeix"	

Taula 8: EndPoint: GET user/id

/changeRole/:id/:cif			
Descripció	Permet canviar el rol que té un usuari en una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	Id (usuari), cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "S'ha actualitzat el rol d'aquest usuari"	
Error	404	Missatge: "No s'ha pogut actualitzar el rol d'aquest usuari"	

Taula 9: EndPoint: PUT changeRole/id/cif

/user/:id			
Descripció	Permet editar i actualitzar un usuari		
Mètode	Capçalera	Paràmetres	Tipus de resposta

PUT	Authorization: token	Id (usuari)	JSON
<b>Respostes</b>			
Cas	Status	Resposta	
Èxit	200	Missatge: "L'usuari s'ha actualitzat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut actualitzar l'usuari"	

Taula 10: EndPoint: PUT user/id

<b>/user/id</b>			
Descripció	Permet esborrar un usuari existent		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	Id (usuari)	JSON
<b>Respostes</b>			
Cas	Status	Resposta	
Èxit	200	Missatge: "L'usuari s'ha esborrat correctament"	
Error	500	Missatge: "No s'ha pogut esborrar l'usuari"	

Taula 11: EndPoint: DELETE user/id

<b>/password</b>			
Descripció	Permet actualitzar la contrasenya		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	-	-	JSON
<b>Respostes</b>			
Cas	Status	Resposta	
Èxit	200	Missatge: "L'usuari s'ha actualitzat correctament"	

Error	404	Missatge: "Aquest usuari no existeix"
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut actualitzar l'usuari"

Taula 12: EndPoint: PUT password

/communities/:id			
<b>Descripció</b>	Permet trobar una llistat de les comunitats a les quals, un usuari pertany		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
GET	Authorization: token	Id (usuari)	JSON
Respostes			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Comunitats	
Error	500	Missatge: "Error al buscar les teves comunitats"	

Taula 13: EndPoint: GET communities/id

/joinCommunity/:cif			
<b>Descripció</b>	Permet obtenir un llistat d'usuaris que pertanyen a una comunitat		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
GET	Authorization: token	cif (comunitat)	JSON
Respostes			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Usuaris	
Èxit	200	Missatge: "No s'ha trobat cap usuari"	
Error	500	Missatge: "Hi ha hagut algun error i no s'ha pogut fer la cerca"	

Taula 14: EndPoint: GET joinCommunity/cif

<b>/joinCommunity/:id</b>			
<b>Descripció</b>	Permet realitzar una petició per accedir a una comunitat		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
POST	Authorization: token	id (usuari)	JSON
<b>Respostes</b>			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Missatge: "Les teves dades són correctes"	
Error	400	Missatge: "Les teves dades ja han enviat una sol·licitud amb anterioritat"	
Error	500	Missatge: "No s'ha pogut realitzar la teva petició"	

Taula 15: EndPoint: POST joinCommunity/cif

<b>/joinCommunity</b>			
<b>Descripció</b>	Permet a l'administrador d'una comunitat afegir o refusar un usuari		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
PUT	Authorization: token	-	JSON
<b>Respostes</b>			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Missatge: "La teva petició s'ha realitzat correctament"	
Error	500	Missatge: "Les teves dades ja han enviat una sol·licitud amb anterioritat"	

Taula 16: EndPoint: PUT joinCommunity

<b>/joinCommunity/:user/:cif</b>			
<b>Descripció</b>	Permet esborrar a un usuari d'una comunitat		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
DELETE	Authorization: token	user (id usuari), cif (comunitat)	JSON

Respostes		
Cas	Status	Resposta
Èxit	200	Missatge: "La teva petició s'ha realitzat correctament"
Error	500	Missatge: "No s'ha pogut realitzar la teva petició"

Taula 17: EndPoint: DELETE joinCommunity/user/cif

/pendingAccess/id			
Descripció	Permet obtenir llistat de comunitats en les quals, un usuari esta pendent d'accés		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	user (id usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Comunitats	
Error	500	Missatge: "Error en la cerca de peticions pendents"	

Taula 18: EndPoint: GET pendingAccess/id

/requestAccessCommunity/:id			
Descripció	Permet al administrador d'una comunitat obtenir un llistat d'usuaris que han demanat accés		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Usuaris	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 19: EndPoint: GET requestAccessCommunity/id

/community/:adress			
Descripció	Permet trobar una comunitat a la barra de cerca per la seva direcció		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	address	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Comunitat	
Error	500	Missatge: "Hi ha hagut algun error i no s'ha pogut fer la cerca"	

Taula 20: EndPoint: GET community/address

/community/:id			
Descripció	Permet registrar una comunitat nova		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La comunitat ha quedat registrada"	
Error	400	Missatge: "Aquesta comunitat ja existeix"	
Error	500	Missatge: "No s'ha pogut crear la comunitat"	

Taula 21: EndPoint: POST community/id

/community/:id			
Descripció	Permet registrar una comunitat nova		
Mètode	Capçalera	Paràmetres	Tipus de resposta



PUT	Authorization: token	id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La comunitat s'ha actualitzat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "No s'ha pogut actualitzar la comunitat"	

Taula 22: EndPoint: PUT community/id

/community/:id			
<b>Descripció</b>	Permet registrar una comunitat nova		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La comunitat s'ha esborrat correctament"	
Error	500	Missatge: "No s'ha pogut esborrar la comunitat"	

Taula 23: EndPoint: DELETE community/id

/phone/:cif			
<b>Descripció</b>	Permet obtenir llistat de telèfons d'una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Telèfons	

Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"
-------	-----	---

Taula 24: EndPoint: GET phone/cif

/phone/:cif/:string			
Descripció	Permet fer la cerca d'un telèfon concret pel seu nom		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	cif (comunitat), string	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Telèfon	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 25: EndPoint: GET phone/cif/string

/phone/:cif			
Descripció	Permet afegir un número de telèfon nou		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	id (usuari)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 26: EndPoint: POST phone/cif

/phone/:cif			
Descripció	Permet actualitzar un número de telèfon		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 27: EndPoint: PUT phone/cif

/phone/:id			
Descripció	Permet esborrar un número de telèfon		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	id (telèfon)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "El telèfon s'ha esborrat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"	

Taula 28: EndPoint: DELETE phone/id

/provider/:cif			
Descripció	Permet obtenir llistat de proveïdors d'una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta

GET	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Proveïdors	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat els teus proveïdors"	

Taula 29: EndPoint: GET provider/cif

/provider/:cif/:string			
Descripció	Permet fer la cerca d'un telèfon concret pel seu nom		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	cif (comunitat), string	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Proveïdor	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 30: EndPoint: GET provider/cif/string

/provider/:cif			
Descripció	Permet afegir un proveïdor nou a la comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "El proveïdor ha quedat registrat"	
Error	400	Missatge: "Aquestes dades ja existeixen"	

Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"
-------	-----	---

Taula 31: EndPoint: POST provider/cif

/provider/:cif			
Descripció	Permet actualitzar un proveïdor		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 32: EndPoint: PUT provider/cif

/provider/:cif			
Descripció	Permet esborrar un proveïdor		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "El proveïdor s'ha esborrat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"	

Taula 33: EndPoint: DELETE provider/cif

/provider/:provider/:cif			
<b>Descripció</b>	Permet actualitzar el rol d'un proveïdor		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "No s'ha pogut actualitzar el rol d'aquest proveïdor"	

Taula 34: EndPoint: PUT provider/provider/cif

/profession			
<b>Descripció</b>	Permet obtenir llistat de professions		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
GET	Authorization: token	-	JSON
Respostes			
<b>Cas</b>	<b>Status</b>	<b>Resposta</b>	
Èxit	200	Professions	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 35: EndPoint: GET profession

/profession			
<b>Descripció</b>	Permet afegir una professió nova a la comunitat		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>

POST	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La professió ha quedat registrada"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar"	

Taula 36: EndPoint: POST profession

/documents/:cif/:documentType			
Descripció	Permet obtenir llistat de proveïdors d'una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat), documentType	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Documents	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 37: EndPoint: GET documents/cif/documentType

/documents/:cif/:documentType/:string			
Descripció	Permet obtenir llistat de documents, segons el seu tipus		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat), documentType , string	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Document	

Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"
-------	-----	---

Taula 38: EndPoint: GET documents/cif/documentType/string

/documents/:cif			
Descripció	Permet afegir un document nou a la comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "El document ha quedat registrat"	
Error	400	Missatge: "Aquest document ja existeix"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"	

Taula 39: EndPoint: POST documents/cif

/documents/:cif			
Descripció	Permet actualitzar un document		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"	

Taula 40: EndPoint: PUT documents/cif



/documents/:cif			
Descripció	Permet esborrar un document		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "El document s'ha esborrat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"	

Taula 41: EndPoint: DELETE documents/cif

/refurbishment/:cif			
Descripció	Permet obtenir llistat de reformes d'una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Reformes	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les teves reformes"	

Taula 42: EndPoint: GET refurbishment /cif

/refurbishment/:cif/:string			
Descripció	Permet fer la cerca d'un proveïdor concret		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat), string	JSON
Respostes			

Cas	Status	Resposta
Èxit	200	Reforma
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"

Taula 43: EndPoint: GET refurbishment /cif/string

/refurbishment/:cif			
Descripció	Permet afegir una reforma nova a la comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La reforma ha quedat registrada"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"	

Taula 44: EndPoint: POST refurbishment /cif

/refurbishment/:cif			
Descripció	Permet actualitzar una reforma		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	

Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"
-------	-----	---

Taula 45: EndPoint: PUT refurbishment /cif

/refurbishment/:cif			
Descripció	Permet esborrar una reforma		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La reforma s'ha esborrat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"	

Taula 46: EndPoint: DELETE refurbishment /cif

/workstate			
Descripció	Permet obtenir llistat de reformes d'una comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Estats	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 47: EndPoint: GET workstate

/advertisement/:cif			
Descripció	Permet obtenir les missatges deixats al tauler d'anuncis		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatges	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les teves dades"	

Taula 48: EndPoint: GET advertisement/cif

/advertisement/:cif/:string			
Descripció	Permet fer la cerca d'un missatge per una cerca concreta		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat), string	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 49: EndPoint: GET advertisement/cif/string

/advertisement			
Descripció	Permet afegir una missatge al tauler d'anuncis		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	-	JSON
Respostes			

Cas	Status	Resposta
Èxit	200	Missatge: "El teu anunci ha quedat registrat"
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"

Taula 50: EndPoint: POST advertisement

/advertisement /:cif			
<b>Descripció</b>	Permet actualitzar un missatge del tauler d'anuncis		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar la operació"	

Taula 51: EndPoint: PUT advertisement/cif

/advertisement/:cif			
<b>Descripció</b>	Permet esborrar un missatge del tauler d'anuncis		
<b>Mètode</b>	<b>Capçalera</b>	<b>Paràmetres</b>	<b>Tipus de resposta</b>
DELETE	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "L'anunci s'ha esborrat correctament"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"	

Taula 52: EndPoint: DELETE advertisement/cif

/meeting/:cif			
Descripció	Permet obtenir llistat de juntes de la comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Juntes	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 53: EndPoint: GET meeting/cif

/meeting/:cif/:string			
Descripció	Permet fer la cerca d'una junta en concret		
Mètode	Capçalera	Paràmetres	Tipus de resposta
GET	Authorization: token	Cif (comunitat), string	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Junta	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut trobat les dades"	

Taula 54: EndPoint: GET meeting/cif/string

/meeting			
Descripció	Permet afegir un proveïdor nou a la comunitat		
Mètode	Capçalera	Paràmetres	Tipus de resposta
POST	Authorization: token	-	JSON
Respostes			

Cas	Status	Resposta
Èxit	200	Missatge: "La junta ha quedat registrada"
Error	400	Missatge: "Aquestes dades ja existeixen"
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"

Taula 55: EndPoint: POST Meeting

/meeting/:cif			
Descripció	Permet actualitzar una reforma		
Mètode	Capçalera	Paràmetres	Tipus de resposta
PUT	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "Les teves dades s'han guardat correctament"	
Error	400	Missatge: "Aquestes dades ja existeixen"	
Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut realitzar aquesta operació"	

Taula 56: EndPoint: PUT meeting/cif

/meeting/:cif			
Descripció	Permet esborrar una junta de propietaris		
Mètode	Capçalera	Paràmetres	Tipus de resposta
DELETE	Authorization: token	cif (comunitat)	JSON
Respostes			
Cas	Status	Resposta	
Èxit	200	Missatge: "La junta s'ha esborrat correctament"	

Error	500	Missatge: "Hi ha hagut algun problema i no s'ha pogut esborrar les dades"
-------	-----	---

Taula 57: EndPoint: DELETE meeting/cif

#### 4.3.4. FrontEnd

El desenvolupament del FrontEnd s'ha implementat amb Angular, que és un llenguatge de programació lliure basat en JavaScript. Aquest entorn de programació permet treballar amb components, els quals són reutilitzables en diferents parts de l'aplicació, alhora que gràcies a les directives que fa servir, permet estendre funcionalitats i personalitzar-les en funció de les possibilitats. A banda, treballa internament sobre arquitectura MVC<sup>16</sup> (model, vista, controlador), la qual cosa permet al programador estalviar temps al no haver de crear aquests components per ell mateix.

Altre avantatge que té Angular és que té una llibreria dedicada només als formularis, tant a la seva creació com a la validació de dades, de manera que aquesta aplicació, que fa servir diferents formularis en diferents seccions, es veu afavorida amb un estalvi de temps i línies de codi.

## 4.4. Disseny gràfic

### 4.4.1. Estils

#### Logotip

El logotip que s'ha ideat conté el nom complet de la pàgina per a una identificació senzilla. A banda del logotip principal, s'ha establert una variació en blanc i negre i una icona.

**PROPIETARIS ● CAT**

**PROPIETARIS ● CAT**



Figura 22: Logotips

<sup>16</sup> Arquitectura MVC (Model-Vista-Controlador).

<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>



Els colors corporatius són els següents:



Figura 23: Colors Corporatius

La família tipogràfica del logotip és Saira Stencil One.

Regular 400

**Almost before we knew it, we had left the ground.**

Figura 24: família Tipogràfica Logotip

### Paleta de colors

S'estableix que els colors del logotip han de ser els colors principals de la pàgina per tal de tenir un disseny que reforça la marca corporativa, alhora que els tons suaus proporcionen un disseny net, poc recarregat i agradable a la vista, amb un contrast suau però suficient per no tenir problemes de llegibilitat.

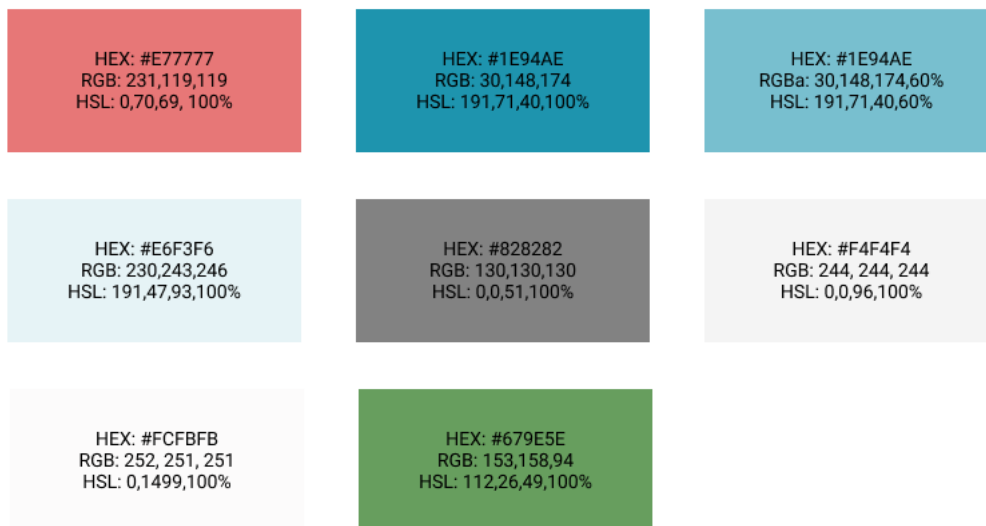


Figura 25: Paleta de colors

## Família Tipogràfica

Tota la pàgina web s'ha fet amb una sola família tipogràfica, *Roboto*. Aquesta decisió està basada en el fet que en la versió d'escriptori, el menú principal fa de títol principal de cada secció, amb la qual cosa s'ha preferit jugar amb la mida de la font per facilitar la llegibilitat.

12px	Lorem ipsum dolor sit amet, consectetur adipiscing elit
14px	Lorem ipsum dolor sit amet, consectetur adipiscing elit
18px	Lorem ipsum dolor sit amet, consectetur adipiscing elit
20px	Lorem ipsum dolor sit amet, consectetur adipiscing elit
24px	Lorem ipsum dolor sit amet, consectetur adipiscing elit

Figura 26: Família Tipogràfica

## Botons

Altre element important de la pàgina web són els botons. S'han dissenyat tres tipus diferents d'aquests i que tenen una mida en funció de la importància i del pes que té la seva funcionalitat dintre del conjunt de la pàgina, com es pot apreciar en la següent imatge.



Figura 27: Estils: Botons

Per a aquelles funcions que tenen a veure amb afegir noves dades dintre de la base de dades, es fa servir el botó 1, amb un color de farciment que reforça la importància de la acció a desenvolupar. El botó 2 té un ús que queda restringit a afegir noves dades (com ara pujar arxius) que prèviament no s'havien introduït en camps existents. Per últim, el botó 3 es fa servir per aquelles accions que tenen a veure amb la edició o eliminació de dades.

### **4.4.2. Usabilitat/UX**

La usabilitat de la pàgina web s'ha realitzat tenint en compte que ha de complir les següents regles d'usabilitat:

#### Visibilitat de l'estat del sistema

L'usuari es qui té el control de l'aplicació i rep contínuament missatges de l'estat de totes les accions que impliquin canvis en el sistema, mostrant en tot moment informació actualitzada. A banda, la pàgina

web es capaç de mostrar on estan els límits del sistema, de manera que l'usuari sap en tot moment que pot fer i que no, fet que implica no transmetre sensació de desorientació.

#### Consistència

El comportament de l'aplicació és constant i previsible, de manera que el mateix tipus d'informació arriba sempre de forma idèntica. Això implica també que totes les tasques similars s'executen respectant sempre la mateixa seqüència d'accions. També implica que la terminologia i els elements gràfics mantenen una constància en tot el sistema.

#### Control de l'usuari

El sistema s'ha d'adaptar sempre a l'usuari i això implica que el disseny ha de ser senzill, intuïtiu i no poden haver-hi accions amb conseqüències imprevistes. Totes les accions s'han de poder desfer i poder tornar a l'estat anterior.

#### Prevenió d'errors

Això implica la implementació de regles de validació en els formularis de manera que, per exemple, en els camps numèrics no es pugui introduir text o viceversa. Els camps de selecció d'ítems es fan servir menús desplegable per ajudar a l'usuari en aquest sentit. Altre forma de prevenció d'errors és oferir opcions de gestió de fitxers on es mostra un llistat de fitxers seleccionables, en comptes d'obligar a l'usuari a teclejar el seu nom.

#### Estructura visible

L'usuari pot veure el menú en tot moment, de manera que li permeti construir-se un mapa de l'estructura interna de la pàgina.

#### Interfície explorable

L'usuari té llibertat de navegació per tota la pàgina i no hi ha camins sense sortida. Les rutes de navegació són molt clares al establir-se un menú visible contínuament i havent-hi una estructura de navegació d'un o dos nivells màxim.

## 5. Implementació

### 5.1. Requisits d'instal·lació

Aquest projecte no necessita de cap instal·lació prèvia ja que es tracta d'una aplicació web, el qual significa que tan sols es necessita d'un dispositiu amb connexió a internet i un navegador com Firefox, Chrome o Safari. D'aquesta manera, l'aplicació es pot trobar en aquest enllaç:

<https://floating-sierra-14629.herokuapp.com/>

#### Instal·lació en local

En el cas de voler fer una instal·lació en local, s'haurà de seguir els següents passos:

1. Instal·lar un servidor Apache que contingui un servidor de base de dades relacional, tipus XAMPP<sup>17</sup>.
2. Inicialitzar el servidor Apache y MySQL.

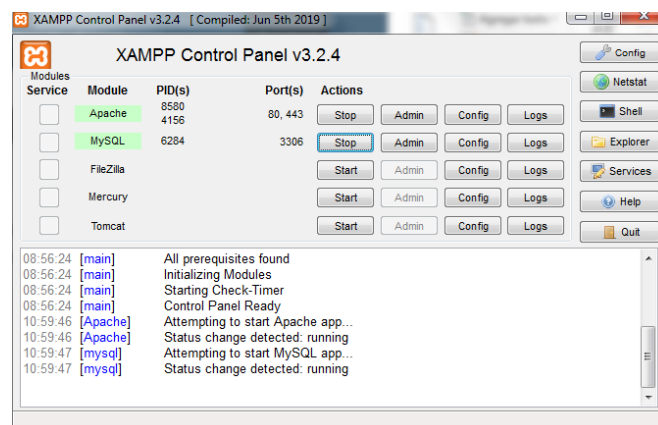


Figura 28: Panell control XAMPP

3. Clicar el botó “Admin” de MySQL per accedir a phpMyAdmin, un gestor de base de dades i un cop allà, afegir una nova base de dades amb el nom “*Propietarispuntcat*”.
4. A la secció “Importa”, importar l'arxiu “*Propietarispuntcat.sql*” inclòs en els arxius lliurats.
5. Guardar en el disc dur les dues carpetes existents dintre de l'arxiu de codi lliurat, “*API\_REST*”, on es troba la part de la API de l'aplicació, i la carpeta “*Propietarispuntcat*” amb el frontEnd.

<sup>17</sup> XAMPP <https://www.apachefriends.org/download.html>

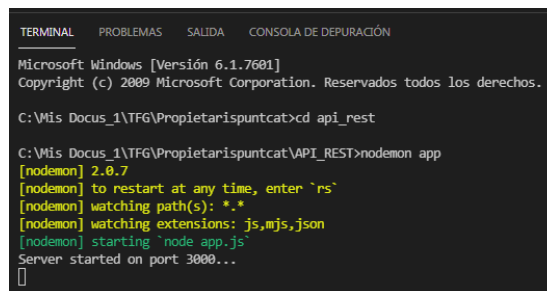
6. Obrir l'aplicació Terminal i anar a la carpeta que conté API\_REST, amb la següent instrucció:

`"cd <ruta carpeta API REST>"`

7. Instal·lar les següents llibreries amb les següents instruccions:

- Bcrypt<sup>18</sup>: **`npm i bcrypt`**
- MySQL<sup>19</sup>: **`npm i mysql`**
- CORS<sup>20</sup>: **`npm i cors`**
- Express<sup>21</sup>: **`npm i express`**
- Express-fileupload<sup>22</sup>: **`npm i express-fileupload`**
- Express-validator<sup>23</sup>: **`npm i express-validator`**
- Jsonwebtoken<sup>24</sup>: **`npm i jsonwebtoken`**
- dotenv<sup>25</sup>: **`npm i dotenv`**
- Nodemon<sup>26</sup>: **`npm i nodemon`**

8. Un cop instal·lades totes aquestes llibreries, executar la ordre "**Nodemon app**" per iniciar la API de l'aplicació al port 3000 del localhost.



```

TERMINAL  PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Wis Docus_1\TFG\Propietarispuntcat>cd api_rest

C:\Wis Docus_1\TFG\Propietarispuntcat\API_REST>nodemon app
[nodemon] 2.0.7
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Server started on port 3000...

```

Figura 29: Posada en funcionament de l'API en local

9. A continuació, obrir una altre finestra del terminal i anar a la carpeta que conté el FrontEnd, executant la següent instrucció:

`"cd <ruta carpeta Propietarispuntcat>"`

10. Instal·lar la següent llibreria:

- file-saver<sup>27</sup>: **`npm i file-saver`**

<sup>18</sup> Bcrypt: <https://www.npmjs.com/package/bcrypt>

<sup>19</sup> MySQL: <https://www.npmjs.com/package/mysql>

<sup>20</sup> CORS: <https://www.npmjs.com/package/cors>

<sup>21</sup> Express: <https://www.npmjs.com/package/express>

<sup>22</sup> Express-fileupload: <https://www.npmjs.com/package/express-fileupload>

<sup>23</sup> Express-validator: <https://www.npmjs.com/package/express-validator>

<sup>24</sup> Jsonwebtoken: <https://www.npmjs.com/package/jsonwebtoken>

<sup>25</sup> Dotenv: <https://www.npmjs.com/package/dotenv>

<sup>26</sup> Nodemon: <https://www.npmjs.com/package/nodemon>

<sup>27</sup> File-saver: <https://www.npmjs.com/package/file-saver>

11. Executar la següent instrucció per iniciar el FrontEnd:

`“ng serve”`

12. Un cop iniciat el projecte, obrir l'enllaç <http://localhost:4200> al navegador per començar a fer servir la aplicació.

## 6. Demostració

### 6.1. Prototips

#### 6.1.1. Prototips Lo-Fi

A continuació, es mostren els primers prototips creats. Aquests prototips estan fets sota el concepte de mobile-first.

En primer lloc trobem la pàgina d'inici de Sessió, on ens podem registrar mitjançant un formulari. Un cop registrats a l'aplicació, accedirem a la pàgina d'inici on podem veure totes les nostres comunitats, i, en el cas que no en tinguem cap, podem registrar una de nova:

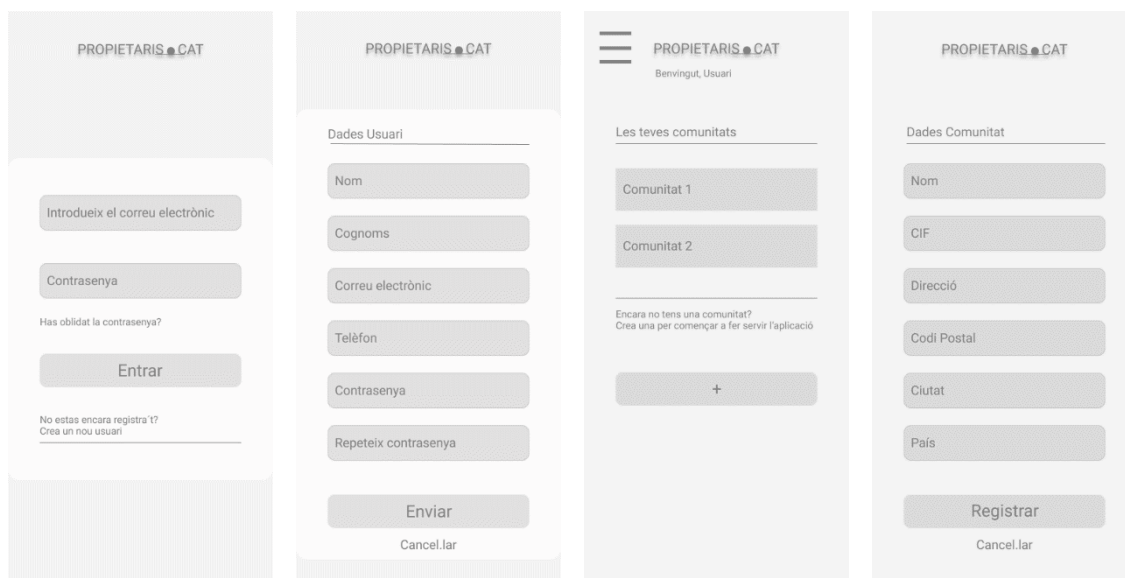


Figura 30: Wireframes: Inici Sessió, Comunitat i registre

Al seleccionar la comunitat que volem veure, accedirem a les dades principals de la comunitat, de la mateixa manera que podem veure quins propietaris formen part de la mateixa, així com els diferents rols que poden exercir. Des d'aquí, podem afegir i/o canviar el president, el secretari i l'administrador de finques. El botó d'Editar envia l'usuari a la pàgina de registre (en funció de les dades que volem editar, ens enviarà a Registre de Comunitat, Usuari, Proveïdor, ...) per canviar les dades corresponents.

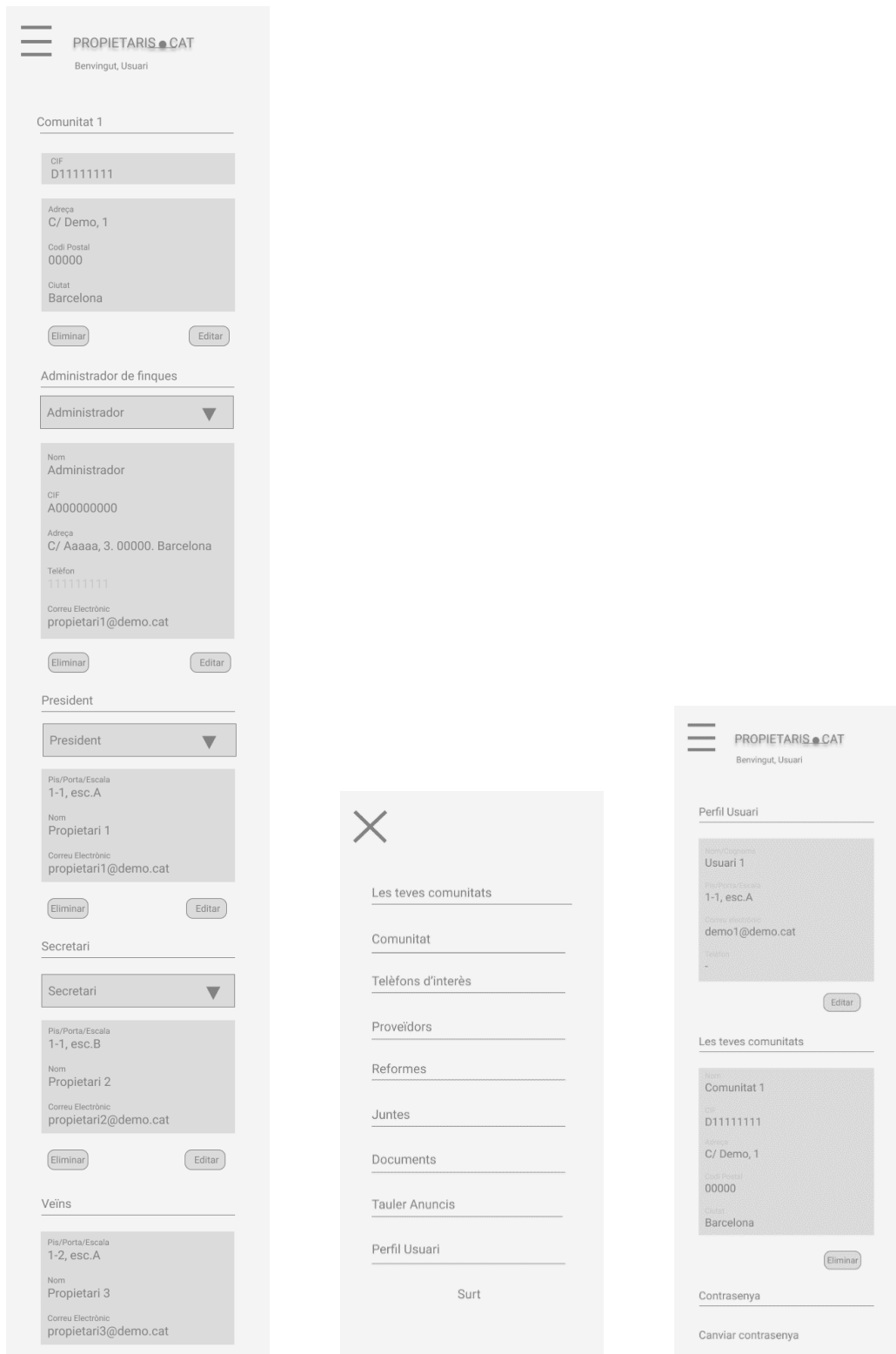


Figura 31: Wireframes: Comunitat, Menú i Perfil Usuari

Com es pot observar, en la versió mòbil podem veure que el menú s'amaga i cal clicar la icona d'hamburguesa per tal de mostrar les diferents opcions de navegació.



En la secció de Telèfons d'interès podrem observar una sèrie de telèfons que són habituals i en podrem afegir a través del botó "+", que ens portarà al formulari de registre d'un nou telèfon.

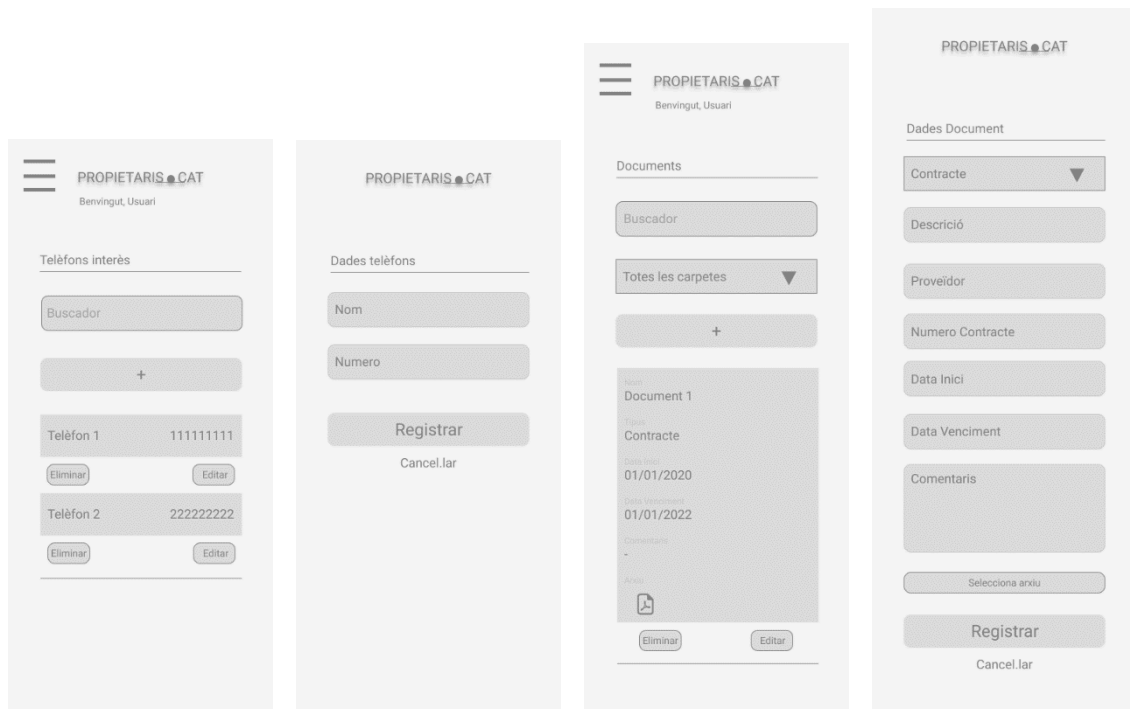


Figura 32: Wireframes: Telèfons interès i Documentació

En la secció de documentació, podem observar un buscador per trobar documents segons el nom. També els podem filtrar segons el tipus de document que volem veure i en podrem afegir de nous mitjançant el formulari corresponent.

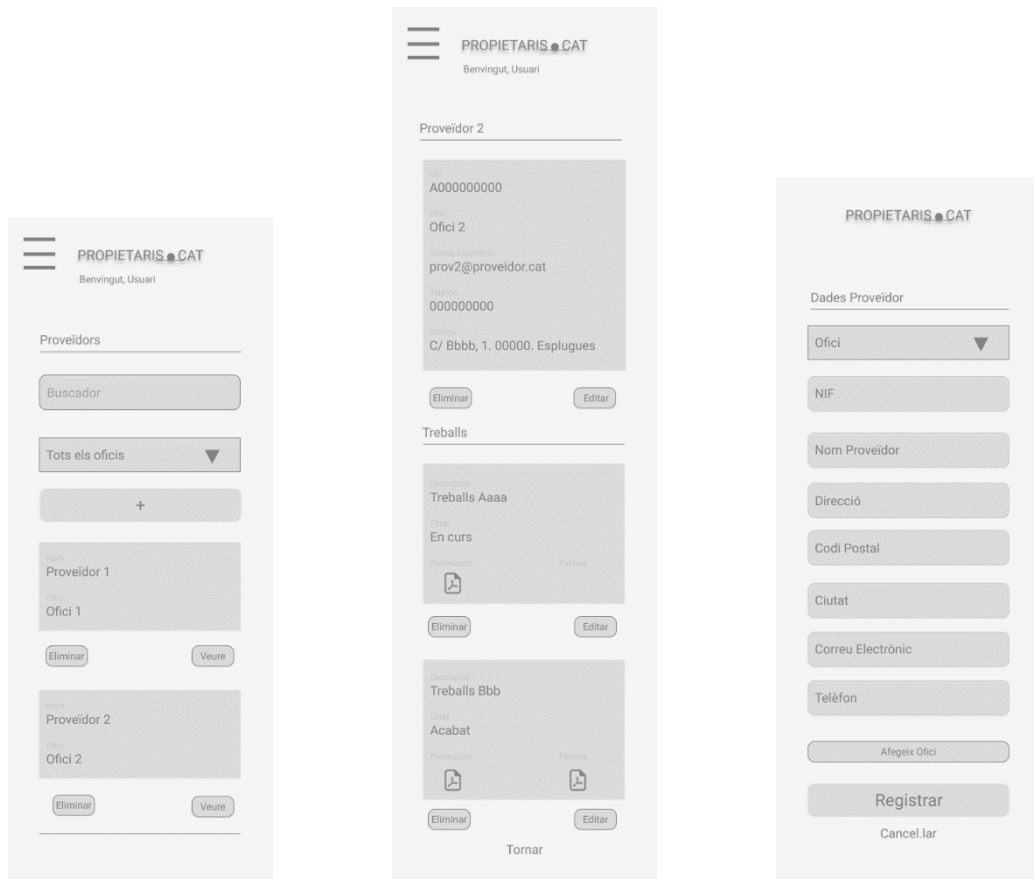


Figura 33: Wireframes: Proveïdors

En la secció de Proveïdors, ens trobem una pàgina similar a la de Documents, amb un buscador i un filtre i des d'aquí podrem accedir a una nova pàgina de detall de cada proveïdor i al formulari de registre d'un de nou. Des de la fitxa de proveïdors, es podran veure els treballs realitzats per cada proveïdor, treballs que es crearan des de la pàgina de reformes.

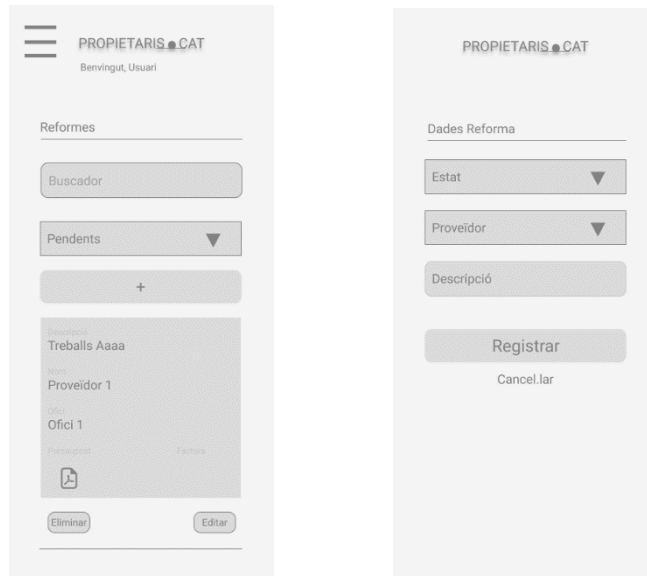


Figura 34: Wireframes: Reformes

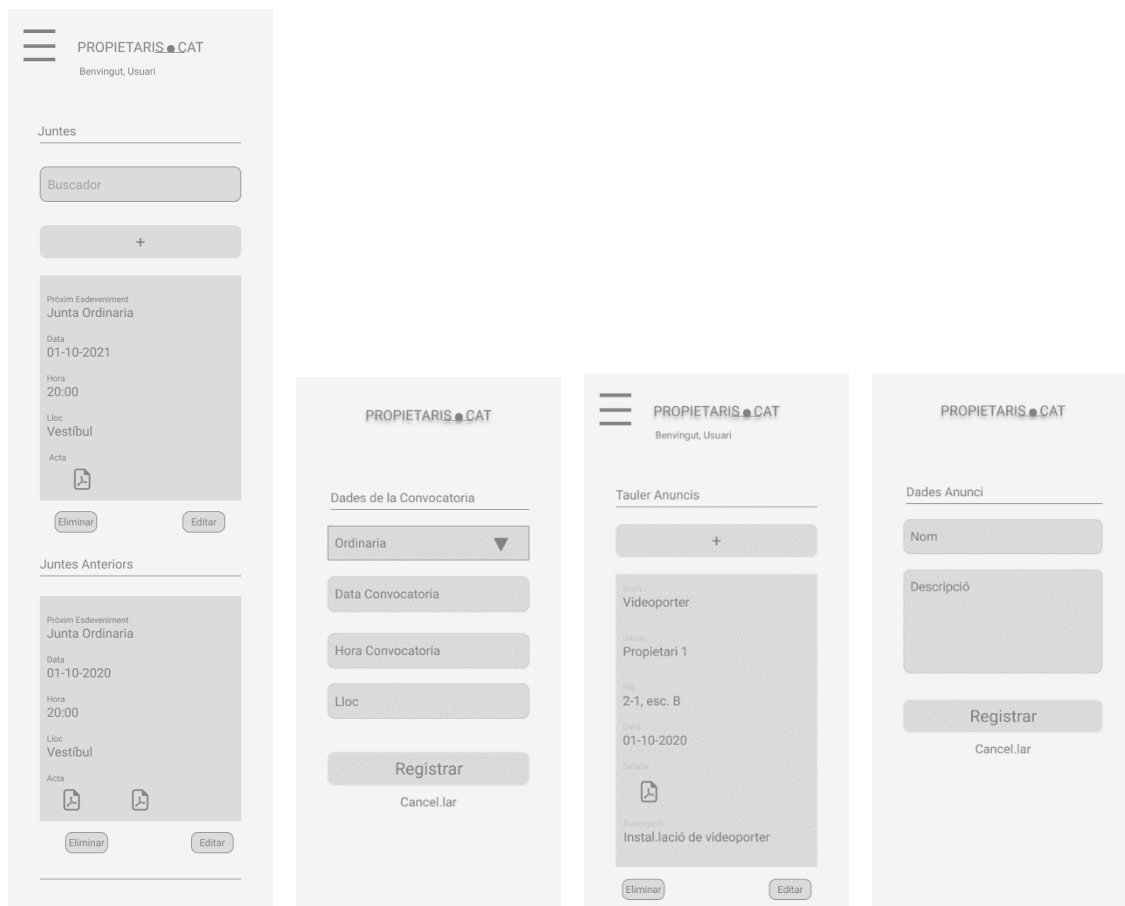


Figura 35: Wireframes: Junes i Tauler d'Anuncis

Les últimes funcionalitats de la web consisteix en veure i afegir Juntes de veïns, així com afegir missatges dirigits a la comunitat a través d'un Tauler d'Anuncis.

## 6.1.2. Prototips Hi-Fi

Un cop realitzat els wireframes, es va desenvolupar un prototip interactiu per tal de veure una aproximació real del que seria el funcionament de l'aplicació.

<https://www.figma.com/proto/2wFu7usuKOIT8DsvJy1xm/TFG?node-id=1%3A3&scaling=scale-down&page-id=0%3A1>

A continuació, es mostren alguns exemples en imatges d'aquests prototips:

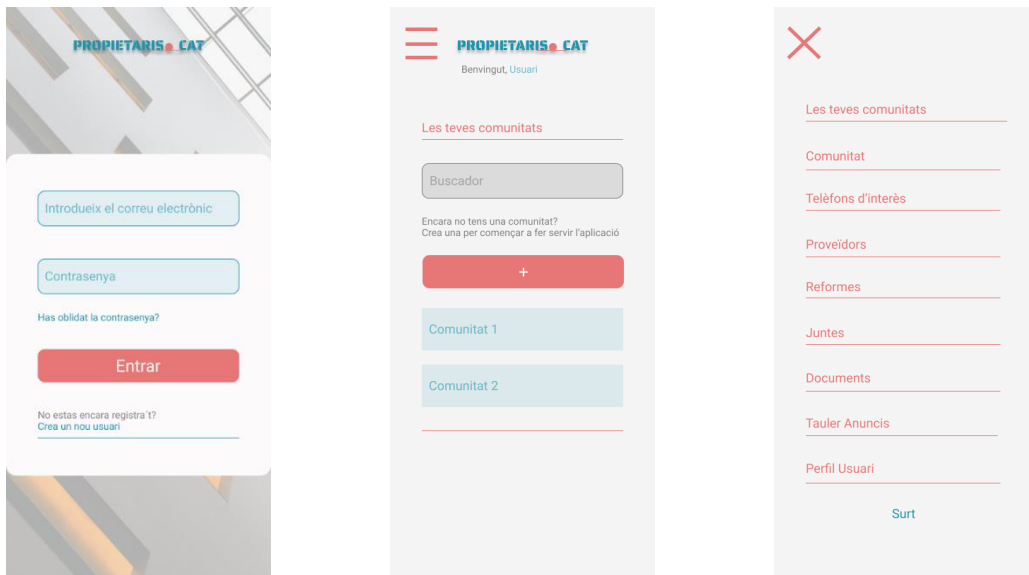


Figura 36: Prototips Hi-Fi (1)

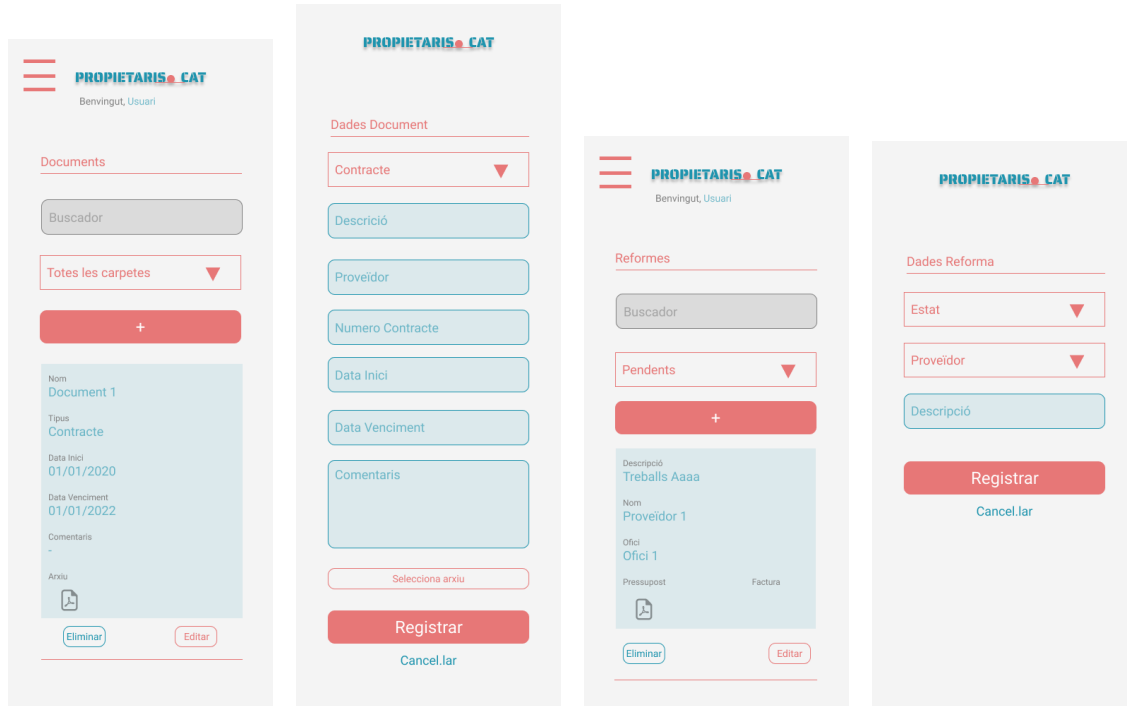


Figura 37: Prototips Hi-Fi (2)

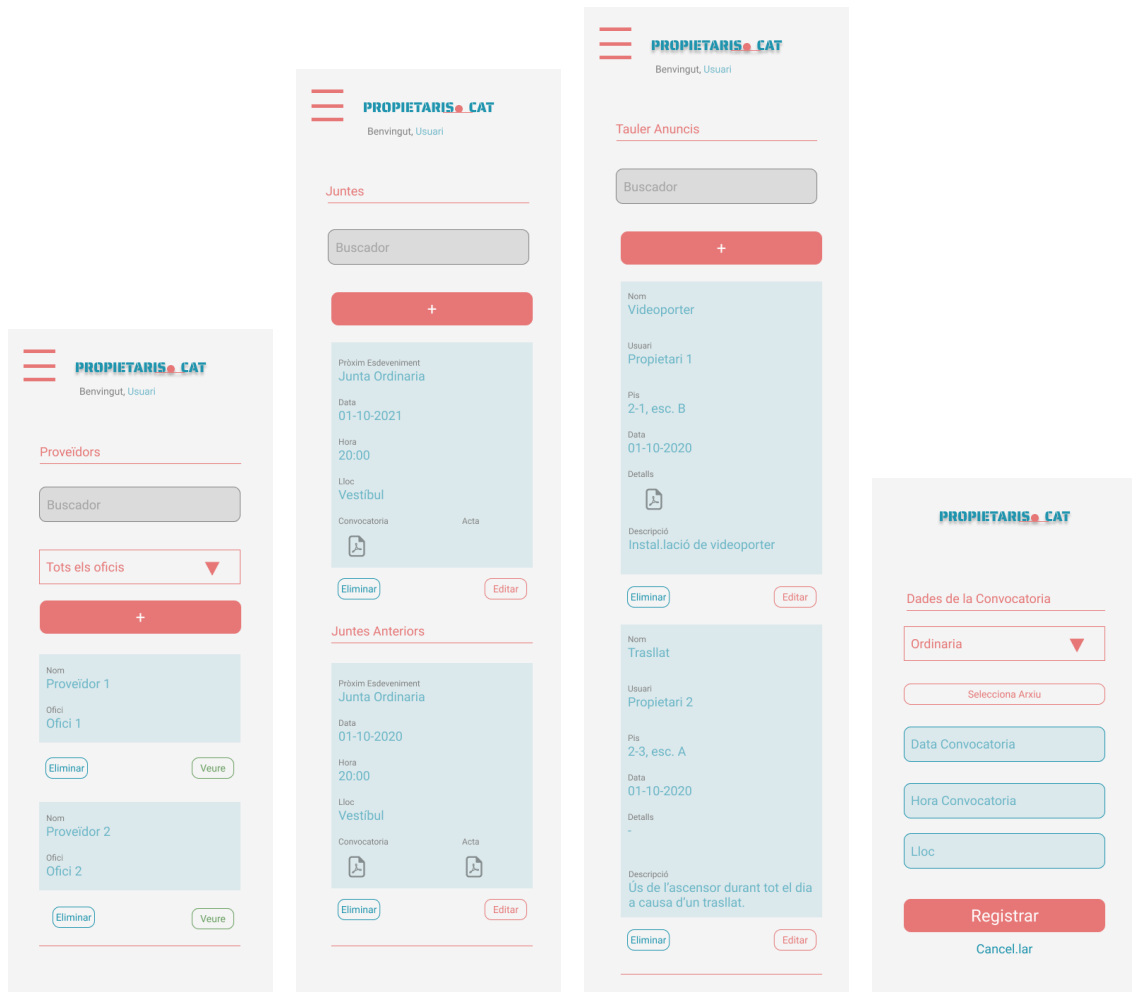


Figura 38: Prototips Hi-Fi (3)

## 7. Conclusions i línies de futur

### 7.1. Conclusions

L'objectiu principal d'aquest projecte era desenvolupar una aplicació web pensada per a la gestió d'una comunitat de propietaris i dirigida pels propis propietaris. Tenint en compte això, aquest projecte aconsegueix satisfactòriament aquest objectiu, doncs, tot i que hi ha pendents algunes millores i coses que m'hagués agradat fer-les d'una altra manera, l'aplicació funciona de manera prou sòlida com per a complir les expectatives inicials.

Sobre la planificació inicial, tot i que s'han acomplert els diferents terminis d'entrega previstos, ha calgut una dedicació força més important a la prevista inicialment en el desenvolupament de la aplicació degut, sobretot, a la gran quantitat de taules i relacions implementades a la base de dades, que en part, ha provocat que a mida que avançava el desenvolupament, hagués d'anar modificant algunes taules per solucionar problemes no detectats inicialment; a la falta d'experiència en la realització d'un projecte d'aquesta mida i a un alt desconeixement dels frameworks utilitzats (Angular i Node), que he hagut d'anar descobrint sobre la marxa.

Des de l'inici de projecte, la meua gran preocupació ha estat precisament la part de desenvolupament degut als motius indicats anteriorment, falta d'experiència en un projecte com aquest, desconeixement dels frameworks utilitzats així com uns terminis que preveia molt justos, com així ha estat. De tota manera, la meua conclusió final és que tot l'esforç ha valgut la pena ja que he pogut adquirir un coneixement que em dona la confiança d'ara en endavant, d'abordar qualsevol projecte d'aquest tipus. Com a contrapartida d'això, un dels propòsits que em vaig fer a l'inici del desenvolupament era de fer un codi el més clar, polit i entenedor que pogués i crec que, tot i que al principi em pensava que anava per bon camí, a mida que ha avançat el projecte, hi ha parts que poden arribar a ser una mica confuses i no hi estic gaire content d'això.

Finalment, com a conclusió personal vull deixar constància que estic molt content de la feina realitzada durant tota la etapa del projecte, ja que calgut un esforç personal important per superar els diferents reptes que s'havien plantejat, així com els diferents dubtes i problemes personals que han anat sorgint al llarg de tot aquest temps.

## 7.2. Línies de futur

El projecte ha assolit tots els objectius previstos inicialment, tot i així, hi ha alguns punts que donen marge de millora en fases posteriors:

- **Implementació mapa Google Maps**

Estava previst inicialment de visualitzar algunes direccions a través d'un mapa de Google Maps en algunes pàgines, però durant la etapa d'implementació s'ha descartat per temes de disseny. La base de dades està preparada ja que s'hi guarden les diferents coordenades.

- **Millora del sistema de cerca de direccions durant el registre de comunitats/proveïdors**

S'ha fet servir la API Geocoding de Google i els resultats que es mostren poden no ser molt intuïtiu.

- **Registre i accés Administrador de Finques (rol Gestor)**

Aquesta és una de les parts que em penso que no ha prou bé i que es podria millorar, per a facilitar el registre i accés per part del Gestor, sense haver de compartir un únic correu electrònic per comunitat.

- **Millorar i netejar el codi**

S'ha intentat de fer un codi clar i entenedor, però a mida que progressava el projecte i amb els terminis d'entrega apropant-se, ha estat més difícil de mantenir net, per la qual cosa, s'hauria de mirar de replantejar algunes parts.

- **Login i contrasenya**

Caldria implementar algun sistema de seguretat extra per realitzar el login i el canvi de contrasenya, com fer servir una verificació de dues passes. També implementar l'accés a l'aplicació a través de Google.

A banda d'aquestes millores, tot i que l'aplicació conté totes aquelles funcionalitats per les quals es va pensar fer aquesta aplicació, al ser una plataforma modular es podria afegir en un futur altres seccions que poguessin resultar interessants a les comunitats de propietaris.



# Bibliografia

1. **¿Cuánto cobra un diseñador web?** [en línea]. Disponible a: <https://wexpertos.com/disenador-web-freelance-precio>. Consultat 28/02/2021
2. **Angular** [en línea]. Disponible a: <https://angular.io/>. Consultat 04/03/2021
3. **BCrypt** [en línea]. Disponible a: <https://www.npmjs.com/package/bcrypt>. Consultat 18/03/2021
4. **Colindar app** [en línea]. Disponible a: <https://colindar.com/>. Consultat 01/03/2021  
Consultat 02/03/2021  
Consultat 04/03/2021
5. **CSS: Cascading Style Sheets** [en línea]. Disponible a: <https://developer.mozilla.org/en-US/docs/Web/CSS>. Consultat 04/03/2021
6. **Desarrollo en cascada** [en línea]. Disponible a: [https://es.wikipedia.org/wiki/Desarrollo\\_en\\_cascada](https://es.wikipedia.org/wiki/Desarrollo_en_cascada). Consultat 26/02/2021
7. **EndPoint** [en línea]. Disponible a: <https://www.paloaltonetworks.com/cyberpedia/what-is-an-endpoint>. Consultat 04/05/2021
8. **Express** [en línea]. Disponible a: <http://expressjs.com/>. Consultat 18/03/2021
9. **Fynkus.es** [en línea]. Disponible a: <https://www.fynkus.es/simulacion-propietario/>.
10. **HTML: HyperText Markup Language** [en línea]. Disponible a: <https://developer.mozilla.org/en-US/docs/Web/HTML>. Consultat 04/03/2021
11. **JSON** [en línea]. Disponible a: <https://en.wikipedia.org/wiki/JSON>. Consultat 18/03/2021
12. **JSON WEB Token** [en línea]. Disponible a: <https://www.npmjs.com/package/jsonwebtoken>. Consultat 18/03/2021
13. **Marco de aplicaciones** [en línea]. Disponible a: [https://es.wikipedia.org/wiki/Marco\\_de\\_aplicaciones](https://es.wikipedia.org/wiki/Marco_de_aplicaciones). Consultat 04/03/2021
14. **Modelo-Vista-Controlador** [en línea]. Disponible a: <https://es.wikipedia.org/wiki/Modelo%20%80%93vista%20%80%93controlador>. Consultat 18/03/2021
15. **NodeJS** [en línea]. Disponible a: <https://nodejs.org/es/>. Consultat 04/03/2021
16. **What is CRUD?** [en línea]. Disponible a: <https://www.codecademy.com/articles/what-is-crud>
17. **What is geocoding?** [en línea]. Disponible a: <https://developers.google.com/maps/documentation/geocoding/overview>. Consultat 18/03/2021

---

# Annexos

## Annex A: Glossari

Glossari de termes i acrònims utilitzats en el treball:

**Angular:** framework de JavaScript, destinat a implementar totes les funcionalitats del FrontEnd.

**BackEnd:** és la part del desenvolupament web que s'encarrega de tota la lògica del sistema. Implementa un conjunt d'accions per atendre les peticions que li arriben del costat del client (usuari), connectant amb la base de dades, si es necessari.

**CRUD:** Acrònim que fa referència a les funcions principals que s'han d'implementar en bases de dades: Create, Read, Update, Delete (Crear, Llegir, Actualitzar i Eliminar).

**CSS (Cascading Style Sheets):** llenguatge per a crear fulles d'estils per a presentar un document escrit en HTML.

**EndPoint:** Tipus de node d'una xarxa de comunicació que queda exposat com a canal de comunicació.

**Framework:** consisteix en un marc d'aplicacions de programari fet servir pels desenvolupadors de software per a implementar la estructura estàndard d'una aplicació.

**FrontEnd:** és la part visible del desenvolupament web que s'encarrega d'implementar la relació que s'estableix amb l'usuari, recollint totes les seves accions.

**HTML (HyperText Markup Language):** llenguatge de marcat bàsic per a construir una pàgina web.

**HTTP (HyperText Transfer Protocol):** Protocol de comunicació web que permet transferència d'informació.

**JavaScript:** llenguatge de programació interpretat. S'utilitza principalment del costa client, implementat com a part d'un navegador web permetent millores en la interfície d'usuari i pàgines web dinàmiques.

**MySQL:** és un servei de base de dades de tipus relacional.

**NodeJS:** entorn d'execució per a JavaScript.

**Stakeholder:** qualsevol individu o organització que són part interessada en la planificació d'una activitat.

**Entitat:** En base de dades, una entitat és la representació d'un objecte o concepte del mon real.

**Primary Key (Clau Primària):** En base de dades, identifica de manera unívoca una fila de cada taula de manera que no hi puguin haver dues files duplicades.

**Foreign Key (Clau Forana):** En base de dades, és l'atribut que relaciona una fila d'una taula amb la clau primària d'un altre.

**Superclasse/Subclasse:** Són entitats que tenen una relació de generalització/especialització, de manera que la superclasse conté les característiques comunes a totes les entitats i les subclasses contenen les específiques.

## Annex B: Lliurables del projecte

Llista d'arxius lliurats i la seva descripció.

- PAC\_FINAL\_guia\_Aparicio\_Jaime.pdf.
- PAC\_FINAL\_mem\_Aparicio\_Jaime.pdf.
- PAC\_FINAL\_prs\_Aparicio\_Jaime.pptx.
- PAC\_FINAL\_vid\_Aparicio\_Jaime.mp4
- Carpeta amb els diferents recursos fets servir a la presentació
- PAC\_FINAL\_prj\_Aparicio\_Jaime.zip, que inclou:
  - Arxiu base de dades: “propietarispuntcat.sql”
  - API REST de la aplicació
  - Carpeta “Propietarispuntcat” amb el codi frontend

## Annex C: Captures de pantalla

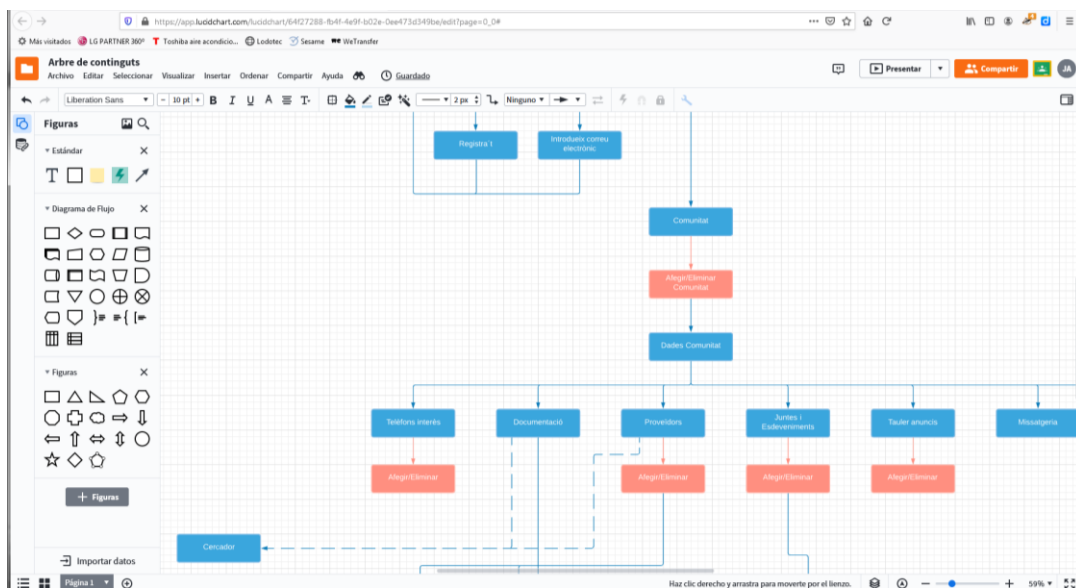


Figura 39: Annex C: Arbre de Navegació



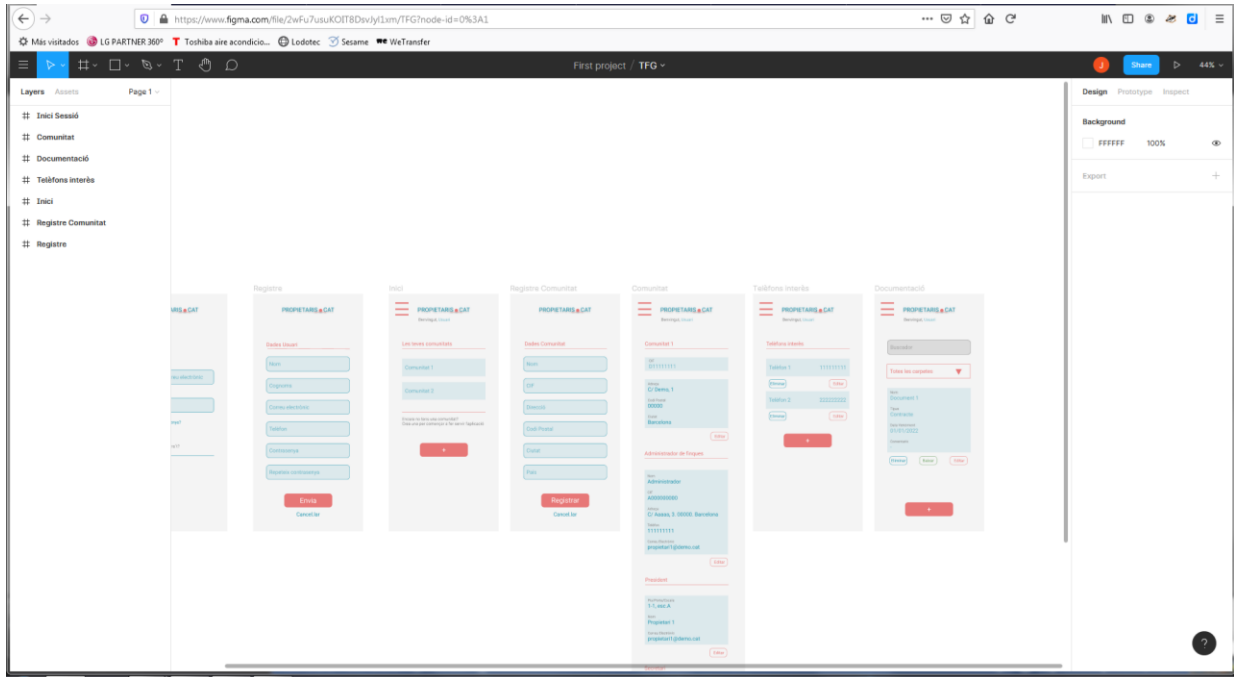


Figura 42: Annex C: Creació Prototips Hi-Fi versió mòbil

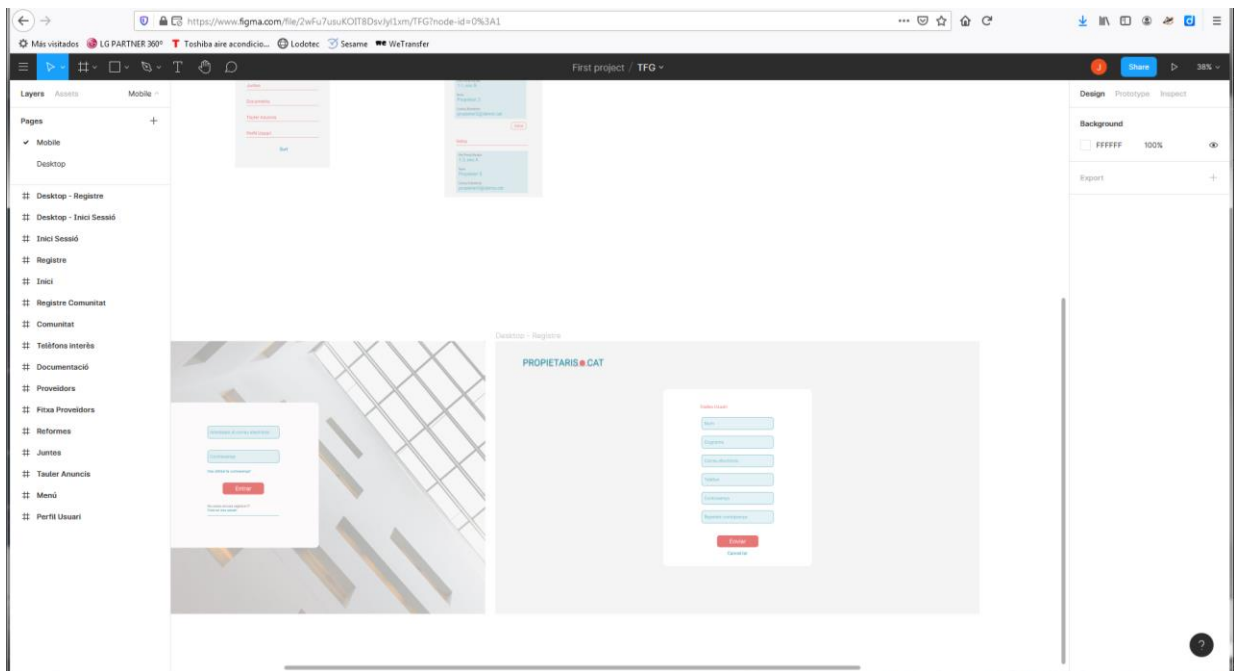


Figura 43: Annex C: Creació Prototips Hi-Fi versió web