

Sistema de referències pel món del lloguer d'habitatges de llarga duració

Bartolomé Roig Mas

Grau d'Enginyeria Informàtica
TFG - Java EE

Nom Consultor/a: Albert Grau Perisé

Nom Professor/a responsable de l'assignatura: Santi Caballe Llobet

9 de gener de 2019



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

*A tu, per la paciència i suport durant totes les hores d'absència.
Aquest petit "èxit" no és meu, sinó nostre.*

Sempre és sempre.

FITXA DEL TREBALL FINAL

Títol del treball:	<i>Sistema de referències pel món del lloguer d'habitatges de llarga duració</i>
Nom de l'autor:	<i>Bartolomé Roig Mas</i>
Nom del consultor/a:	<i>Albert Grau Perisé</i>
Nom del PRA:	<i>Santi Caballe Llobet</i>
Data de lliurament (mm/aaaa):	<i>01/2019</i>
Titulació o programa:	<i>Grau d'Enginyeria Informàtica</i>
Àrea del Treball Final:	<i>TFG – Java EE</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>plataforma, referències, lloguer</i>

Resum del Treball

Totes aquelles persones que han llogat una propietat o bé han viscut de lloguer coneixen les problemàtiques habituals d'aquest tipus d'intercanvi: impagaments, habitatges deixats en mal estat, problemes a l'hora de recuperar la fiança, etc. La majoria d'aquestes situacions deriven de la falta de informació sobre les persones implicades abans de signar el contracte de lloguer.

Aquest projecte neix amb la finalitat de minimitzar aquest problema de desconfiança, plantejant una senzilla solució tant a propietaris com a inquilins: una plataforma web amb un sistema de referències on puguin consultar les experiències de lloguer prèvies d'ambdues parts, contactar amb anteriors propietaris i inquilins per contrastar informació, així com publicar els seus anuncis d'oferta o demanda d'habitatge.

En quant a la metodologia aplicada, abans de començar el projecte s'ha fet una planificació detallada de totes les tasques (plasmada en un diagrama de Gantt). Durant la posterior fase de desenvolupament, he seguit una metodologia àgil, desglossant les fites més grans en petites tasques, amb la finalitat de controlar millor el timing i reduir possibles retards.

Com a resultat, s'ha obtingut una plataforma web totalment funcional on s'han assolit pràcticament totes les funcionalitats plantejades en la fase de planificació.

Desenvolupar un projecte IT des de zero és una tasca complexa, ja que sovint ens solem especialitzar en alguna àrea. El principal objectiu d'aquest TFG era superar les dificultats que això suposa i puc afirmar que l'objectiu s'ha assolit amb èxit i que el camí m'ha servit per aprendre noves tecnologies, bones pràctiques i aptituds.

Abstract

All those who have rented a property or have lived in a rented house know the main problems of this type of exchange: non-payments, houses left in bad conditions, difficulties to get back the deposit, etc. Most of these situations derive from the lack of information about the involved persons before signing the letting agreement.

This project was created to minimize this serious problem of distrust, proposing a simple solution to both landlords and tenants: a web platform with a references system where they can consult previous rental experiences from both parties, to contact previous landlords and tenants to verify the information, as well as to publish their offer or demand rental ads.

Regarding the applied methodology, before starting the project, I made a detailed Planning of all the tasks (expressed in a Gantt Diagram). Then, during the development phase, I have followed an agile methodology, splitting the biggest milestones into smaller tasks, in order to be able to control the timing and reducing possible delays.

As a result, we obtain a totally functional web platform, where all the features planned in the initial planning have been achieved.

Developing an IT project from scratch is a very complex task, as we often tend to specialize in some of the areas. The main goal of this TFG was to overcome the difficulties that this implies, and I can say the objective has been successfully achieved and the way has helped me to learn new technologies, good practices and aptitudes.

Índex

1. Introducció.....	1
1.1 Context i justificació del Treball	1
1.2 Objectius del Treball.....	1
1.3 Enfocament i mètode seguit	2
1.4 Planificació del Treball.....	3
1.5 Breu sumari de productes obtinguts	3
1.6 Breu descripció dels altres capítols de la memòria	5
2. Requeriments d'alt nivell	7
2.1. Sistemes transversals.....	7
2.2. Plataforma web	7
2.3. API de consulta	9
3. Arquitectura	10
3.1. Diagrama.....	10
3.2. Tecnologia i frameworks utilitzats.....	11
3.3. Eines de desenvolupament	12
4. Anàlisi.....	13
4.1. Actors	13
4.2. Diagrama de casos d'ús	14
4.3. Fitxes de casos d'ús	16
5. Prototipat.....	35
5.1. Registre i autenticació	35
5.2. Gestió d'anuncis.....	37
5.3. Gestió de referències i reputació.....	38
5.4. Gestió de contactes.....	39
5.5. Sistema de missatgeria	40
5.6. Gestió de perfil d'usuari.....	40
5.7. Motor de disponibilitat.....	41
6. Disseny.....	43
6.1. Disseny de la bases de dades de grafs.....	43
6.2. Diagrama de classes	46
6.3. Arquitectura.....	49
7. Implementació	52
7.1 Projectes implementats	52
7.2 Plataforma web	57
7.3 Webservice de consulta de referències.....	62
7. Conclusions.....	64
8. Glossari	66
9. Bibliografia.....	68
10. Annexos	70
10.1 Guia de instal·lació a Amazon EC2.....	70
10.2 Guia de configuració Amazon S3	73
10.3 Configuració projecte en local	77

1. Introducció

1.1 Context i justificació del Treball

Totes aquelles persones que han llogat una propietat o bé han viscut de lloguer en algun moment de la seva vida coneixen les problemàtiques habituals d'aquests tipus d'intercanvi: impagaments, habitatges deixats o trobats en mal estat, problemes amb els veïnats i a l'hora de recuperar la fiança, etc. La majoria d'aquestes situacions solen derivar de la falta de informació sobre les persones implicades, prèvia a la signatura del contracte de lloguer.

Abans de prendre la decisió de llogar una propietat per una llarga temporada, a qualsevol inquilí li sorgeixen dubtes que difícilment podria resoldre fins que ja està instal·lat en la casa o pis:

- Trobaré l'habitatge en les condicions contractades?
- M'ajudarà el propietari si tinc cap problema o avaria en la casa?
- Tindrè problemes per recuperar la fiança un cop acabat el període de lloguer?

Des de el punt de vista del propietari, la incertesa respecte l'inquilí és encara major:

- Pagarà les mensualitats amb puntualitat?
- Deixarà la propietat en bones condicions?
- Serà respectuós amb els veïnats?

En definitiva, existeix entre propietaris i potencials inquilins un important problema de desconfiança, a dia d'avui agreujat a més per altres factors com el lloguer turístic (menys riscos a la hora de llogar), l'ocupació il·legal de pisos, etc.

Amb la finalitat de minimitzar el problema de desconfiança indicat, aquest projecte planteja una senzilla solució tant a propietaris com a inquilins: un sistema de referències online on es puguin consultar les experiències de lloguer prèvies d'ambdues parts, oferint als usuaris la possibilitat de contactar amb anteriors propietaris/inquilins per ampliar i contrastar informació.

1.2 Objectius del Treball

Desenvolupar un projecte IT totalment des de zero, participant activament en totes les seves fases: anàlisi, disseny i implementació (i en aquesta darrera fase, en totes les seves capes: dades, lògica de negoci i part visual) és una tasca que a més de ser complicada, difícilment podem dur a terme en la nostra vida professional, ja que sovint ens especialitzem en algunes d'aquestes parts.

És per aquest motiu que he enfocat el TFG cap al desenvolupament d'un projecte complet que perfectament podria tenir sortida en el món real. La meua experiència professional en JAVA probablement ha decantat la balança cap a l'àrea JAVA EE enfront altres àrees. No obstant, he vist en el TFG la oportunitat d'aprendre noves tecnologies, com ara les bases de dades de grafs, frameworks

web actuals com angular, etc. així com de profunditzar en tecnologies amb les quals ja tinc certa experiència.

Així, els principals objectius a nivell general són:

- Entendre i superar les dificultats que suposa desenvolupar un projecte de software des de l'inici.
- Valorar la importància que té actualment el software open-source en el desenvolupament de projectes informàtics, inclús a nivell empresarial.
- Profunditzar en tecnologies amb les quals personalment ja tinc experiència, com JAVA EE.
- Aprendre noves tecnologies, com ara "Neo4j", millorant així la capacitat d'adaptació a entorns que a nivell tecnològic canvien molt ràpidament.
- Millorar la meua capacitat d'orientació a l'usuari: disseny d'interfícies, usabilitat, etc.
- Familiaritzar-me amb algun framework web javascript modern i actual (en aquest aspecte considero que JSF ha quedat una mica obsolet enfront tecnologies com "React", "Angular", etc).
- Aplicar patrons de disseny en una aplicació real.

1.3 Enfocament i mètode seguit

En quant a la metodologia aplicada, abans de començar el projecte s'ha fet una planificació detallada de totes les tasques (plasmada en un diagrama de Gantt) i s'ha consensuat amb el consultor. Durant les posteriors fase de disseny, desenvolupament i desplegament, he seguit una metodologia àgil, desglossant les fites més grans en petites tasques, amb la finalitat de controlar millor el timing i reduir possibles retards.

A mode resum, vaig dividir les tasques en tres grans blocs: "To do", "In process" i "Done". Inicialment, totes les tasques del Planning estaven a la llista "To do", juntament amb una descripció detallada i una valoració aproximada de dies de feina estimats. En el moment en que abordava la tasca, aquesta passava a la llista "In Process" i un cop acabada a "Done". Ha estat clau intentar tenir com a màxim dues o tres tasques simultànies a la llista "In process", a fi de poder focalitzar i intentar ajustar el temps dedicat a la planificació estimada inicialment.

Per aplicar aquesta metodologia àgil, he fet ús de l'eina Trello, la qual proporciona una útil interfície visual a través de la qual es pot veure l'estat general de les tasques amb un simple cop d'ull.

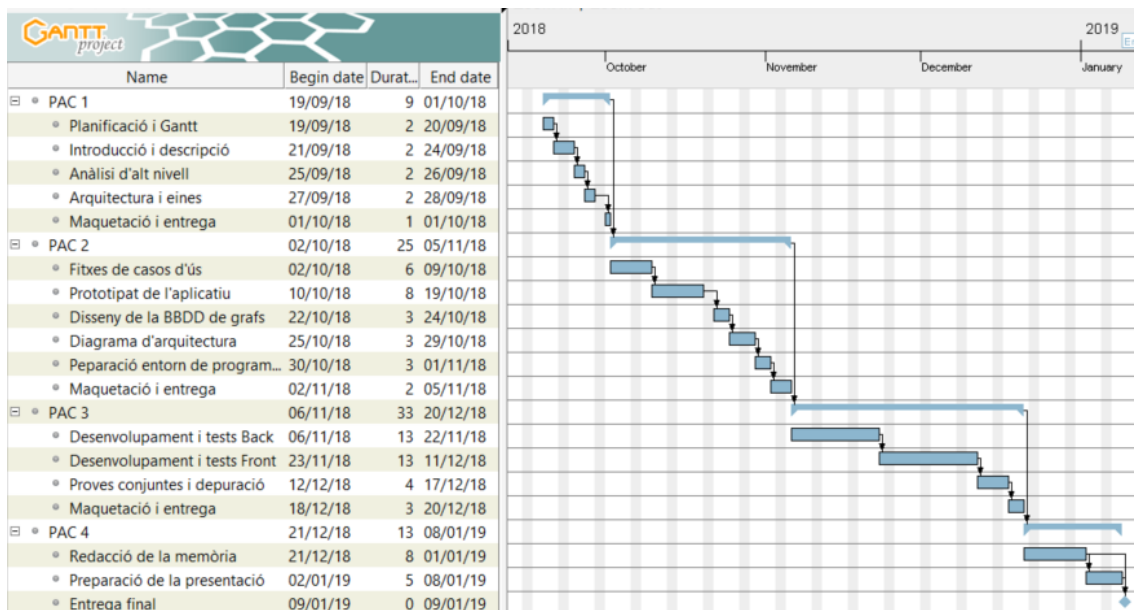
Una vegada obtingut el producte final, on s'han pogut desenvolupar satisfactòriament i dins les dates previstes pràcticament totes les funcionalitats proposades en el planning inicial, puc concloure que la metodologia seguida és adequada per aquest tipus de projecte i

1.4 Planificació del Treball

1.4.1 Dates clau

PAC 1	Pla de treball, especificació dels requeriments d'alt nivell del projecte, així com de les tecnologies, frameworks i eines utilitzats.	01/10/2018
PAC 2	Anàlisi i disseny detallat que inclou els models de casos d'ús i actors, les fitxes dels casos d'ús més rellevants. En quant a l'arquitectura: definició del diagrama de classes així com el model de grafs de la base de dades. Prototipat bàsic de l'aplicatiu.	05/11/2018
PAC 3	Implementació i tests de l'aplicatiu.	20/12/2018
PAC 4	Redacció de la memòria així com entrega del producte final desenvolupat.	09/01/2019

1.4.2 Diagrama de Gantt



1.5 Breu sumari de productes obtinguts

El producte obtingut és una plataforma web “Responsive” totalment funcional centrada en el món del lloguer d'habitatges de llarga duració, la qual, a través d'un sistema reputacional, intenta minimitzar els problemes de desconfiança existents entre propietaris i inquilins descrit en el primer apartat.

Tot usuari registrat en la plataforma web pot accedir a un sistema de referències online i escriure una referència a un inquilí o propietari explicant com ha estat la seva experiència de lloguer amb aquesta persona. Així, aquestes referències podran ajudar a altres usuaris a l'hora de prendre la decisió de si llogar o no una propietat.

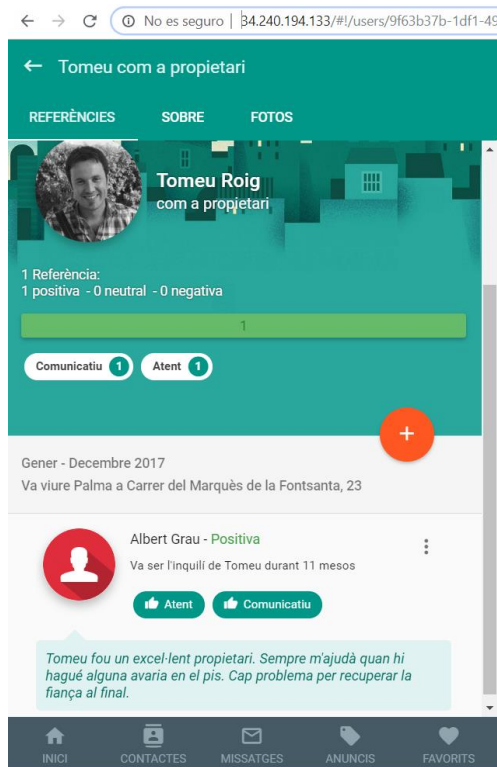


Figura 1. Vista referències d'un usuari propietari

A més, la plataforma ofereix a futurs propietaris / inquilins la possibilitat de contactar directament amb altres usuaris (a través d'un servei de missatgeria privat), per tal d'ampliar informació, concertar visites als immobles, contrastar i verificar referències, etc.

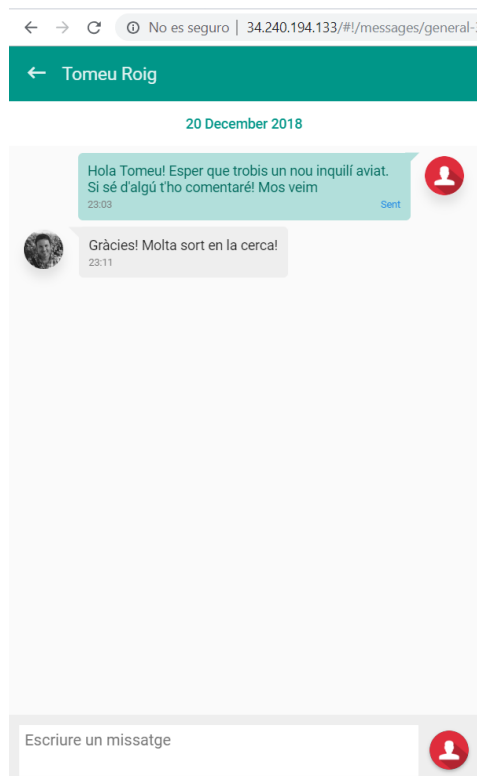


Figura 2. Vista conversa privada entre dos usuaris

L'aplicatiu permet crear, publicar i gestionar anuncis d'oferta o demanda d'habitatges de lloguer. A més, els usuaris de la plataforma tenen accés a un senzill cercador on poden trobar els anuncis disponibles en una determinada zona i filtrar la resposta en base a les seves preferències.

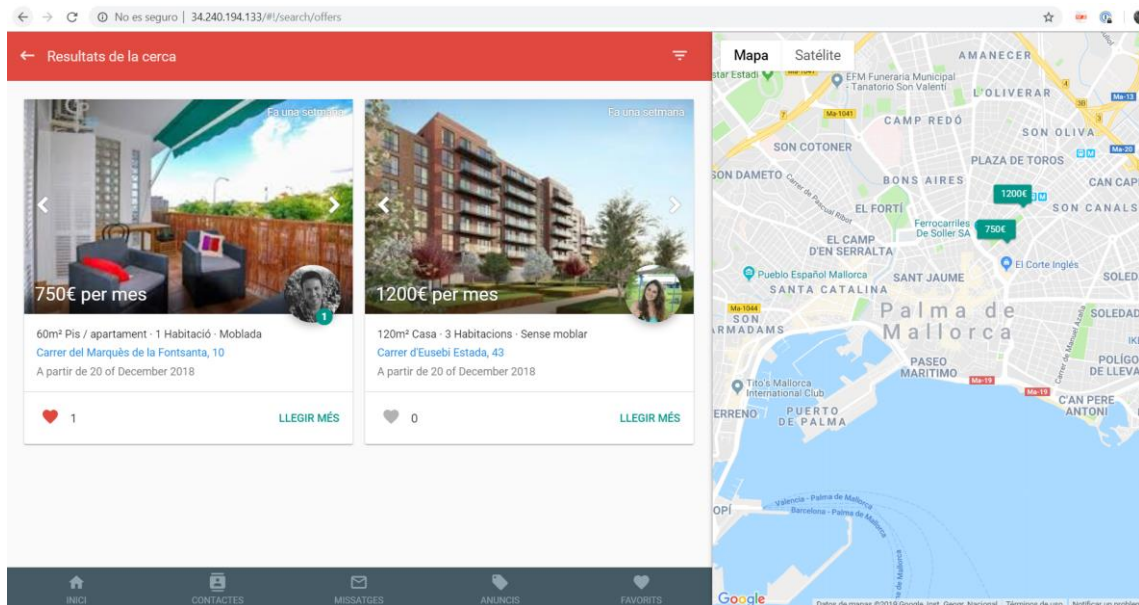


Figura 3. Vista amb el resultat d'una cerca d'anuncis d'oferta a Palma de Mallorca

A part de la plataforma web, el projecte inclou un petit webservice o API per tal d'oferir a tercers la possibilitat de consultar les referències que té un usuari en la plataforma. Aquesta informació pot ser útil, per exemple, per altres empreses del sector immobiliari. Al igual que la web, tota petició enviada al webservice requereix credencials d'autenticació.

Tot i que per falta de temps no està a l'abast d'aquest TFG, el fet de que la plataforma web sigui totalment responsive deixa la porta oberta a que en un futur es pugui empaquetar amb algun framework tipus "Cordova" i publicar-la com aplicació Android o IOs. Així mateix, en un futur es podria estendre el projecte amb funcionalitats tals com:

- Recepció de notifikacions per mail: per exemple, al rebre una referència o un missatge.
- Suggeriment automàtic de possibles bons inquilins.
- Poder crear alarmes personalitzades en el cercador d'anuncis.
- Signatura digital de contractes de lloguer entre propietari / inquilí

1.6 Breu descripció dels altres capítols de la memòria

Els següents capítols de la memòria (2 – 6) descriuen en detall totes les fases del desenvolupament del projecte en la seva totalitat. En resum:

- Capítol 2: Anàlisi d'alt nivell, especificant els principals requeriments i els diferents subsistemes transversals de l'aplicatiu.
- Capítol 3: Arquitectura d'alt nivell. Descripció de l'stack tecnològic, així com de les principals eines i frameworks utilitzats.

- Capítol 4: Anàlisi detallat. Identificació dels diferents actors i casos d'ús.
- Capítol 5: Prototipat de tota la interfície d'usuari de la plataforma web.
- Capítol 6: Disseny, incloent els diagrames de capes, model de dades, classes i arquitectura (ja més a baix nivell)
- Capítol 7: Implementació. Descripció del desenvolupament dels diferents mòduls del projecte així fins arribar al producte final.

En el capítol 8 s'inclouen les conclusions mentre que els capítols 9 i 10 fan referència al glossari i bibliografia utilitzada en la memòria.

Ja per acabar, en el darrer capítol, annexes, he inclòs:

- Una guia de instal·lació a Amazon EC2, servidor cloud on he desplegat l'aplicatiu.
- Una guia de configuració de Amazon S3, servei que he emprat per emmagatzemar en el cloud els recursos estàtics com les fotografies de perfil i les associades als anuncis de lloguer.
- Instruccions per poder arrancar en local els dos mòduls de l'aplicatiu: tfg-back (backoffice amb la lògica de negoci) i tfg-front (client web amb tota la interfície d'usuari)

2. Requeriments d'alt nivell

A continuació enumerem els principals subsistemes o serveis de l'aplicatiu, així com els requeriments funcionals més rellevants de cadascun d'ells. Tal com s'ha indicat en la descripció del projecte, l'aplicatiu consta de dos punts d'entrada: la plataforma web principal i la API bàsica de consulta de referències orientada a tercers. No obstant, poden existir certs serveis transversals a ambdós mòduls.

2.1. Subsistemes transversals

En aquest apartat s'inclouen els serveis transversals utilitzats tant en la plataforma web com en la API de consulta.

2.1.1. Gestió d'usuaris i autenticació

Com ja hem comentat, tota interacció amb l'aplicatiu, tant a través de la plataforma web com de l'API de consulta, requerirà que l'usuari s'hagi registrat en el sistema. El subsistema de gestió d'usuaris inclou dues funcionalitats bàsiques:

- Registre d'usuari: Donar d'altra l'usuari per primer cop en l'aplicatiu a través d'un formulari de registre.
- Login: Autenticació en el sistema a través del mail i la contrasenya proporcionats en el registre.

Pensant en la futura escalabilitat del sistema, és molt important que l'autenticació sigui "State-less", de tal forma que tota la informació que requereixi l'usuari per interactuar amb l'aplicatiu viatgi en el token proporcionat en l'autenticació

2.2. Plataforma web

És a través de la plataforma web des d'on els usuaris registrats poden gestionar les diferents entitats de l'aplicatiu.

2.2.1. Gestió d'anuncis

Aquest subsistema engloba tota la gestió d'anuncis tant d'oferta com de demanda d'habitatge: alta, baixa, modificació i consulta (l·listat i detallada). Cal fer especial menció a la possibilitat de desactivar un anunci, de manera que deixarà d'estar disponible sense necessitat d'eliminar-lo.

A mode resum, un anunci d'oferta d'habitatge consta de:

- Dades bàsiques de la propietat: tipus, estat, nombre d'habitacions i banys, metres quadrats, serveis proporcionats (wifi, cuina, calefacció..), ubicació, etc.
- Descripció de l'anunci: Títol, descripció i fotografies.
- Condicions i restriccions: Preu, estància mínima i màxima, restriccions a certs elements com animals de companyia, fumadors, etc.

Per altra banda, un anunci de demanda d'habitatge, molt més senzill, requereix:

- Tipus d'habitatge, ubicació i serveis opcionals desitjats.
- Descripció textual explicant amb més detall que s'està cercant.
- Pressupost i condicions desitjades: Per exemple, màxim 750 € mensuals i ha de permetre animals de companyia.

2.2.2. Gestió de referències

Un usuari de l'aplicatiu ha de poder deixar una referència a un altre usuari, explicant com ha estat la seva experiència de lloguer amb aquesta persona. Les referències poden ser en el sentit Propietari → Inquilí i viceversa: Inquilí → Propietari.

Una referència s'encapsula sota el marc d'una estància de lloguer: rang de dates i una adreça determinada. Per exemple: Entre gener i desembre de 2017, per la propietat ubicada en Avinguda Diagonal 44 de Can Picafort. Així, bàsicament consta de:

- Autor: L'usuari que escriu la referència (propietari o inquilí)
- Destinatarí. L'usuari que la rep (propietari o inquilí)
- Estància a la qual fa referència
- Detall de l'experiència de lloguer: positiva o negativa, text descriptiu i en funció de si la referència és cap a un propietari o cap a un inquilí, certs qualitats a destacar d'aquesta persona. Per exemple: Pagador puntual, respectuós amb els veïns, etc. (si la referència és d'un propietari cap a un inquilí) o comunicatiu, prest a ajudar, etc (en cas contrari).

L'entitat de referència té una casuística un tant especial: a fi de garantir-ne la fiabilitat, no es permetrà esborrar-les un cop escrites (seria interessant implementar una funcionalitat de denunciar referència, per aquelles que puguin resultar ofensives o spam, encara que no entra dins l'àmbit d'aquest TFG). S'han de poder consultar les referències que ha rebut un usuari així com les que l'usuari registrat ha escrit a altres usuaris.

2.2.3. Gestió de contactes

Un usuari ha de poder cercar altres usuaris donats d'alta dins la plataforma i visualitzar-ne la informació. Opcionalment, pot guardar l'usuari com a contacte, amb la finalitat de facilitar una futura cerca. No es tracta d'una relació del tipus xarxa social, sinó més bé com una simple agenda de contactes. Així, aquest subsistema s'encarrega de mantenir aquesta agenda, permetent:

- Alta i baixa de contactes l'agenda.
- Llistat dels contactes de l'agenda.
- Cerca de contactes dins l'agenda.

2.2.4. Sistema de missatgeria

Uns dels punts claus de l'aplicatiu és la possibilitat de que els usuaris puguin contactar entre ells a través d'un sistema de missatgeria privat, bé per poder concertar una visita a un immoble, per contrastar informació d'una referència, etc. Les funcionalitats principals d'aquest subsistema són:

- Llistat de converses de l'usuari.
- Possibilitat d'enviar i rebre missatges, tant si l'usuari està offline com en temps real si tant emissor com receptor estan connectats.

2.2.5. Gestió del perfil d'usuari

Donat el caire social de l'aplicatiu, els usuaris poden gestionar un perfil bàsic, on inclouen una mica més de informació d'ells mateixos. Aquest perfil podrà ser consultat per qualsevol usuari registrat dins la plataforma. L'edició d'un perfil únicament la podrà dur a terme el propi usuari. A mode resum, consta de:

- Informació personal bàsica: Nom, edat, ciutat, lloc de residència.
- Informació acadèmica i professional: estudis, professió, etc.
- Descripció personal.
- Fotografies de perfil i de portada i altres fotografies addicionals complementàries.

2.2.6. Motor de disponibilitat

Aquest subsistema gestiona el cercador d'anuncis de la plataforma, així com el posterior filtratge opcional de resultats de la resposta. Només es permetrà cercar per tipus d'anunci (oferta o demanda) i zona geogràfica, encara que es podran acotar els resultats de la resposta en base a filtres opcionals com: mínim d'habitacions, preu, etc.

En el cas del cercador d'anuncis d'oferta d'habitatge, serà interessant poder visualitzar els resultats en format llista o geo-posicionats damunt un mapa.

Un cop visualitzada la llista d'anuncis disponibles, s'ha de poder accedir a la informació detallada d'un anunci concret i des d'aquí, poder contactar directament amb l'anunciant a través del servei de missatgeria.

Un procés de cerca optimitzat serà clau per garantir que l'usuari obté la resposta a la seva consulta en un temps adequat.

2.3. API de consulta

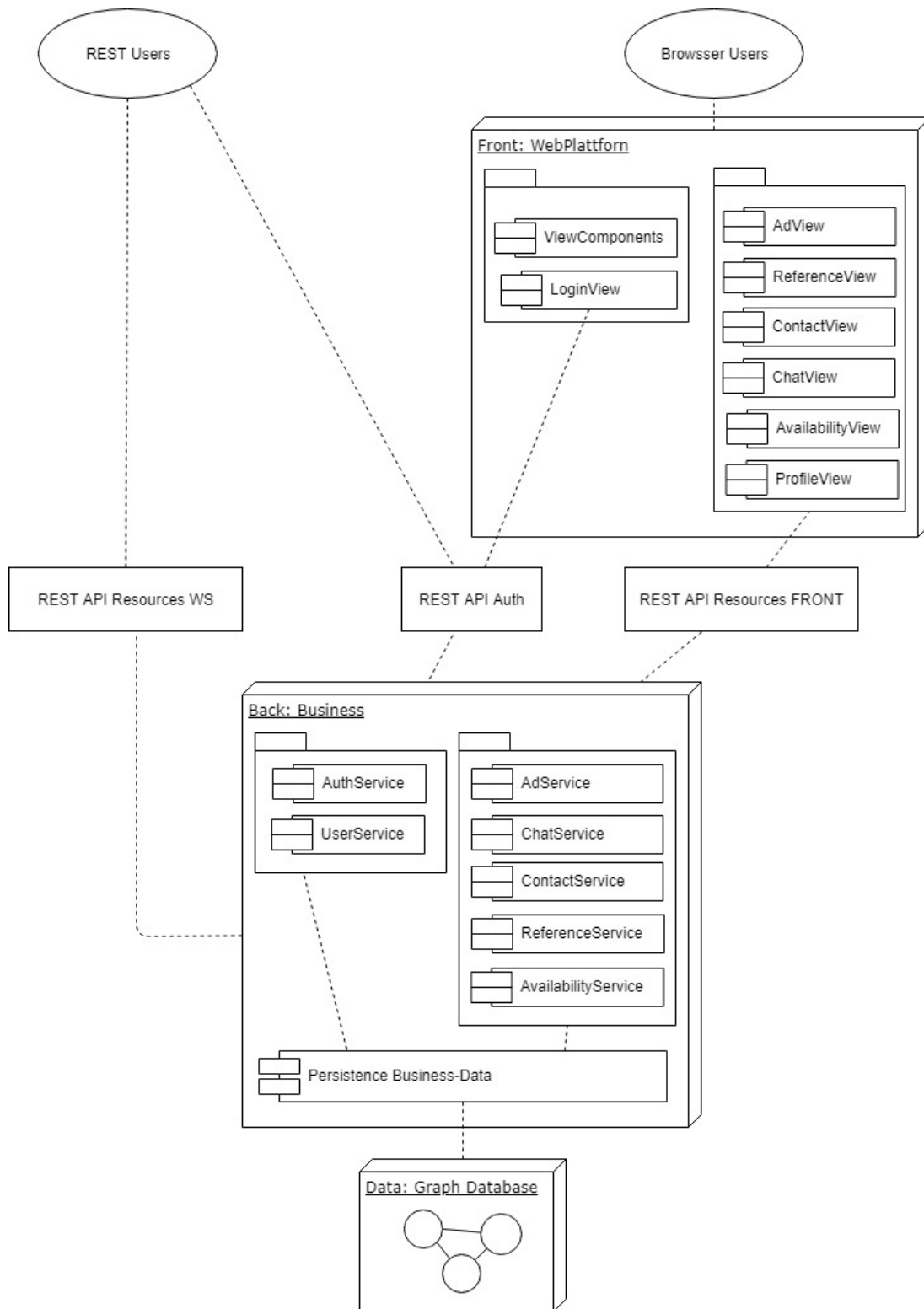
Webservice o API bàsica que ofereix a tercers la possibilitat de consultar les referències que té en la plataforma un determinat usuari.

2.3.1. Consulta de referències

La informació més valuosa i diferenciadora de la plataforma són les referències. Amb la finalitat de poder oferir aquesta informació a tercers (pot ser útil per altres empreses del sector immobiliari), es publicarà una API "Read-only" que permetrà consultar les referències d'un determinat usuari. L'API requerirà credencials d'autenticació.

3. Arquitectura

3.1. Diagrama



3.2. Tecnologia i frameworks utilitzats

3.2.1. Back

Tot el back es desenvoluparà íntegrament amb Java 1.8, sobre l'ecosistema "Spring". Tota la capa de negoci s'organitza en una arquitectura orientada a microserveis, sempre pensant en els beneficis que aporta la modularitat d'aquesta arquitectura així com la facilitat d'escalat. Principals frameworks utilitzats:

- **Spring:** El framework és molt complex i conté una quantitat de mòduls considerable. Els més rellevants que utilitzem en el projecte són:
 - Spring-boot: Per l'empaquetat i execució.
 - Spring-security: Per gestionar l'autenticació, a través del protocol "OAuth2"
 - Spring-data: Persistència amb la base de dades. Més concretament, del projecte spring-data fem spring-data-neo4j, mòdul específic per la base de dades de grafs Neo4j.
 - Spring-mvc: Utilitzat per exposar les diferents API REST de l'aplicatiu.
- **Apache Maven:** Per la gestió de dependències necessàries en l'aplicació, així com el posterior empaquetat i desplegament.
- **Neo4j:** Base de dades de grafs implementada en JAVA que destaca pel seu rendiment i orientació a relacions, adient per a molts casos d'ús del projecte.

3.2.2. Front

El client web es desenvoluparà per complet amb el framework Angular. A continuació destaquem algunes de les llibreries i mòduls externs utilitzats, tots de codi obert, més rellevants:

- **Npm:** Per la gestió de paquets Javascript i les dependències del projecte.
- **Webpack:** Per l'empaquetat dels diferents mòduls Javascript, juntament amb els recursos CSS, imatges, etc. per construir l'aplicació final que s'executa en el navegador.
- **Angular-material:** Interfícies d'usuari. Implementació en angular dels components publicats en la guia d'estil "Material-Design" de Google.
- **Angular-ui-router:** Llibreria utilitzada per definir el flux de navegació en aplicacions web desenvolupades en angular.
- **Restangular:** Llibreria que simplifica tot el diàleg d'aplicacions web angular amb APIs REST. L'emprem per consumir les diferents APIs que publica el back.
- **Websockets:** Per la implementació del sistema de missatgeria, de manera que els missatges que rep un usuari se visualitzin en calent a mesura que se van rebent, sense necessitat d'haver d'actualitzar la pàgina.

3.2.3. Desplegament i hosting

- **Apache:** Servidor web on es desplegarà el client web
- Pel back no necessitarem servidor d'aplicacions, ja que empaquetarem els dos mòduls publicats amb spring-boot, el qual embebeix un tomcat dins cada aplicació desplegada.
- **Git:** Sistema de control de versions dels dos mòduls: back i front.
- **BitBucket:** Servei d'allotjament basat en web per projectes que utilitzen git com sistema de control de versions.
- **Amazon EC2:** Servidor on es desplegarà la BBDD Neo4j, tomcat amb el back i apache amb el front.

- **Amazon S3:** Hosting de imatges.

3.3. Eines de desenvolupament

- **IntelliJIDEA:** Potent IDE de desenvolupament integrat especialment indicat per la programació JAVA. L'utilitzaré per la programació de la part back. S'han plantejat altres opcions igualment vàlides com Eclipse, no obstant, degut a la meva experiència professional estic més familiaritzat amb aquest IDE i dispenso d'una llicència pel seu ús.
- **Visual Studio Code:** Editor de codi obert especialment enfocat a la programació web. Destaca per la seva agilitat i versatilitat. És idoni per projectes Javascript, per lo que l'empraré en el desenvolupament de la part front.
- **MagicDraw:** Per l'elaboració de tots els diagrames UML del TFG.
- **GanttProject:** Eina de codi obert que utilitzaré per la planificació inicial del projecte, més concretament per l'elaboració del diagrama de Gantt del mateix.
- **Trello:** Per l'organització de les tasques durant la fase de disseny i implementació, aplicant una metodologia àgil.
- **Adobe XD:** Eina de prototipat per fer els mockups de l'aplicació. Conté els components material design de Google.

4. Anàlisi

4.1. Actors

A continuació enumerem els diferents actors que d'una manera o una altra interactuen amb l'aplicatiu. A mode recordatori, destacar que es requereix estar registrat per poder accedir a la plataforma.

4.1.1. Usuari no registrat (NRE)

Cas particular, ja que aquest actor únicament podrà accedir al formulari de registre per donar-se d'alta en la plataforma. Per donar-se d'alta, el sistema demanarà nom, llinatges, com utilitzarà l'aplicatiu (com a propietari o com inquilí) i les credencials d'accés per poder-se autenticar posteriorment (mail i contrasenya).

4.1.2. Usuaris web: propietari (PRO) i inquilí (INQ)

Actors principals de l'aplicatiu. Un cop l'usuari s'hagi registrat, podrà fer ús de la plataforma web en la seva totalitat. Les funcionalitats són pràcticament les mateixes independentment de que l'usuari empri la plataforma com a propietari o com inquilí. Ambdós actors poden:

- Autenticar-se en la plataforma web.
- Crear i gestionar anuncis de lloguer:
 - En el cas dels propietaris, els anuncis seran d'oferta d'habitatges per llogar.
 - En el cas dels inquilins, els anuncis seran de demanda d'habitatge.
- Escriure una referència a un altre usuari de la plataforma indicant com ha estat l'experiència de lloguer viscuda amb aquesta persona així com consultar les referències que han rebut els altres usuaris.
- Gestionar el seu perfil: informació personal, fotografies, referències escrites, etc.
- Contactar amb altres usuaris a través d'un servei de missatgeria privat.
- Gestionar l'agenda de contactes: afegir i eliminar contactes.
- Utilitzar un cercador d'anuncis de lloguer per poder trobar els anuncis que s'ajustin als seus criteris, podent a més visualitzar la informació detallada de cadascun.
 - En el cas dels propietaris, els anuncis retornats seran els anuncis de demanda publicats pels inquilins.
 - Els inquilins, per altra banda, poden trobar i consultar els anuncis d'oferta d'habitatge publicats pels propietaris.

4.1.3. Usuari webservice (WSU)

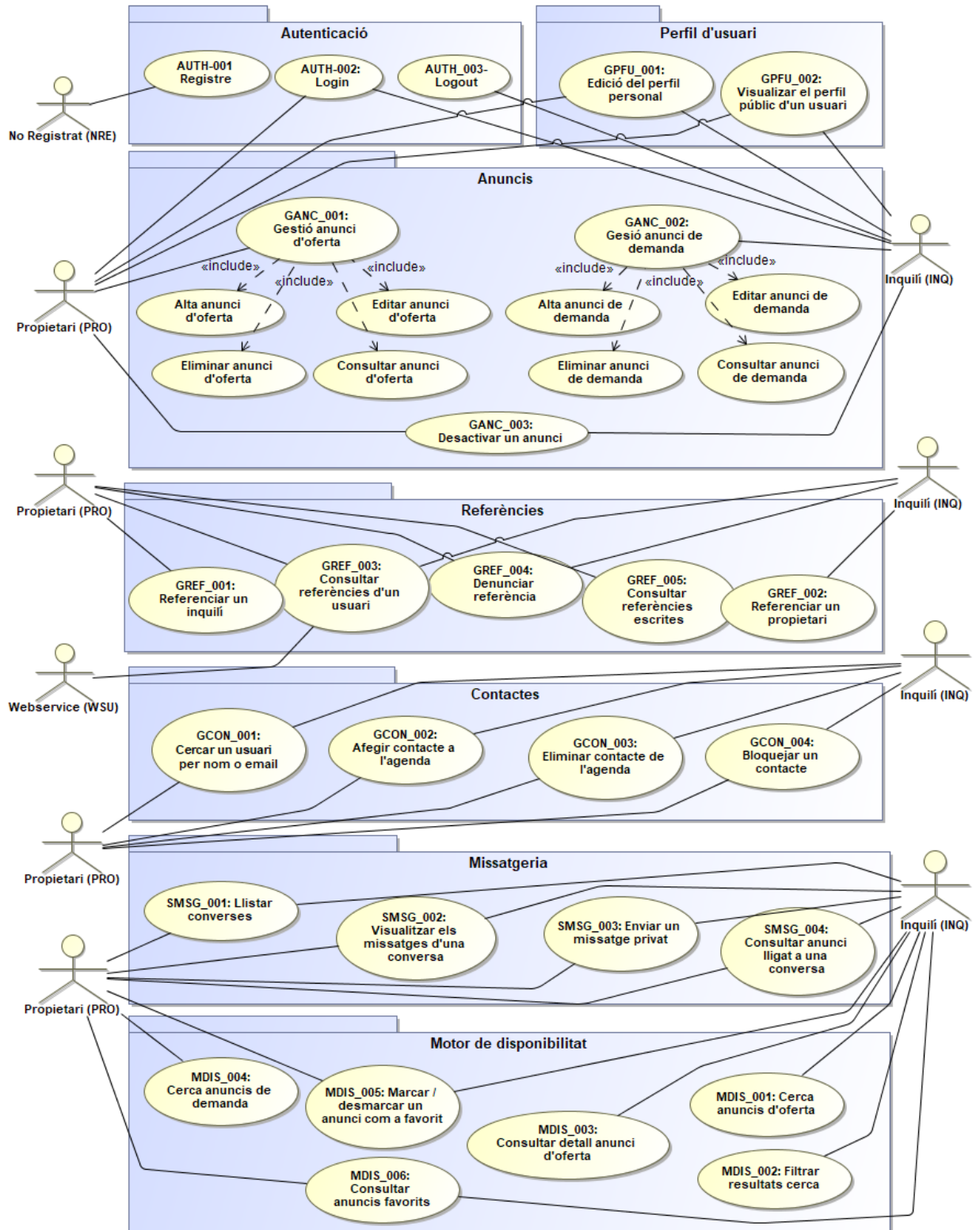
Actor que podrà consultar a través d'un webservice o API les referències que ha rebut un determinat usuari en la plataforma.

4.2. Diagrama de casos d'ús

Per claredat, s'ha decidit agrupar els casos d'ús en diferents paquets sobre la base de la funcionalitat que descriuen, indicant a més quin actor fa ús de cada funcionalitat. En quant a la granularitat, comentar que la gestió habitual d'una entitat (alta, baixa, modificació i consulta) s'ha agrupat en un mateix cas d'ús: Gestionar entitat.

- Registre i autenticació:
 - AUTH_001: Registrar-se en la sistema (NRE)
 - AUTH_002: Iniciar sessió en el sistema (PRO, INQ)
 - AUTH_003: Tancar sessió (PRO, INQ)
- Gestió d'anuncis:
 - GANC_001: Gestionar un anunci d'oferta d'habitatge: alta, baixa, modificació i consulta (PRO)
 - GANC_002: Gestionar un anunci de demanda d'habitatge: alta, baixa, modificació i consulta (INQ)
 - GANC_003: Desactivar temporalment un anunci (PRO, INQ)
- Gestió de referències i reputació:
 - GREF_001: Escriure una referència a un usuari inquilí (PRO)
 - GREF_002: Escriure una referència a un usuari propietari (INQ)
 - GREF_003: Consultar les referències que ha rebut un usuari (PRO, INQ, WSU)
 - GREF_004: Denunciar una referència abusiva o spam (PRO, INQ)
 - GREF_005: Consultar les referències escrites per un mateix (PRO, INQ)
- Gestió de contactes:
 - GCON_001: Cercar un usuari de la plataforma per nom o mail (PRO, INQ)
 - GCON_002: Afegir un usuari a l'agenda de contactes (PRO, INQ)
 - GCON_003: Eliminar un usuari de l'agenda de contactes (PRO, INQ)
 - GCON_004: Bloquejar un contacte (PRO, INQ)
 - GCON_005: Consultar els contactes de l'agenda (PRO, INQ)
- Sistema de missatgeria:
 - SMSG_001: Visualitzar el llistat de converses (PRO, INQ)
 - SMSG_002: Visualitzar els missatges d'una conversa (PRO, INQ)
 - SMSG_003: Enviar un missatge privat a un altre usuari (PRO, INQ)
 - SMSG_004: Consultar l'anunci al que (opcionalment) fa referència una conversa (PRO, INQ)
- Gestió del perfil d'usuari:
 - GPFU_001: Gestió del perfil personal: modificació i consulta (PRO, INQ)
 - GPFU_002: Visualitzar el perfil públic d'un usuari (PRO, INQ)
- Motor de disponibilitat:
 - MDIS_001: Cercar els anuncis d'oferta d'habitatge publicats en una zona (INQ)
 - MDIS_002: Filtrar i ordenar els resultats d'una cerca d'anuncis d'oferta d'habitatge (INQ)

- MDIS_003: Consultar la informació detallada d'un anunci d'oferta d'habitatge (INQ)
- MDIS_004: Cercar els anuncis de demanda d'habitatge d'una zona (PRO)
- MDIS_005: Marcar / desmarcar un anunci com a favorit (PRO, INQ)
- MDIS_006: Consultar anuncis favorits (PRO, INQ)



4.3. Fitxes de casos d'ús

Dels casos d'ús identificats en l'apartat anterior, a continuació s'adjunten les fitxes detallades dels més rellevants i complexes. No s'elaborarà fitxa per aquells casos d'ús trivials (com per exemple: tancar sessió, llistats i consultes bàsiques, etc), ja que no aporten valor de cara a la posterior implementació.

4.3.1. Registre i autenticació

AUTH_001: Registrar-se en la sistema		
Actors	Usuari no registrat (NRE)	
Descripció	Un usuari anònim accedeix al formulari de registre per tal de donar-se d'alta en la plataforma.	
Pre-condició	-	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la pantalla d'inici de la plataforma web.	
		2- El sistema mostra les opcions de "Login" i "Registre".
	3- L'usuari selecciona l'opció "Registre".	
		4- El sistema mostra un formulari demanant les dades de registre: "Nom i llinatges, "Tipus d'usuari (inquilí o propietari)", "Mail i contrasenya".
	5- L'usuari omple el formulari i fa click en el botó "Registrar-se".	
		6- El sistema valida que les dades introduïdes són correctes, dona d'alta l'usuari i el redirigeix a la home privada ja autenticat.
Flux alternatiu		6a1- El sistema valida les dades i detecta errors. Es mostra de nou el formulari de registre amb els camps invàlids ressaltats.
	6a2- L'usuari corregeix els errors del formulari .	
		6a3. Cas d'ús torna al punt 6.

AUTH_002: Iniciar sessió en el sistema		
Actors	Usuari propietari (PRO),Usuari inquilí (INQ)	
Descripció	Un usuari ja registrat inicia sessió en la plataforma web, autenticant-se amb les seves credencials.	
Pre-condició	L'usuari s'ha registrat en la plataforma (<i>Cas d'ús AUTH_001</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la pantalla d'inici de la plataforma web.	
		2- El sistema mostra les opcions de "Login" i "Registre".
	3- L'usuari selecciona la opció "Login".	
		4- El sistema mostra un formulari demanant les credencials d'accés: "Mail i contrasenya".
	5- L'usuari omple el formulari i fa click en el botó "Entrar".	
		6- El sistema valida que les dades introduïdes són correctes i redirigeix l'usuari autenticat a la home privada.
Flux alternatiu		2a1- El sistema detecta que l'usuari ja havia iniciat sessió amb anterioritat i automàticament autentica l'usuari i el redirigeix a la home privada. Opció coneguda com "Auto-login".
		6a1- El sistema intenta autenticar l'usuari però detecta que les credencials són incorrectes (usuari o contrasenya invàlids). Es mostra un missatge d'error d'autenticació.
	6a2- Cas d'ús torna al punt 5.	

Casos d'ús del subsistema que degut a la seva simplicitat s'ha decidit no fer fitxa detallada:

- AUTH_003: Tancar sessió (PRO, INQ)

L'usuari fa click en l'opció "Tancar sessió" de la home privada. El sistema redirigeix l'usuari a la pantalla d'inici de la plataforma web, la qual conté les opcions de "Login" i "Registre".

4.3.2. Gestió d'anuncis

NOTA: Aquells casos d'ús corresponents a la gestió d'una entitat (típicament alta, baixa, modificació i consulta), només si són especialment complexes o tenen alguna casuística particular, es dividiran en sub-casos d'ús, segons l'acció a realitzar. Exemple, "GANG_001: Gestionar un anunci d'oferta d'habitatge" se desglossarà en "GANG_001-1: Alta", "GANG_001-2 Modificació", i així successivament.

GANC_001-1: Gestionar un anunci d'oferta d'habitatge		
Actors	Usuari propietari (PRO)	
Descripció	Un usuari propietari dona d'alta un nou anunci d'oferta d'habitatge des de zero. Al ser una aplicació mobile-first, el formulari d'alta d'anunci es planteja com un "Wizard" o formulari guiat pas a pas, millorant la usabilitat i simplificant el procés.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (Cas d'ús AUTH_002)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Anuncis" de la home privada.	
		2- El sistema mostra un llistat amb els anuncis ja creats (amb un menú amb les opcions "Editar", "Eliminar" i "Desactivar") per a cadascun i l'opció "Crear nou anunci".
	3- L'usuari fa click en el botó "Crear nou anunci".	
		4- El sistema mostra el pas 1 del formulari guiat, demanant la informació corresponent a la propietat: "Tipus" (apartament, pis,...), "Metres quadrats", "Estat" (Nou, a reformar,...), "Nombre d'habitacions", "Nombre de banys", "Mobiliari" (Moblats, sense mobles,...), "Llistat de serveis que inclou" (wifi, calefacció, jardí,...).
	5- L'usuari omple el formulari i fa click en el botó "Següent".	
		6- El sistema valida les dades i passa al següent pas (2) del formulari guiat, demanant la informació corresponent a la ubicació: "Adreça".
7- L'usuari escriu i selecciona l'adreça, assistit amb l'opció d'autocompletat del formulari de		

	direccions (similar a com ho fa per exemple google maps)	
		8- El sistema mostra la ubicació de l'adreça introduïda per l'usuari geoposicionada damunt un mapa.
	9- L'usuari confirma que l'adreça és correcte fent click en el botó "Següent".	
		10. El sistema mostra un diàleg de confirmació amb un formulari editable amb les dades de l'adreça introduïda desglossades en: "Carrer", "Número", "Pis", "Porta", "Codi Postal" i "País".
	11- L'usuari, si escau, completa el formulari d'adreça desglossat (per exemple, corregint el codi postal i completant el número de porta) i fa click en el botó "Ok"	
		12- El sistema passa al següent pas del formulari guiat (3), demanant la informació corresponent a l'anunci: "Títol", "Descripció detallada" i "Fotografies".
	13- L'usuari omple el formulari i fa click en el botó "Següent".	
		14- El sistema valida les dades i passa al següent pas (4) del formulari guiat, demanant la informació corresponent a les condicions de l'anunci: "Preu", "Duració mínima i màxima" i "Elements restringits" (fumadors, animals...).
	15- L'usuari omple el formulari i fa click en el botó "Finalitzar".	
		16- El sistema valida les dades i dona d'alta l'anunci. Redirigeix l'usuari a la secció "Anuncis", on ja apareix el nou anunci creat en el llistat.
Flux alternatiu		6a1- El sistema valida les dades i detecta errors. Es mostra de nou el formulari amb els camps invàlids ressaltats.
	6a2- Cas d'ús torna al punt anterior.	
		14a1- Ídem pas 6a1
	6a2- Cas d'ús torna al punt anterior.	

		16a1- Ídem pas 6a1
	16a2- Cas d'ús torna al punt anterior.	

GANC_001-2: Gestionar un anunci d'oferta d'habitatge: Modificació		
Actors	Usuari propietari (PRO)	
Descripció	Un usuari propietari modifica les dades d'un anunci d'oferta d'habitatge que ha creat amb anterioritat. El flux és idèntic al procés d'alta (<i>Cas d'ús GANC_001-1</i>), però tots els camps del formulari guiat ja apareixen omplerts amb la informació actual de l'anunci, permetent-ne l'edició.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) L'usuari ha creat un anunci d'oferta d'habitatge (<i>Cas d'ús GANC_001-1</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Anuncis" de la home privada.	
		2- El sistema mostra un llistat amb els anuncis ja creats (amb un menú amb les opcions "Editar", "Eliminar" i "Desactivar") per a cadascun i l'opció "Crear nou anunci".
	3- L'usuari desplega les opcions del menú de l'anunci que vol editar i fa click en l'opció "Editar"	
		4- El sistema mostra el pas 1 del formulari guiat, omplert amb les dades actuals de l'anunci.
	5- L'usuari fa les modificacions pertinents sobre el formulari amb les dades actuals i fa click en l'opció següent.	
		6- A partir d'aquest punt el sistema fa el mateix procés que el punt 6 del cas d'ús <i>GANC_001-1: Gestionar un anunci d'oferta d'habitatge: Alta</i> , únicament que presenta el formulari ja omplert amb les dades actuals.
Flux alternatiu	Ídem GANC_001-1: Gestionar un anunci d'oferta d'habitatge	

GANC_001-3: Gestionar un anunci d'oferta d'habitatge: Eliminar		
Actors	Usuari propietari (PRO)	
Descripció	Un usuari propietari elimina un anunci d'oferta d'habitatge que ha creat amb anterioritat.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) L'usuari ha creat un anunci d'oferta d'habitatge (<i>Cas d'ús GANC_001-1</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Anuncis" de la home privada.	
		2- El sistema mostra un llistat amb els anuncis ja creats (amb un menú amb les opcions "Editar", "Eliminar" i "Desactivar") per a cadascun i l'opció "Crear nou anunci".
	3- L'usuari desplega les opcions del menú de l'anunci que vol eliminar i fa click en l'opció "Eliminar"	
		4- El sistema mostra un diàleg demanant confirmació de l'acció: "Cancel·lar" o "Eliminar"
	5- L'usuari confirma l'acció fent click en el botó "Eliminar"	
		6- El sistema elimina l'anunci i redirigeix l'usuari a la secció "Anuncis", on ja no hi apareix l'anunci eliminat en el llistat.
Flux alternatiu	5a1- L'usuari cancel·la l'acció fent click en el botó "Cancel·lar"	
		5a2- Cas d'ús torna al punt 2

GANC_002-1: Gestionar un anunci de demanda d'habitatge: Alta		
Actors	Usuari inquilí (INQ)	
Descripció	Un usuari inquilí dona d'alta un nou anunci de demanda d'habitatge.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>)	

Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Anuncis" de la home privada.	
		2- El sistema mostra un llistat amb els anuncis ja creats (amb les opcions "Editar", "Eliminar" i "Desactivar") i l'opció "Crear nou anunci".
	3- L'usuari fa click en el botó "Crear nou anunci".	
		4- El sistema mostra un formulari, demanant la informació corresponent a l'anunci: "Tipus de propietat desitjada" (pis, apartament,...), "Ciutat o regió on vol viure", "Descripció del que està cercant", "Pressupost" (preus mínim i màxim), "Restriccions de duració" (mínim i màxim de mesos), "Restriccions permeses" (que accepti mascotes, fumadors,...)
	5- L'usuari omple el formulari i fa click en el botó "Crear".	
		6- El sistema valida les dades i dona d'alta l'anunci. Redirigeix l'usuari a la secció "Anuncis", on ja apareix el nou anunci creat en el llistat.
Flux alternatiu		6a1- El sistema valida les dades i detecta errors. Es mostra de nou el formulari amb els camps invàlids ressaltats.
	6a2- Cas d'ús torna al punt anterior.	

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- GANC_001-4: Gestionar un anunci d'oferta d'habitatge: Desactivar

El flux és idèntic al cas d'ús "GANC_001-3: Gestionar un anunci d'oferta d'habitatge: Eliminar". En aquest cas, l'usuari selecciona l'opció "Desactivar" i en el punt 6, quan finalitza el cas d'ús, el sistema desactiva l'anunci i redirigeix l'usuari a la secció "Anuncis", on l'anunci apareix ressaltat com a desactivat.

- GANC_001-5: Gestionar un anunci d'oferta d'habitatge: Consultar

L'usuari accedeix a la secció "Anuncis" de la home privada i el sistema mostra un llistat amb els anuncis ja creats. Fent click damunt un anunci del llistat, l'usuari visualitza la informació detallada del mateix.

- GANC_002-2: Gestionar un anunci de demanda d'habitatge: Modificar
- GANC_002-3: Gestionar un anunci de demanda d'habitatge: Esborrar
- GANC_002-4: Gestionar un anunci de demanda d'habitatge: Desactivar
- GANC_002-5: Gestionar un anunci de demanda d'habitatge: Consultar

Casos d'ús anàlegs a la gestió d'anuncis d'oferta d'habitatge, però per anuncis de demanda: *GANC_001-2*, *GANC_001-3*, *GANC_001-4*, *GANC_001-5*,

4.3.3. Gestió de referències i reputació

GRAF_001: Escriure una referència a un usuari inquilí		
Actors	Usuari propietari (PRO)	
Descripció	<p>Un usuari propietari escriu una referència a un usuari inquilí amb el qual ha compartit una experiència de lloguer prèvia.</p> <p>L'opció "Escriure Referència" està disponible en diverses seccions de l'aplicatiu:</p> <ul style="list-style-type: none"> • Des de el perfil públic d'un usuari. • Des de la pantalla de visualització de referències rebudes per un usuari • Des de l'agenda de contactes <p>Al igual que en la creació d'anunci, el formulari també es planteja com un "Wizard" o formulari guiat pas a pas, millorant la usabilitat i simplificant el procés.</p>	
Pre-condició	<p>L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>)</p> <p>Una de les següents:</p> <ul style="list-style-type: none"> • L'usuari visualitza el perfil públic d'un usuari (<i>Cas d'ús GPFU_002</i>) • L'usuari visualitza les referències rebudes per un usuari (<i>Cas d'ús GREF_003</i>) • L'usuari visualitza els contactes de l'agenda (<i>Cas d'ús GCON_005</i>) 	
Flux principal	Actor	Sistema
	1- L'usuari selecciona l'opció "Escriure referència".	
		2- El sistema mostra el pas 1 del formulari guiat, demanant la informació corresponent a l'estància de la referència: "Rang de dates" i "Adreça"
	3- L'usuari omple les dates, escriu i selecciona l'adreça, assistit amb l'opció d'autocompletat del formulari de direccions (similar a com ho fa per exemple google maps)	
		4- El sistema mostra la ubicació de l'adreça introduïda per l'usuari geoposicionada damunt un mapa.

	5- L'usuari fa click en el botó "Següent".	
		6. El sistema mostra un diàleg de confirmació amb un formulari editable amb les dades de l'adreça introduïda desglossades en: "Carrer", "Número", "Pis", "Porta", "Codi Postal" i "País".
	7- L'usuari, si escau, completa el formulari d'adreça desglossat (per exemple, corregint el codi postal i completant el número de porta) i fa click en el botó "Ok"	
		8- El sistema valida les dades i passa al següent pas del formulari guiat (2), demanant la informació corresponent a l'experiència viscuda: "Com va ser?" (positiva o negativa), "Descripció detallada", i un llistat de possibles "Qualitats a destacar" (pagador puntual, de confiança, net,...)
	9- L'usuari omple el formulari i fa click en el botó "Finalitzar".	
		10- El sistema valida les dades i dona d'alta la referència. Es redirigeix la navegació a la pàgina de consulta de referències rebudes de l'usuari referenciat.
Flux alternatiu		8a1- El sistema valida les dades i detecta errors. Es mostra de nou el formulari amb els camps invàlids ressaltats.
	8a2- Cas d'ús torna al punt anterior.	
		10a1- Ídem pas 8a1
	6a2- Cas d'ús torna al punt anterior.	

GRAF_010: Denunciar una referència abusiva o spam	
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)
Descripció	Un usuari denuncia una referència ja que considera que el seu contingut és inapropiat, fals o spam.

Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) L'usuari visualitza les referències rebudes per un usuari (<i>Cas d'ús GREF_003</i>)	
Flux principal	Actor	Sistema
	1- L'usuari fa click sobre el menú d'opcions de la referència que vol denunciar i selecciona l'opció "Denunciar"	
		2- El sistema mostra un diàleg demanant que s'indiqui el motiu de la denuncia: "Spam", "Contingut inapropiat" o "Informació no verídica".
	3- L'usuari selecciona el motiu de la denuncia i fa click en el botó "Denunciar"	
		4- El sistema mostra un diàleg demanant confirmació de l'acció: "Cancel·lar" o "Denunciar"
	5- L'usuari confirma l'acció fent click en el botó "Denunciar"	
		6- El sistema envia un mail al administrador del sistema amb la informació de la referència denunciada i mostra un missatge a l'usuari indicant que "La referència serà revisada en breu, gràcies per la col·laboració".
Flux alternatiu	5a1- L'usuari cancel·la l'acció fent click en el botó "Cancel·lar"	
		5a2- Cas d'ús torna al punt 2

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- GREF_002: Escriure una referència a un usuari propietari (INQ)

Cas d'ús anàleg a "GREF_001: Escriure una referència a un usuari inquilí" però en el sentit invers (autor: inquilí, destinatari: propietari). L'única variació és que les "Qualitats a destacar" del punt 8 seran diferents (Comunicatiu, prest a ajudar, localitzable,...)

- GREF_003: Consultar les referències que ha rebut un usuari (PRO, INQ, WSU)
 - Des de el perfil públic d'un usuari (Veure cas d'ús "GPFU_002: Visualitzar el perfil públic d'un usuari") hi haurà una secció per poder consultar les referències que aquest ha rebut. Es tracta d'un simple llistat amb la informació detallada al complet de cada referència rebuda per l'usuari, ordenades per data.

- En quant a la consulta via API, en la petició ens indicaran el mail de l'usuari a consultar, mentre que en la resposta retornarem la informació detallada de les referències rebudes en format XML o JSON (per definir)
- GREF_003: Consultar les referències que ha escrit un mateix (PRO, INQ)

Un usuari podrà veure les referències que ha escrit a altres usuaris. En la home privada, hi haurà una opció "Referències escrites" on s'hi podrà consultar un llistat bàsic de totes les referències escrites. Cal recordar que per mantenir la fiabilitat i veracitat de les referències, un cop escrites no es poden editar ni eliminar, així que es tracta únicament d'una pàgina de consulta.

4.3.4. Gestió de contactes

GCON_001: Cercar un usuari de la plataforma per nom o mail		
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)	
Descripció	Un usuari cerca un altre usuari dins la plataforma, a partir del nom o correu electrònic.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Contactes" de la home privada.	
		2- El sistema mostra un llistat amb els contactes actuals i un formulari amb un únic camp amb l'opció "Cercar"
	3- L'usuari escriu el nom o mail (o part d'ells) en el formulari de cerca.	
		4- A mesura que l'usuari va escrivint, el sistema mostra un llistat d'usuaris tals que el seu nom o mail coincideix amb l'input de l'usuari. Els resultats es van actualitzant dinàmicament a mesura que l'usuari canvia les dades de la cerca.
	5- L'usuari selecciona del llistat de suggeriments l'usuari que vol consultar fent click sobre ell.	
	6- El sistema redirigeix la navegació a la pàgina del perfil públic de l'usuari seleccionat, mostrant la seva informació.	

Flux alternatiu		4a1- No hi ha cap usuari donat d'alta en el sistema que compleixi amb els paràmetres de cerca. El sistema mostra un missatge indicant que no s'han trobat resultats.
	5a2- Cas d'ús torna al punt 3	

GCON_002: Afegir un usuari a l'agenda de contactes		
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)	
Descripció	<p>Un usuari afegeix un altre usuari a l'agenda de contactes.</p> <p>L'opció "Afegir a contactes" està disponible en diverses seccions de l'aplicatiu:</p> <ul style="list-style-type: none"> • Des de el perfil públic d'un usuari. • Des de la pantalla de visualització de referències rebudes per un usuari • Des de la pròpia agenda de contactes, en el formulari de cerca d'usuaris. 	
Pre-condició	<p>L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) Una de les següents:</p> <ul style="list-style-type: none"> • L'usuari visualitza el perfil públic d'un usuari (<i>Cas d'ús GPFU_002</i>) • L'usuari visualitza les referències rebudes per un usuari (<i>Cas d'ús GREF_003</i>) • L'usuari ha trobat l'usuari cercant-lo per mail o nom a través del formulari de cerca de la secció contactes (<i>Cas d'ús GCON_002</i>) 	
Flux principal	Actor	Sistema
	1- L'usuari selecciona l'opció "Afegir a Contactes"	
		2- El sistema afegeix l'usuari a l'agenda de contactes i mostra un missatge indicant que l'acció s'ha executat correctament.
Flux alternatiu		2a1- Si l'acció s'ha executat des de l'agenda de contactes, el sistema afegeix l'usuari i mostra el llistat de contactes on apareix el nou usuari.

GCON_003: Eliminar un usuari a l'agenda de contactes		
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)	
Descripció	Un usuari elimina un usuari de la seva agenda de contactes.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) L'usuari ha afegit com a mínim un usuari a l'agenda de contactes (<i>Cas d'ús GCON_002</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Contactes" de la home privada.	
		2- El sistema mostra un llistat amb els contactes actuals, amb un menú amb les opcions "Eliminar" i "Bloquejar" per a cadascun d'ells.
	3- L'usuari desplega les opcions del menú del contacte que vol eliminar i fa click en l'opció "Eliminar"	
		4- El sistema mostra un diàleg demanant confirmació de l'acció: "Cancel·lar" o "Eliminar"
	5- L'usuari confirma l'acció fent click en el botó "Eliminar"	
		6- El sistema elimina el contacte i retorna a la secció "Contactes", on ja no apareix l'usuari eliminat de l'agenda.
Flux alternatiu	5a1- L'usuari cancel·la l'acció fent click en el botó "Cancel·lar"	
		5a2- Cas d'ús torna al punt 2

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- GCON_004: Bloquejar un contacte (PRO, INQ)

El flux és idèntic al cas d'ús "GCON_003: Eliminar un usuari a l'agenda de contactes". En aquest cas, l'usuari selecciona l'opció "Bloquejar" i en el punt 6, quan finalitza el cas d'ús, el sistema bloqueja l'anunci i redirigeix l'usuari a la secció "Contactes", on l'usuari apareix ressaltat com a bloquejat.

- GCON_005: Consultar els contactes de l'agenda

L'usuari accedeix a la secció "Contactes" de la home privada i el sistema mostra un llistat amb els usuaris actuals de l'agenda. Fent click damunt un usuari del llistat, l'usuari accedeix a la pàgina de visualització del seu perfil públic.

4.3.5. Sistema de missatgeria

SMSG_003: Enviar un missatge privat a un altre usuari		
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)	
Descripció	<p>Un usuari envia un missatge a un altre usuari a través del servei de missatgeria privat del sistema.</p> <p>L'opció "Envia missatge" està disponible en diverses seccions de l'aplicatiu:</p> <ul style="list-style-type: none"> • Des de la secció "Missatges", on apareix el llistat de converses actuals. • Des de el perfil públic d'un usuari. • Des de la pantalla de visualització de referències rebudes per un usuari • Des de l'agenda de contactes. • Des de la informació d'un anunci retornat en la resposta d'una cerca. <p>El flux principal definit en aquesta fitxa suposa que l'usuari envia el missatge des de la secció "Missatges" de la home privada. En el cas de fer-ho des de una de les altres possibles seccions, l'usuari únicament necessitarà fer click en el botó "Envia missatge" i directament s'obrirà la conversa en curs (sense missatges si ambdós usuaris encara no n'han intercanviat cap). Veure flux alternatiu.</p>	
Pre-condició	<p>L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) Una de les següents:</p> <ul style="list-style-type: none"> • L'usuari visualitza el llistat de converses (<i>Cas d'ús SMSG_001</i>) • L'usuari visualitza el perfil públic d'un usuari (<i>Cas d'ús GPFU_002</i>) • L'usuari visualitza les referències rebudes per un usuari (<i>Cas d'ús GREF_003</i>) • L'usuari visualitza el llistat de contactes de l'agenda (<i>Cas d'ús GCON_005</i>) • L'usuari és propietari i ha fet una cerca d'anuncis de demanda d'habitatge (<i>Cas d'ús MDIS_004</i>) • L'usuari és inquilí i ha fet una cerca d'anuncis d'oferta d'habitatge (<i>Cas d'ús MDIS_004</i>) i posteriorment ha visualitzat la informació detallada de l'anunci (<i>Cas d'ús MDIS_003</i>) 	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Missatges" de la home privada.	
		2- El sistema mostra un llistat de les converses actuals, indicant per a cadascuna d'elles l'usuari interlocutor, data i text del darrer missatge enviat. Opcionalment, si la conversa fa referència a un anunci, també es mostra el títol de l'anunci.

	3- L'usuari selecciona la conversa de l'usuari al que vol enviar el missatge fent click sobre ella.	
		4- El sistema mostra els missatges de la conversa en curs, amb el típic format de d'un servei de missatgeria estàndard: ordenats per data, missatges enviats a la dreta, rebuts a l'esquerra. A la part inferior, un formulari amb una àrea de text i l'opció "Enviar".
	4- L'usuari escriu el missatge a l'àrea de text i fa click en el botó "Enviar"	
		5- El sistema dóna d'alta el missatge i l'envia a l'usuari receptor. Els missatges visualitzats de la conversa en curs s'actualitzen i ja pareix el nou missatge. Es genera una notificació amb la informació del missatge, que l'usuari receptor visualitzarà en la seva home privada.
Flux alternatiu	1a1- L'usuari selecciona l'opció "Enviar Missatge" des de una de les altres seccions on es permet enviar un missatge.	
		1a2- Cas d'ús passa al punt 4
		2a1- No hi ha cap conversa en curs, finalitza el cas d'ús. L'usuari podrà iniciar una conversa des de una de les altres seccions descrites: des del perfil públic, pàgina de referències, agenda de contactes.

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- SMSG_001: Visualitzar el llistat de converses (PRO, INQ)
- SMSG_002: Visualitzar els missatges d'una conversa (PRO, INQ)
- SMSG_004: Consultar l'anunci al que (opcionalment) fa referència una conversa (PRO, INQ)

El flux d'aquests casos d'ús queda cobert amb els passos 1 i 2 del flux principal del cas d'ús "SMSG_003: Enviar un missatge privat a un altre usuari".

4.3.6. Gestió del perfil d'usuari

GPFU_001: Edició del perfil personal		
Actors	Usuari propietari (PRO) i usuari inquilí (INQ)	
Descripció	Un usuari modifica les dades del seu perfil personal.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>)	
Flux principal	Actor	Sistema
	1- L'usuari accedeix a la secció "Perfil" de la home privada.	
		2- El sistema mostra la informació actual del perfil personal i un botó amb l'opció "Editar"
	3- L'usuari fa click en el botó "Editar"	
		4- El sistema mostra un formulari editable omplert amb la informació actual del perfil d'usuari, que inclou: "Gènere", "Nom i Llinatges", "Telèfon", "Data de naixement", "Ciutats natal i de residència", "Estudis", "Professió", una "Descripció lliure", fotografies de "Perfil", "Portada" i "Addicionals".
	5- L'usuari omple el formulari amb les dades que vol modificar i fa click en el botó "Editar"	
		6- El sistema valida les dades i, realitza les modificacions del perfil i mostra un missatge indicant que els canvis s'han efectuat correctament.
Flux alternatiu		6a1- El sistema valida les dades i detecta errors. Es mostra de nou el formulari amb els camps invàlids ressaltats.
	6a2- Cas d'ús torna al punt anterior.	

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- GPFU_002: Visualitzar el perfil públic d'un usuari (PRO, INQ)

Aquest cas d'ús únicament contempla la consulta de la informació del perfil públic d'un usuari, el qual conté la informació especificada en el pas 4 del cas d'ús "GPFU_001: Edició del perfil personal"

4.3.7. Motor de disponibilitat

MDIS_001: Cercar els anuncis d'oferta d'habitatge publicats en una zona		
Actors	Usuari inquilí (INQ)	
Descripció	Un usuari cerca els anuncis d'oferta d'habitatge disponibles en una determinada zona.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>)	
Flux principal	Actor	Sistema
		1- En la home privada, el sistema mostra un cercador amb un únic camp a omplir ("Població") i el botó "Cercar".
	2- L'usuari escriu i selecciona la població en el cercador, assistit amb l'opció d'autocompletat del formulari (similar a com ho fa per exemple google maps) i fa click en el botó "Cercar".	
		3- El sistema mostra els anuncis disponibles, en format llistat i geo-posicionats damunt un mapa. De cada anunci en mostra la informació més rellevant ("Preu, "Fotografia", "Número habitacions", "Estat" i "Anunciant") juntament amb les opcions "Veure Detall", "Consultar referències anunciant", "Marcar com a favorit". A nivell global, mostra la opció "Filtrar resultats".
Flux alternatiu		3a1- No hi ha anuncis disponibles a la zona indicada. Es mostra un missatge indicant que no existeixen resultats per la cerca. No apareix el botó "Filtrar resultats".
		3b1- La cerca s'ha realitzat des d'un dispositiu mòbil: El sistema mostra els anuncis disponibles, únicament en format llistat. De cada anunci en mostra la informació més rellevant ("Preu, "Fotografia", "Número habitacions", "Anunciant") juntament amb les opcions "Veure Detall", "Consultar referències anunciant", "Marcar com a favorit". A nivell global, mostra la opció "Filtra resultats".

MDIS_002: Filtrar i ordenar els resultats d'una cerca d'anuncis d'oferta d'habitatge		
Actors	Usuari inquilí (INQ)	
Descripció	Un usuari filtra i/o ordena els resultats de la resposta d'una cerca, en base als diferents criteris disponibles.	
Pre-condició	L'usuari ha iniciat sessió en la plataforma web (<i>Cas d'ús AUTH_002</i>) L'usuari fet una cerca d'anuncis d'oferta d'habitatge en una zona i ha obtingut resultats (<i>Cas d'ús MDIS_001</i>)	
Flux principal	Actor	Sistema
	1- L'usuari fa click en el botó "Filtrar resultats" de la pàgina de resultats de la cerca.	
		2- El sistema mostra un formulari amb els diferents camps pels quals es permet filtrar ("Mínim d'habitacions", "Mobiliari, "Estat" i "Serveis inclosos") i els criteris d'ordenació: "Preu ascendent", "Preu descendent" i "Més recent". El sistema calcula els valors de cada camp dinàmicament en funció dels resultats.
	2- L'usuari omple el formulari amb els criteris de filtratge i ordenació desitjats i fa click en el botó "Filtrar".	
		3- El sistema realitza el procés de filtrat ordenació i s'actualitzen els resultats de la resposta.
Flux alternatiu	2a1- L'usuari cancel·la el filtrat / ordenació de resultats fent click en el botó "Cancel·lar".	
		2a2- El sistema torna a la pàgina de resultats original, sense alterar la resposta.

Casos d'ús del subsistema que degut a la seva simplicitat o similitud amb altres casos d'ús s'ha decidit no fer fitxa detallada:

- MDIS_003: Consultar la informació detallada d'un anunci d'oferta d'habitatge (INQ)

Tal com s'ha indicat en el flux del cas d'ús "MDIS_001: Cercar els anuncis d'oferta d'habitatge publicats en una zona", cada anunci té l'opció de consultar-ne el detall. Si l'usuari fa click en aquesta opció, visualitzarà la informació detallada de l'anunci en qüestió. Veure cas d'ús "GANC_001-

1: Gestionar un anunci d'oferta d'habitatge: Alta" per consultar el detall de quina és la informació mostrada.

- *MDIS_004*: Cercar els anuncis de demanda d'habitatge d'una zona (PRO)

Cas d'ús anàleg a "MDIS_001: Cercar els anuncis d'oferta d'habitatge publicats en una zona", però per realitzar la cerca d'anuncis de demanda. El procés de cerca és molt similar, però més simplificat, ja que donada la senzillesa d'aquests tipus d'anuncis, no s'oferirà:

- Possibilitat de filtrar / ordenar els resultats.
 - Geo-posicionament damunt un mapa.
 - Consulta informació detallada (en el llistat ja apareix tota la informació rellevant)
- *MDIS_005*: Marcar / desmarcar un anunci com a favorit (PRO, INQ)
MDIS_006: Consultar anuncis favorits (PRO, INQ)

Com s'ha indicat en el cas d'ús "MDIS_001: Cercar els anuncis d'oferta d'habitatge publicats en una zona", un anunci es pot marcar com a favorit. Posteriorment, els usuaris tindran en la home privada un accés directe on poder consultar aquests anuncis destacats. Els resultats es mostraran exactament amb el mateix format que els resultats d'una cerca d'anuncis normal.

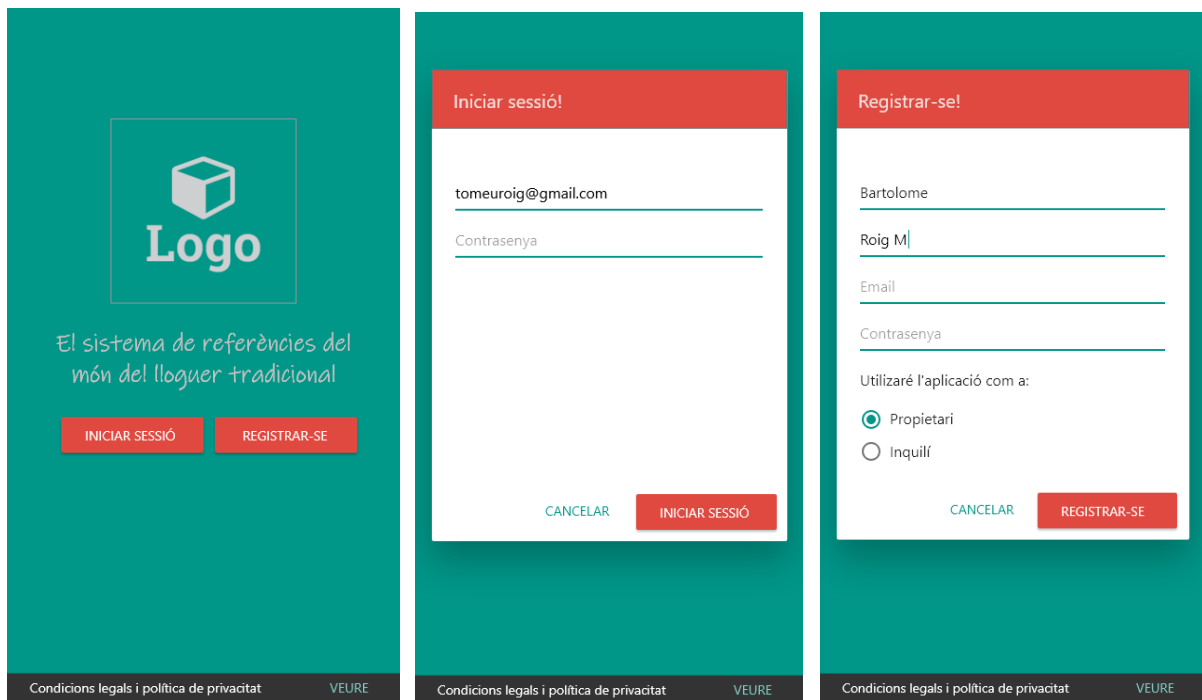
5. Prototipat

Al ser una aplicació mobile-first, la majoria de prototipus s'han fet amb la resolució de pantalla mòbil. No obstant, l'aplicació és responsive i s'adaptarà a resolucions més grans: Tablet i escriptori. Només en aquells casos en que la vista d'escriptori difereix molt de la de mòbil (per exemple en la vista de resultats de la cerca d'anuncis) s'han fet dos prototipus diferents.

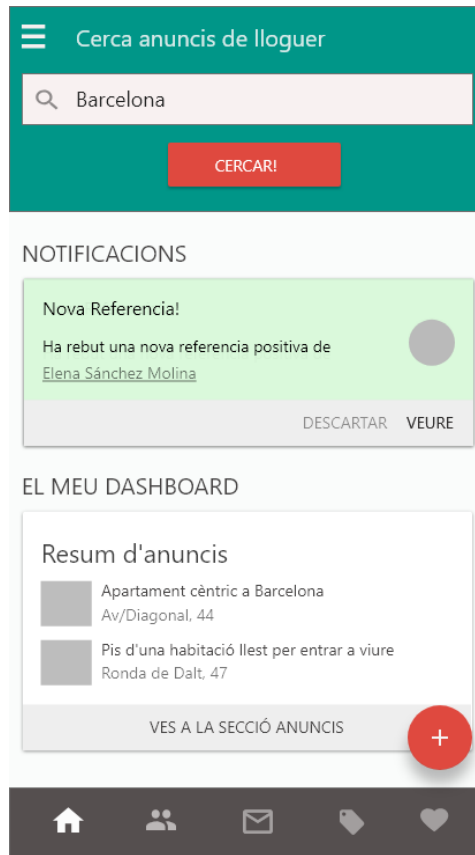
Pel prototipat he emprat l'eina Adobe xd, principalment per tres motius:

- L'eina és gratuïta i no necessita llicència.
- Disposa del set de components de material design, la guia d'estil de google que utilitzaré en la interfície d'usuari. Així, és fàcil fer uns mockups amb un disseny acceptable que s'acosta bastant a la realitat.
- Està especialment enfocat al prototipat d'aplicacions multi-dispositiu.

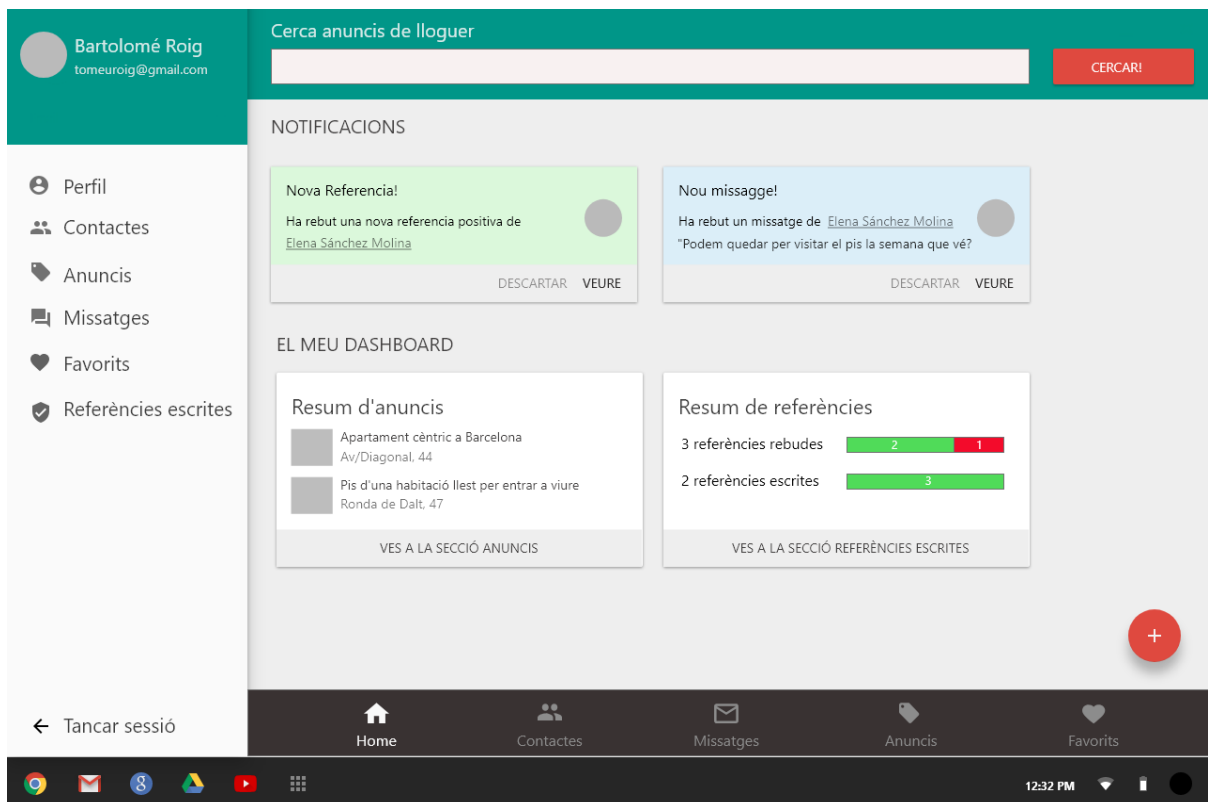
5.1. Registre i autenticació



AUTH_001: Registrar-se en la sistema (NRE)
AUTH_002: Iniciar sessió en el sistema (PRO, INQ)



Home privada mòbil: Vista mòbil de la home privada un cop l'usuari s'ha autenticat




Home privada escriptori: Vista escriptori de la home privada un cop l'usuari s'ha autenticat

5.2. Gestió d'anuncis

← Anuncis d'oferta de lloguer


Apartament d'una habitació en perfecte estat



Editar
Eliminar
Desactivar

750 € pes mes VEURE DETALL

Àtic de luxe en primera línia de platja



Nou anunci +

← Nou anunci de lloguer

1: Informació de la vivenda

Tipus de vivenda ▼

Metres quadrats Estat ▼

Nº habitacions ▼ Nº banys ▼

Mobiliari ▼

Serveis incluits

Wifi
 Parking
 Adaptat

...


SEGÜENT →

← Nou anunci de lloguer

2: Adreça

Av diagonal 44 🔍

Av diagonal 44, Can Pícafort
 Av diagonal 44, Castelledefels
 Av diagonal 44, Barcelona



← ANTERIOR SEGÜENT →

← Nou anunci de lloguer

2: Adreça

Per favor comprovi i completi l'adreça

Avinguda Diagonal

44 Pis Porta

07458

Can Pícafort

Espanya

CANCELAR CONFIRMAR



← Nou anunci de lloguer

3: Descripció de l'anunci

Apartament cèntric d'una habitació

Excel·lent apartament en la millor zona de Can Pícafort. Llest per entrar a viure i totalment equipat.
Col·legis i hospitals a pocs minuts caminant.

No deixis perdre l'oportunitat

+

← ANTERIOR SEGÜENT →

← Nou anunci de lloguer ✓

4: Preu i restriccions

750 € Mensuals

1 mes ▼ Estància màxima ▼

Disponible a partir de 📅

Altres condicions

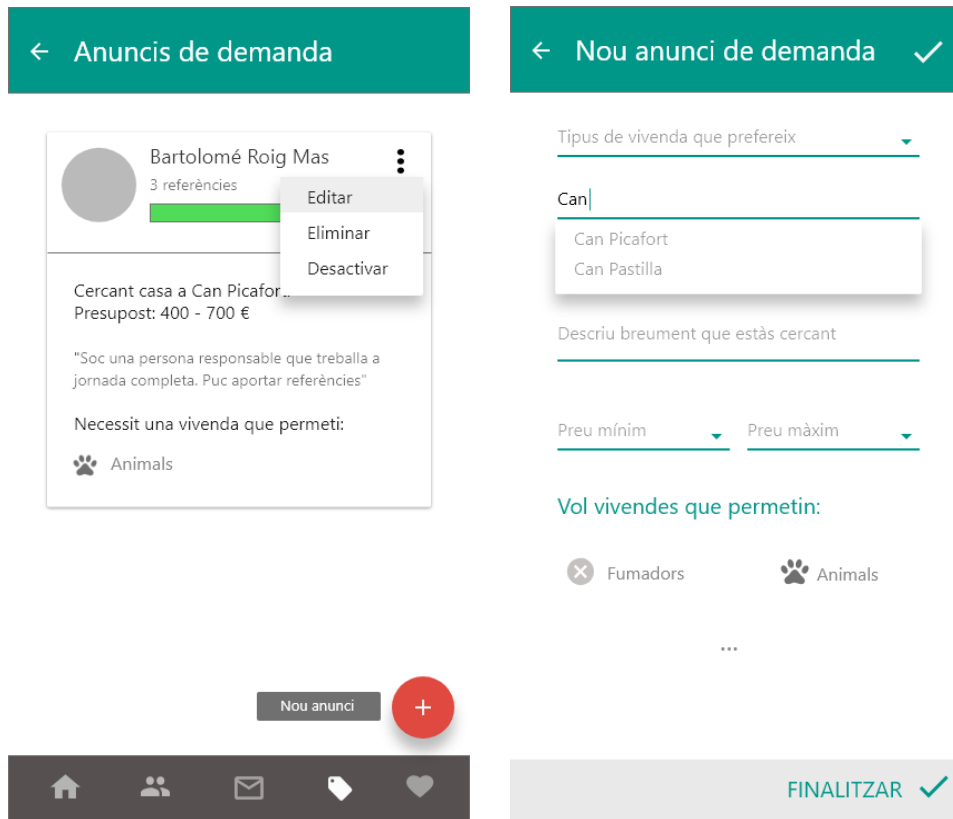
No es permeten:

Fumadors
 Animals

...

← ANTERIOR FINALITZAR ✓

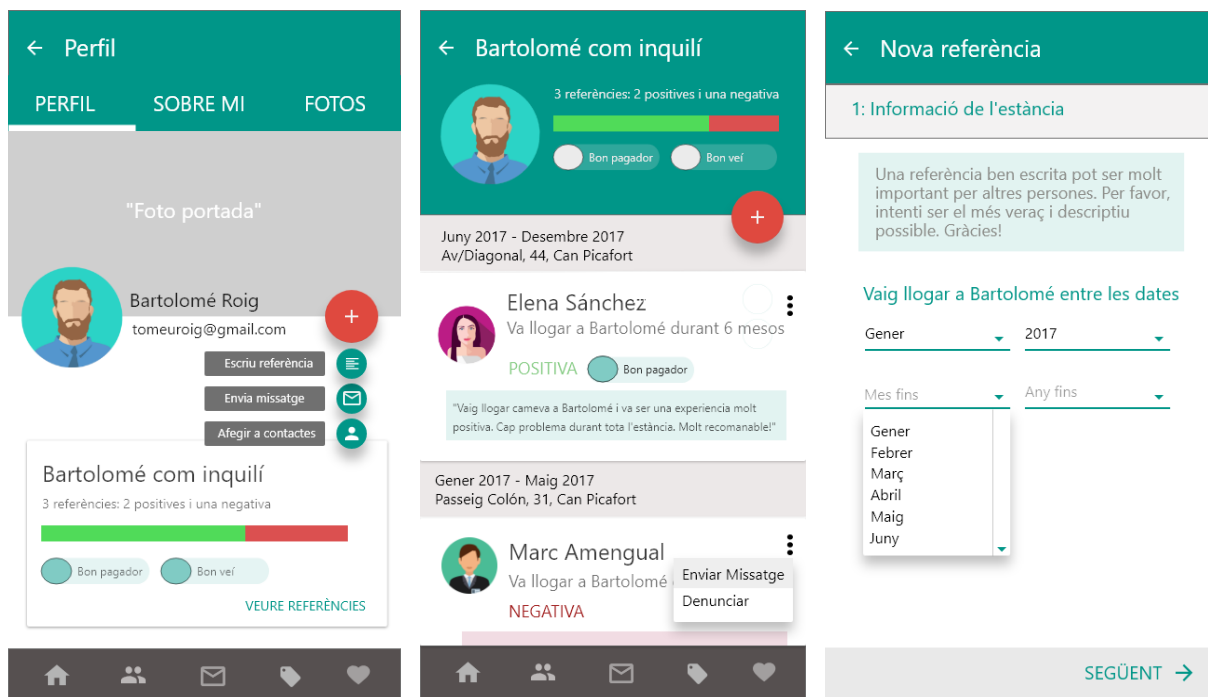
GANC_001: Gestionar un anunci d'oferta d'habitatge: alta, baixa, modificació i consulta (PRO)
GANC_003: Desactivar temporalment un anunci (PRO)

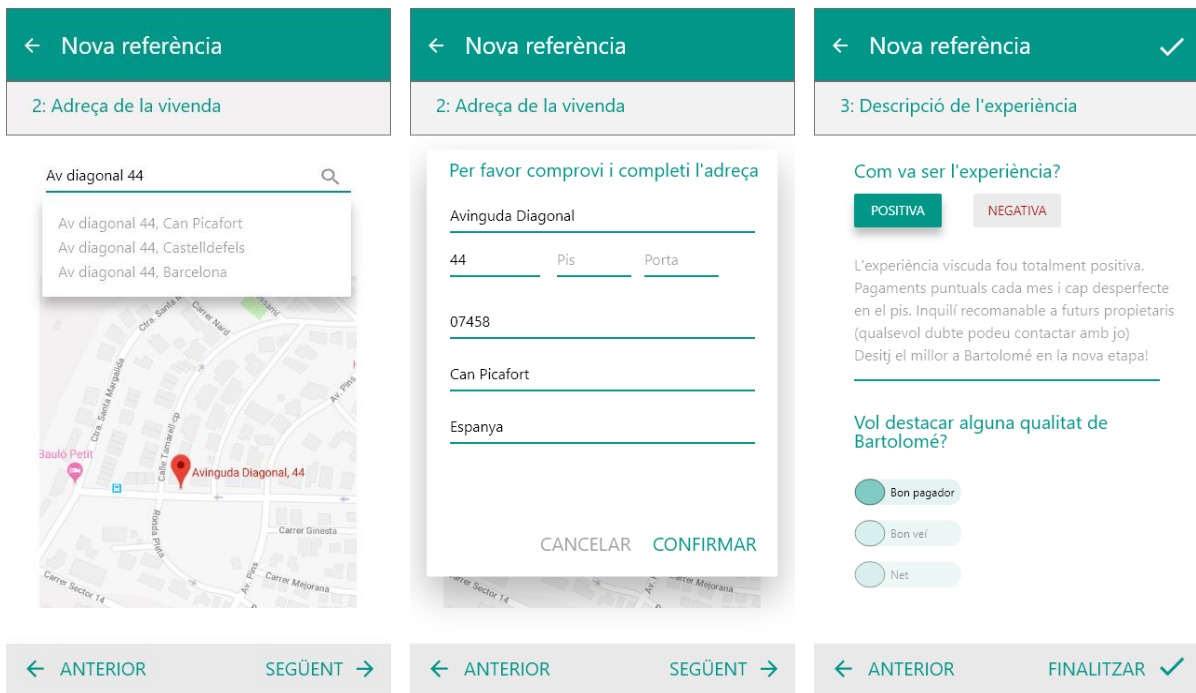


GANC_002: Gestionar un anunci de demanda d'habitatge: alta, baixa, modificació i consulta (INQ)

GANC_003: Desactivar temporalment un anunci (INQ)

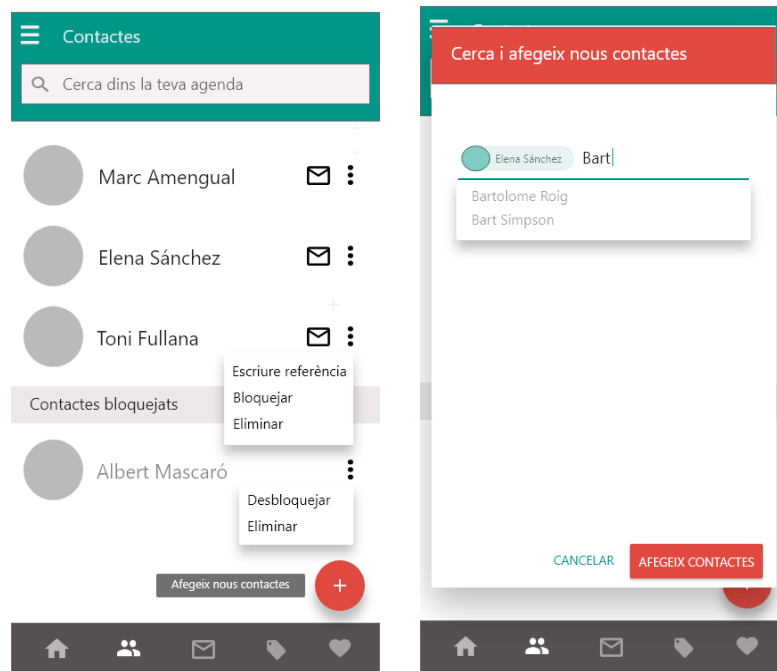
5.3. Gestió de referències i reputació





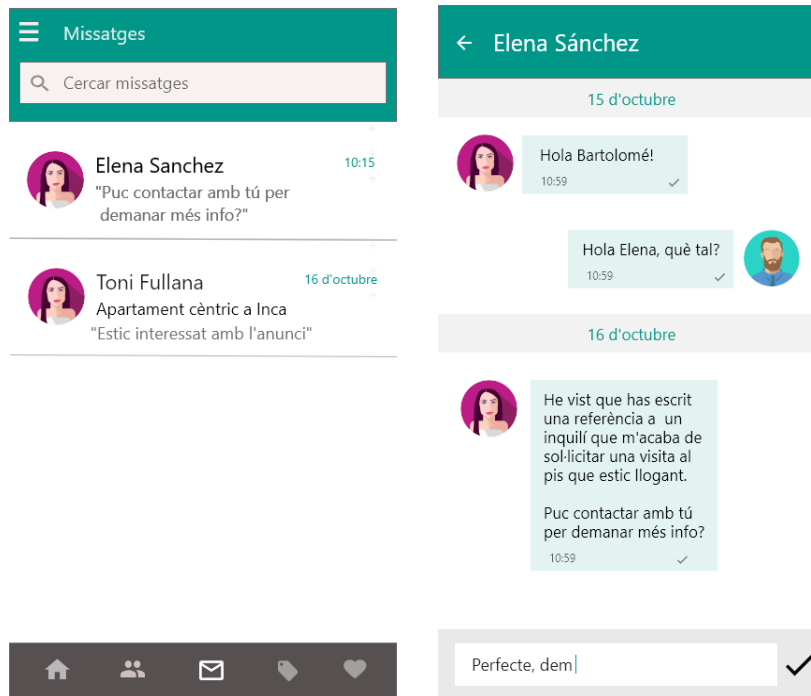
- GRAF_001:** Escriure una referència a un usuari inquilí (PRO)
- GRAF_003:** Consultar les referències que ha rebut un usuari (PRO)
- GRAF_004:** Denunciar una referència abusiva o spam (PRO, INQ)

5.4. Gestió de contactes



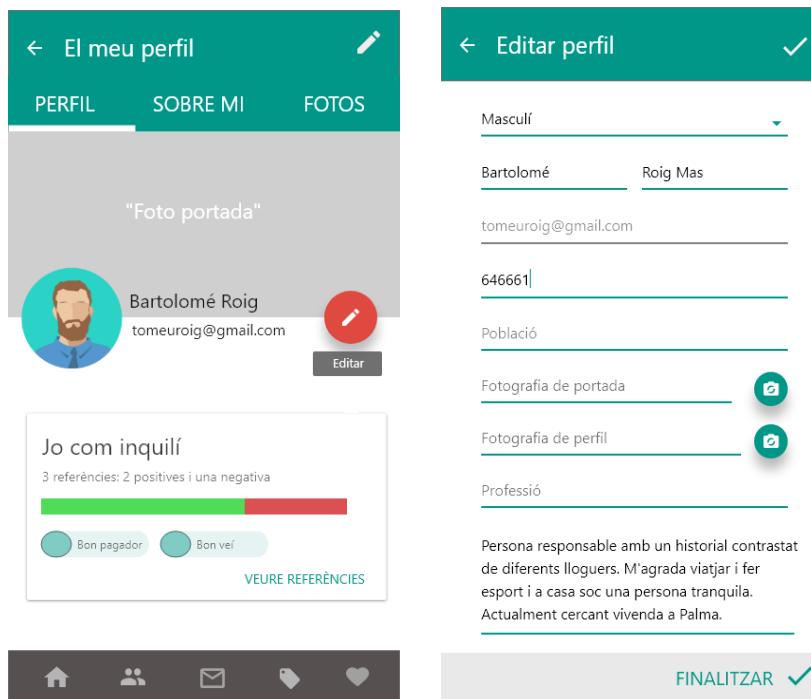
- GCON_001:** Cercar un usuari de la plataforma per nom o email (PRO, INQ)
- GCON_002:** Afegir un usuari a l'agenda de contactes (PRO, INQ)
- GCON_003:** Eliminar un usuari de l'agenda de contactes (PRO, INQ)
- GCON_004:** Bloquejar un contacte (PRO, INQ)
- GCON_005:** Consultar els contactes de l'agenda (PRO, INQ)

5.5. Sistema de missatgeria



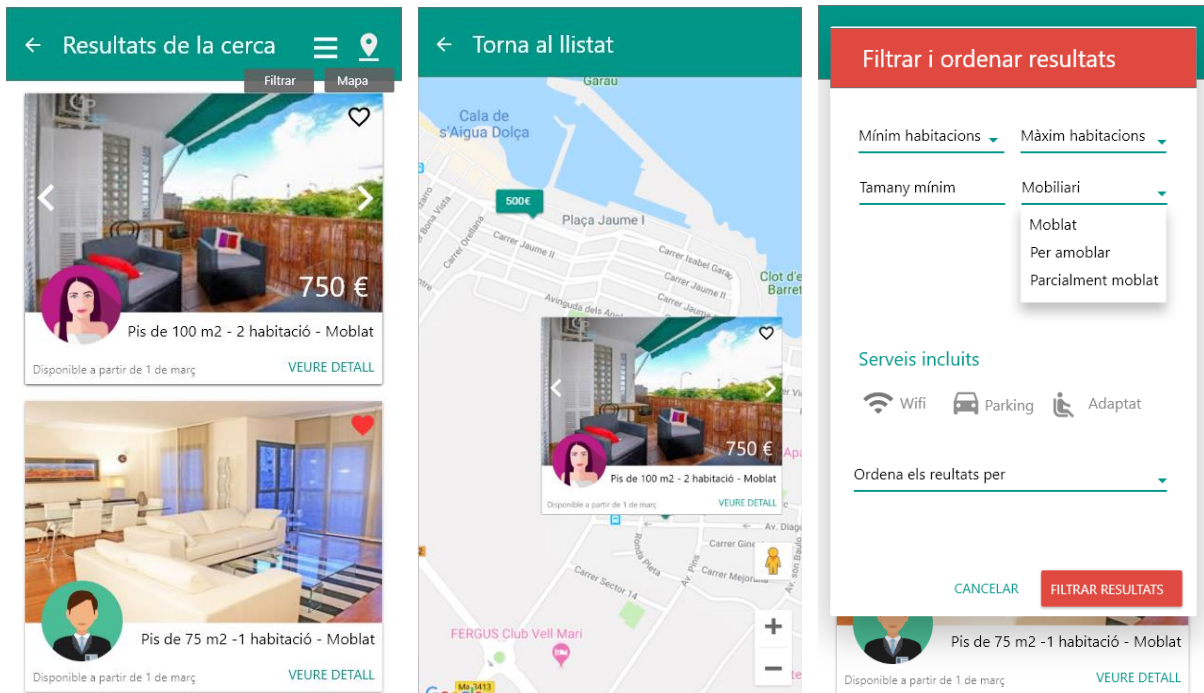
- MSG_001:** Visualitzar el llistat de converses (PRO, INQ)
- MSG_002:** Visualitzar els missatges d'una conversa (PRO, INQ)
- MSG_003:** Enviar un missatge privat a un altre usuari (PRO, INQ)
- MSG_004:** Consultar l'anunci al que (opcionalment) fa referència una conversa (PRO, INQ)

5.6. Gestió de perfil d'usuari



- GPFU_001:** Gestió del perfil personal: modificació i consulta (PRO, INQ)
- GPFU_002:** Visualitzar el perfil públic d'un usuari (PRO, INQ)

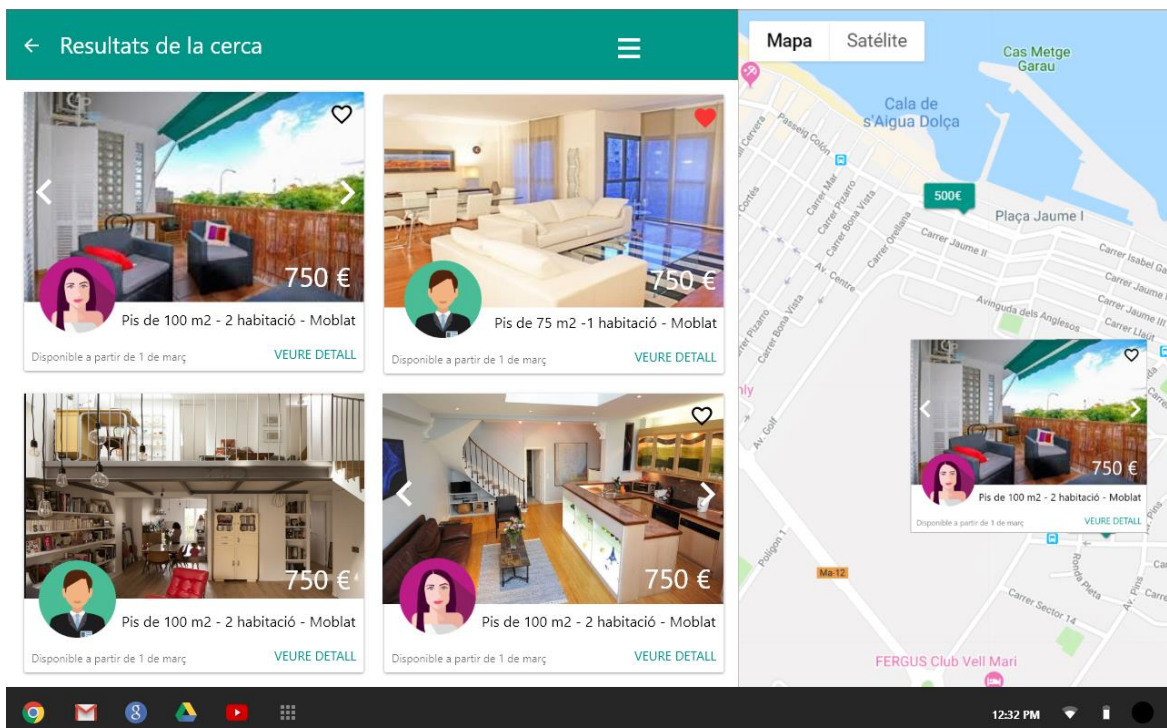
5.7. Motor de disponibilitat



MDIS_001 vista mòbil: Cercar els anuncis d'oferta d'habitatge publicats en una zona (INQ)

MDIS_002: Filtrar i ordenar els resultats d'una cerca d'anuncis d'oferta d'habitatge (INQ)

MDIS_005: Marcar / desmarcar un anunci com a favorit (PRO, INQ)



MDIS_001 vista escriptori: Cercar els anuncis d'oferta d'habitatge publicats en una zona (INQ)

MDIS_005: Marcar / desmarcar un anunci com a favorit (PRO, INQ)

← Apartament cèntric



750 €



Elena com a propietari
3 referències: 2 positives i una negativa

Exce-lent apartament cèntric, a punt per entrar a viure ...

Pis de 100 m2 - 2 habitació - 1 bany - Moblat

Wifi Parking Adaptat

Disponible a partir de 1 de març

CONTACTAR



← Anuncis favorits



Pis de 75 m2 -1 habitació - Moblat

Disponible a partir de 1 de març [VEURE DETALL](#)



MDIS_003: Consultar la informació detallada d'un anunci d'oferta d'habitatge (INQ)
MDIS_006: Consultar anuncis favorits (PRO, INQ)

6. Disseny

6.1. Disseny de la bases de dades de grafs

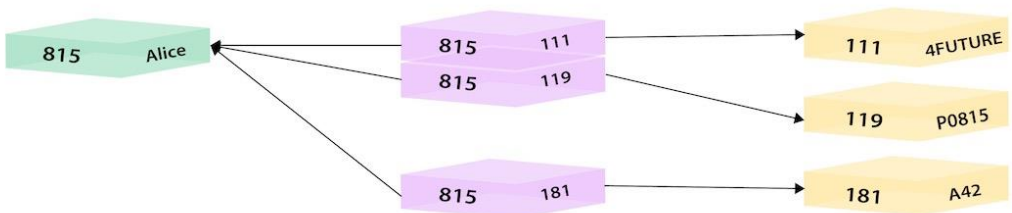
6.1.1. Introducció al modelat de bases de dades de grafs

Al contrari que en les bases de dades relacionals, en les bases de dades de grafs la informació se representa com els nodes d'un graf i les seves relacions com les arestes del mateix. Així, se pot emprar teoria de grafs per recórrer la BBDD, ja que aquesta pot descriure atributs dels nodes (entitats) i les arestes (relacions).
[1]

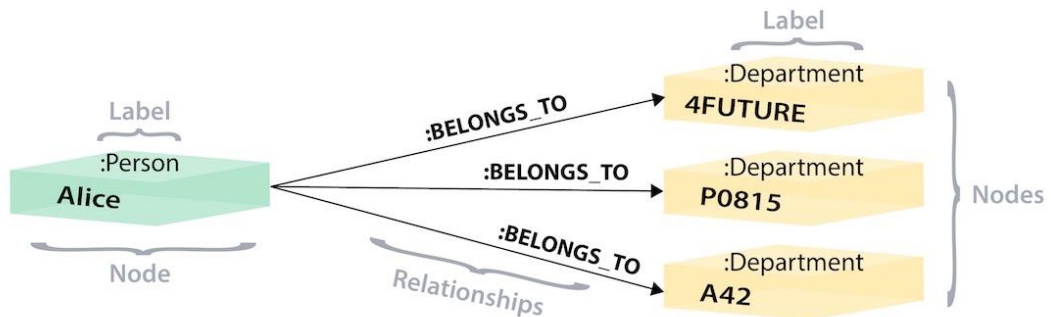
Al igual que existeixen diferents RDBMS per crear, administrar i consultar bases de dades relacionals (Oracle, Mysql, PostgreSQL, etc), també hi ha diferents opcions per les bases de dades de grafs (AllegroGraph, OrientDB, Neo4j, etc). En aquest TFG s'ha decidit emprar **Neo4j**, un motor de bases de dades de grafs open source implementat en java.

Neo4j disposa del llenguatge declaratiu **Cypher** per mantenir i fer consultes a la BBDD. Seria l'equivalent a SQL en les bases de dades relacionals.

A mode d'exemple, farem una senzilla comparativa amb una típica relació N-N relacional entre persona i departament (una persona treballa en N departaments i en un departament hi treballen N persones), indicant com se representa en una base de dades relacional i com en una de grafs.

RELACIONAL: Una persona (Alice) pertany a 3 departaments (4FUTURE, P0815 i A42)			
			
TAULES	<i>PERSON</i>	<i>PERSON_DEPARTAMENT</i>	<i>DEPARTAMENT</i>
SQL	Llista les persones que treballen en el departament "4 FUTURE"		
<pre>SELECT name FROM Person LEFT JOIN Person_Department ON Person.Id = Person_Department.PersonId LEFT JOIN Department ON Department.Id = Person_Department.DepartmentId WHERE Department.name = "4 FUTURE"</pre>			

GRAFS: Una persona (Alice) pertany a 3 departaments (4FUTURE, P0815 i A42)



NODES I RELACIONS

PERSON *:BELONGS_TO* → *DEPARTAMENT*

CYPHER

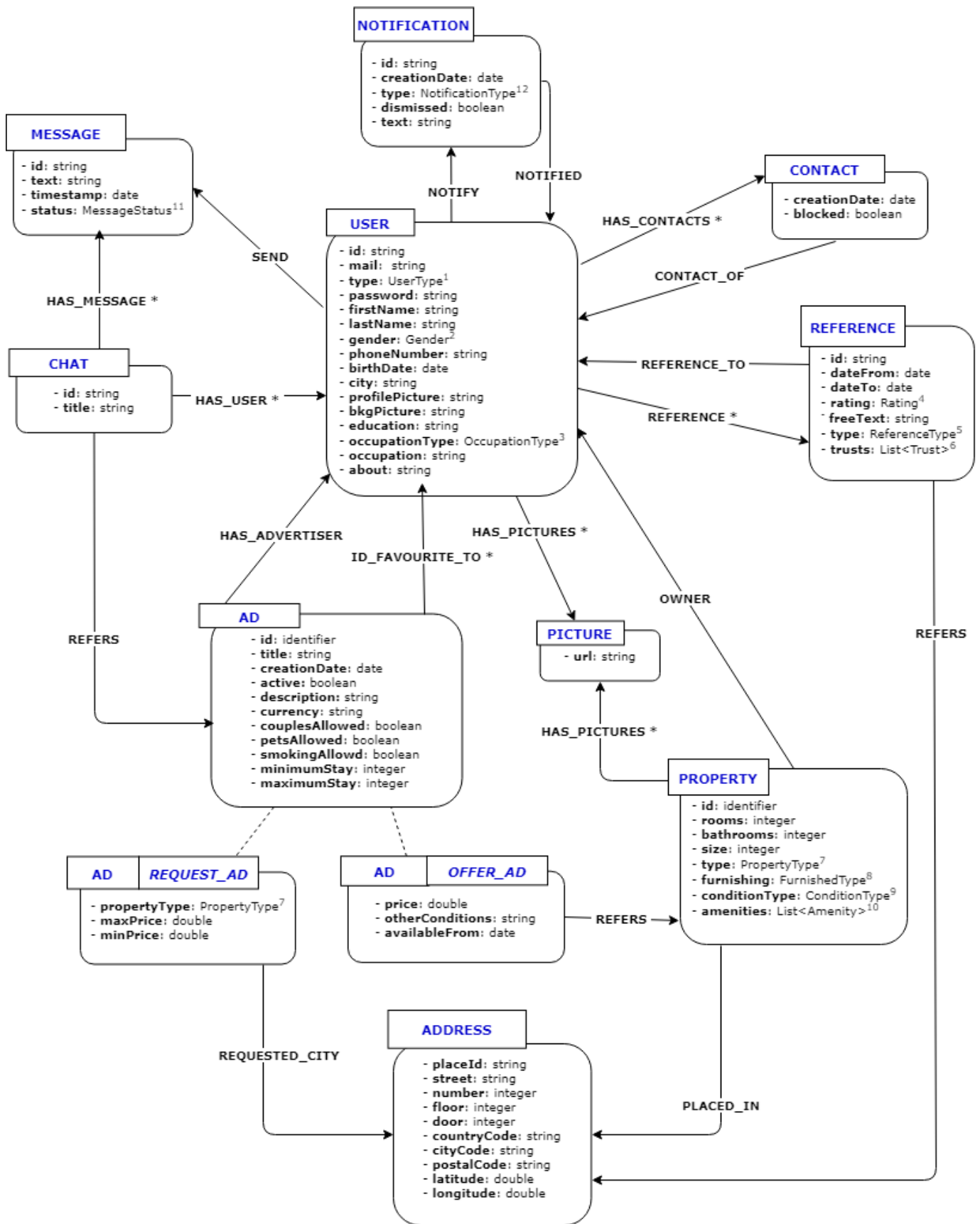
Llista les persones que treballen en departament "4 FUTURE"

```
MATCH (p:Person)-[:BELONGS_TO]->(d:Dept)
WHERE d.name = "4FUTURE"
RETURN p.name
```

Una de les particularitats de modelar bases de dades de grafs és que les relacions entre nodes són molt flexibles, ja que la direcció d'una relació entre dos nodes no és més que un concepte de disseny. Així, seguint amb l'exemple anterior, seria igualment vàlid definir una relació entre Person i Departament en sentit invers: Per exemple *DEPARTMENT :HAS_PERSONS -> PERSON*. El model seria igualment vàlid, només canviarien les consultes Cypher per recórrer el graf. **[2]**

Destacar també que a un mateix node se li poden assignar N etiquetes, no està limitat a una. Això pot ser molt útil per exemple a l'hora de modelar herències entre nodes (veure nodes *RequestAd* i *OfferAd* del model proposat en el següent punt)

6.1.2. Model de grafs

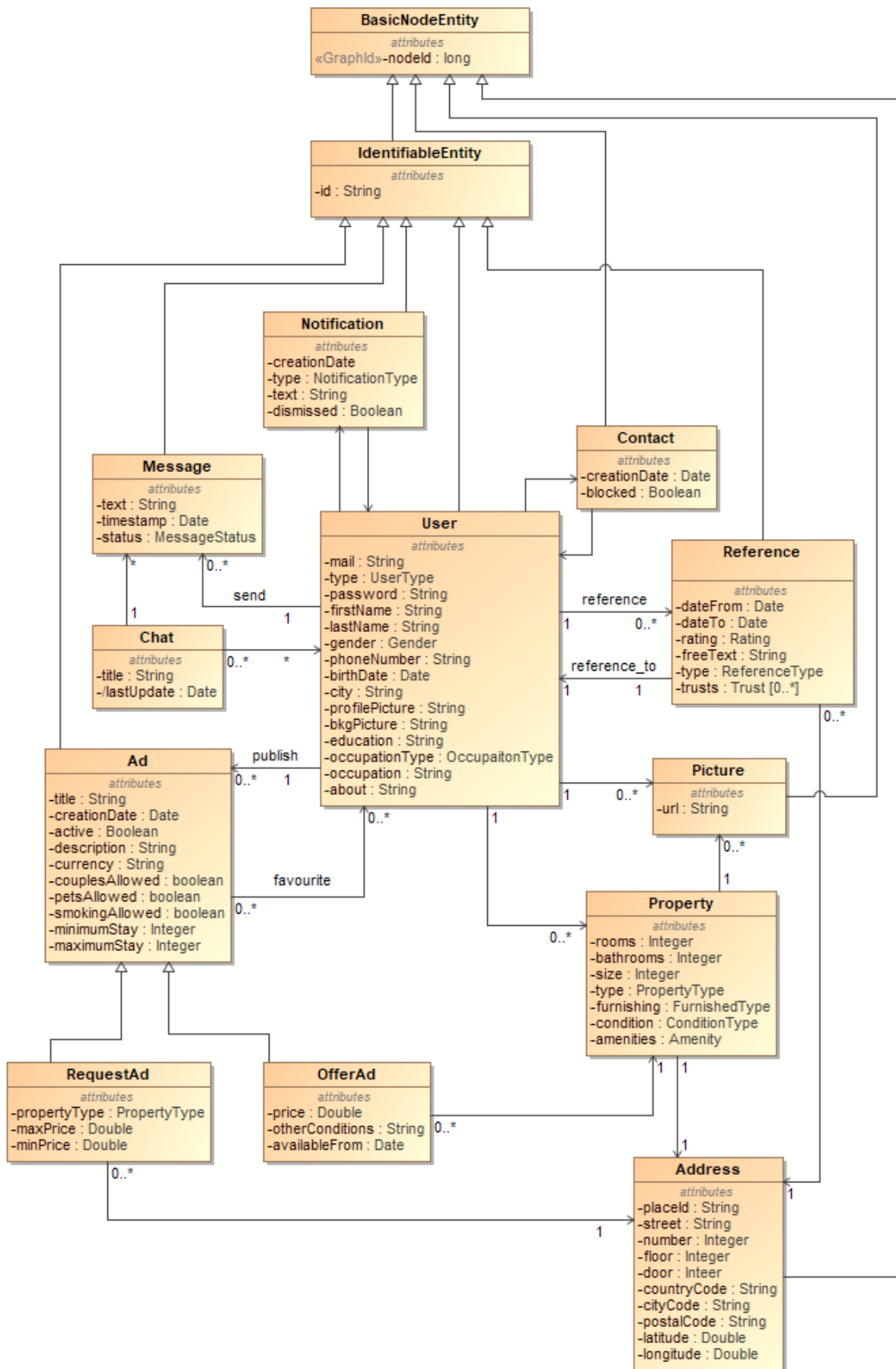


Annex amb els tipus enumerats propis utilitzats en el model de grafs:

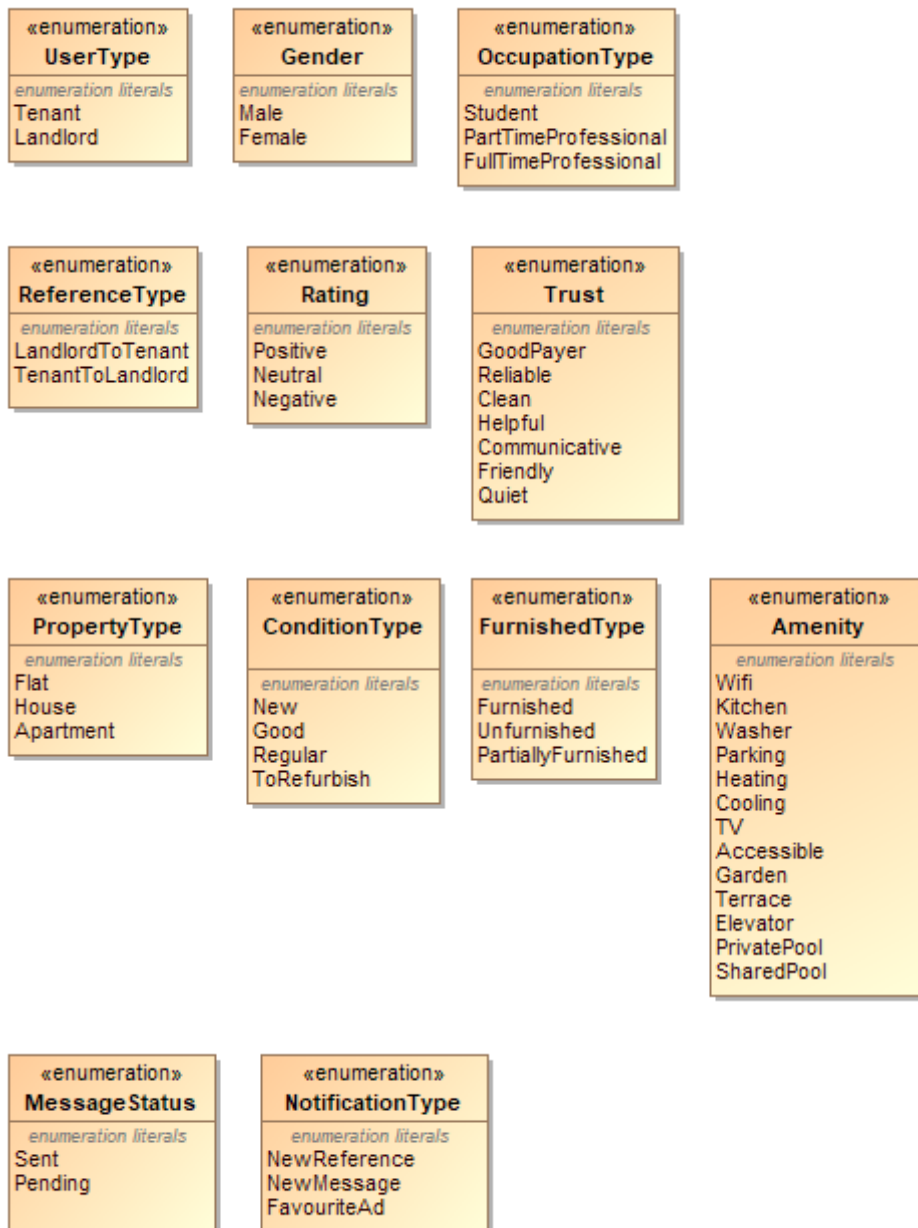
Enumeració	Descripció	Possibles valors
¹ UserType	Tipus d'usuari	<i>Tenant Landlord</i>
² Gender	Gènere de la persona.	<i>Male Female</i>
³ OccupationType	Situació laboral actual	<i>Student PartTimeProfessional FullTimeProfessional</i>
⁴ Rating	Valoració de la referència	<i>Positive Neutral Negative</i>
⁵ ReferenceType	Tipus de referència.	<i>TenantToLandlord LandlordToTenant</i>
⁶ Trust	Qualitat especial a destacar	<i>GoodPayer Reliable Clean Helpful Communicative Friendly Quiet</i>
⁷ PropertyType	Tipus de propietat	<i>Flat House Apartment</i>
⁸ FurnishedType	Tipus de moblat	<i>Furnished Unfurnished PartiallyFurnished</i>
⁹ ConditionType	Estat de la propietat	<i>New Good Regular ToRefurbish</i>
¹⁰ Amenity	Servei inclòs en la propietat	<i>Wifi Kitchen Washer Parking Heating Cooling TV Accessible Garden Terrace Elevator Private_Pool Shared_Pool</i>
¹¹ MessageStatus	Estat del missatge	<i>Sent Pending</i>
¹² NotificationType	Tipus de notificació	<i>NewReference NewMessage FavouriteAd</i>

5.2. Diagrama de classes

Consideracions: Com ja s'ha indicat, s'emprarà la BBDD de grafs *Neo4j* per mantenir les dades i el framework *spring-data-neo4j* en la capa de persistència. Aquest mòdul necessita que tots els nodes tinguin un atribut *@GraphId*, que s'empra internament per identificar cada node dins el graf de manera inequívoca. Per aquest motiu totes les entitats del diagrama hereten de la classe *BasicNodeEntity*, la qual conté aquest atribut. Cal destacar que aquest atribut no es pot utilitzar com un *id* a nivell lògic. Amb aquesta finalitat, s'ha definit una segona classe: *IdentifiableEntity*, la qual sí conté l'atribut *id*. Totes les entitats principals que requereixen manteniment heretaran aquesta classe. [11]

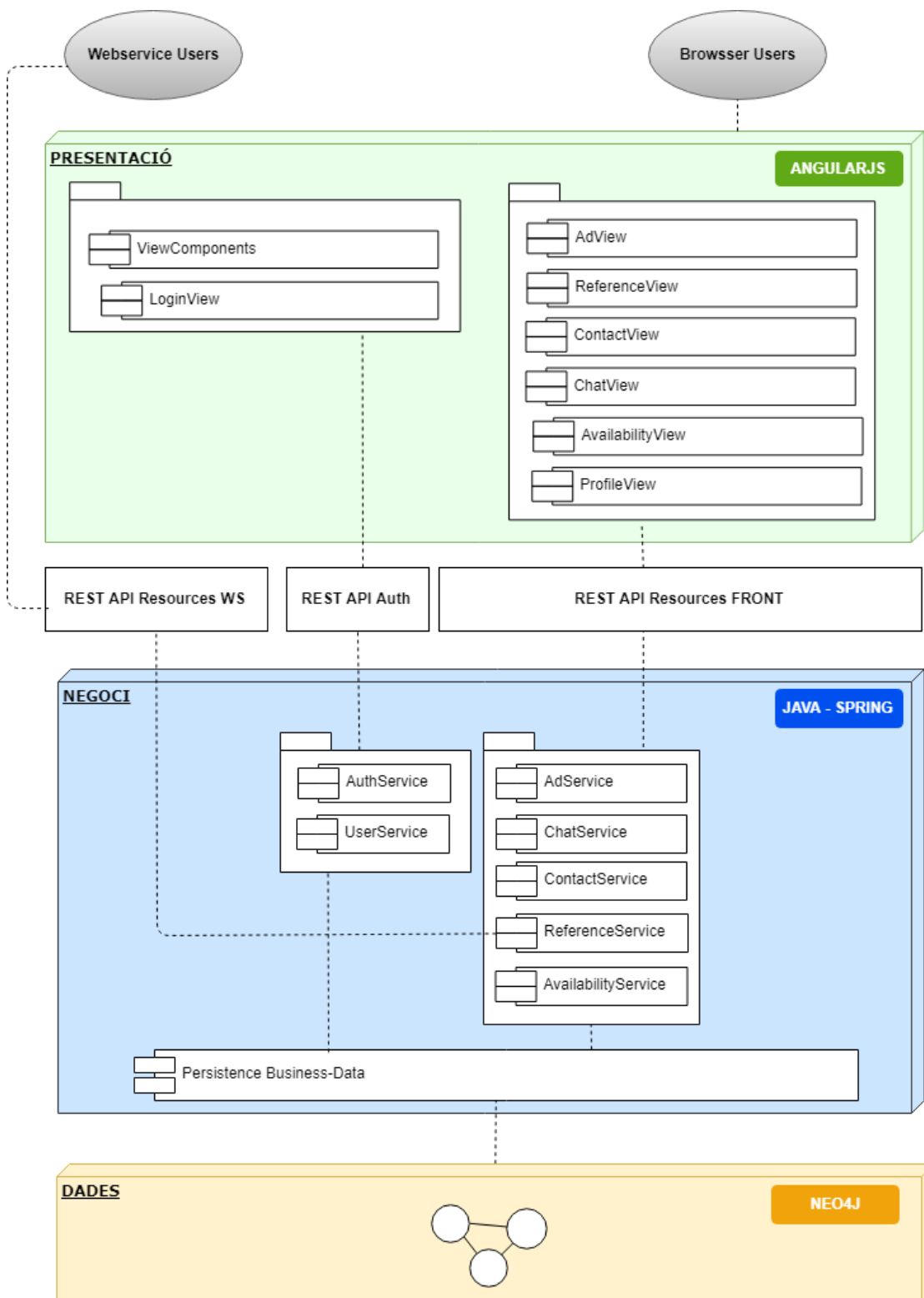


Annex amb les enumeracions pròpies utilitzades en el diagrama de classes UML:



6.3. Arquitectura

6.3.1. Diagrama d'alt nivell (capes)



L'aplicació s'estructura seguint la típica arquitectura de 3 capes: presentació, negoci i dades.

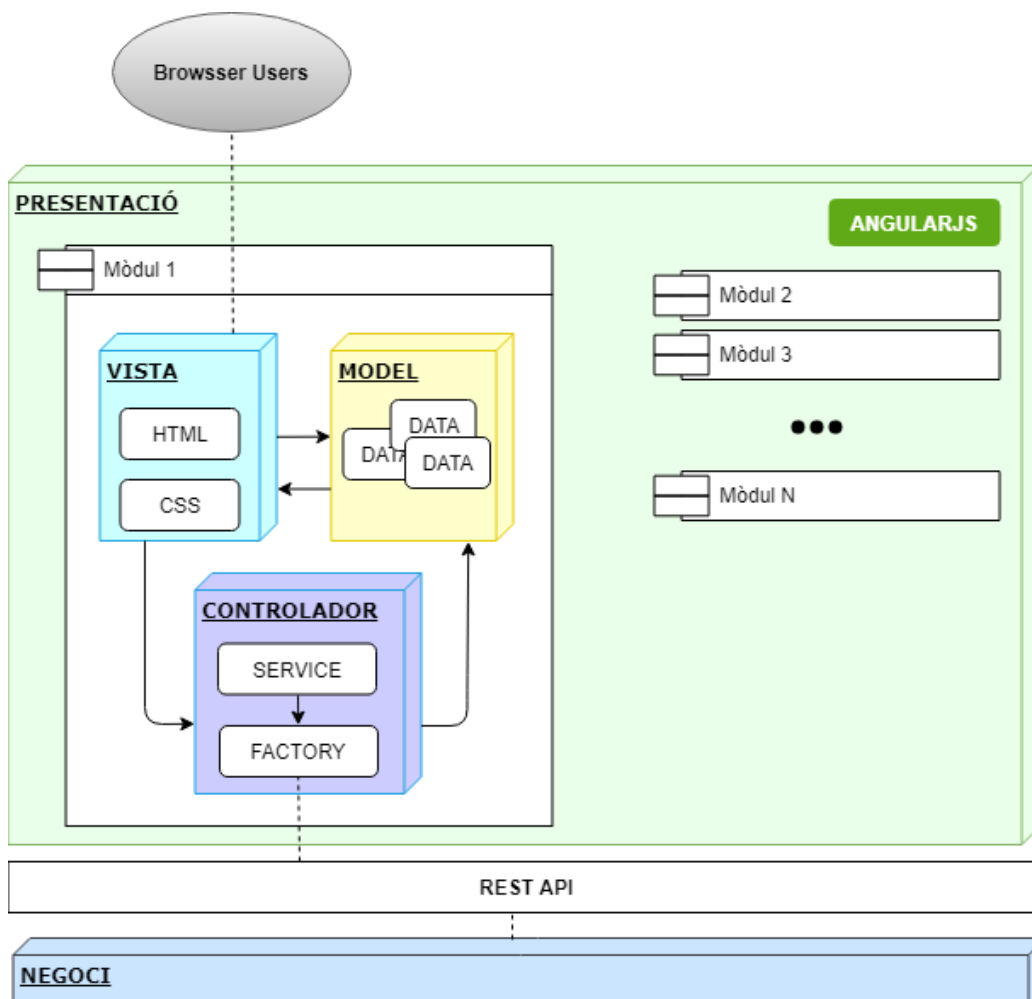
6.3.2. Arquitectura front (capa de presentació)

AngularJS segueix una típica arquitectura MVC (model-vista-controlador), però implementada en la part client (browser).

Un dels aspectes més rellevants de AngularJS és la modularitat. A mode resum, una aplicació es compon de diferents mòduls, totalment independents entre sí, de manera que cadascun té la seva vista, model i controlador propis. Així, si ho trasllem al nostre diagrama d'arquitectura, cada component (*AdView*, *ReferenceView*, etc) equivaldria a un mòdul angular.

Cal fer una especial menció a la característica *two-way data binding*. [14] AngularJS manté la vista i el model sincronitzats, de forma que quan canvia una dada del model, el canvi es reflexa en la vista automàticament. Aquest és un dels principals avantatges enfront tecnologies web anteriors com per exemple JSF, on quan canvia una dada del model s'ha d'indicar expressament quines seccions de la vista s'han d'actualitzar perquè es poden veure afectades pel canvi.

Imaginem que fem zoom a la capa de presentació del diagrama d'arquitectura d'alt nivell definit en el punt 6.3.1, internament la capa de presentació segueix la següent arquitectura:

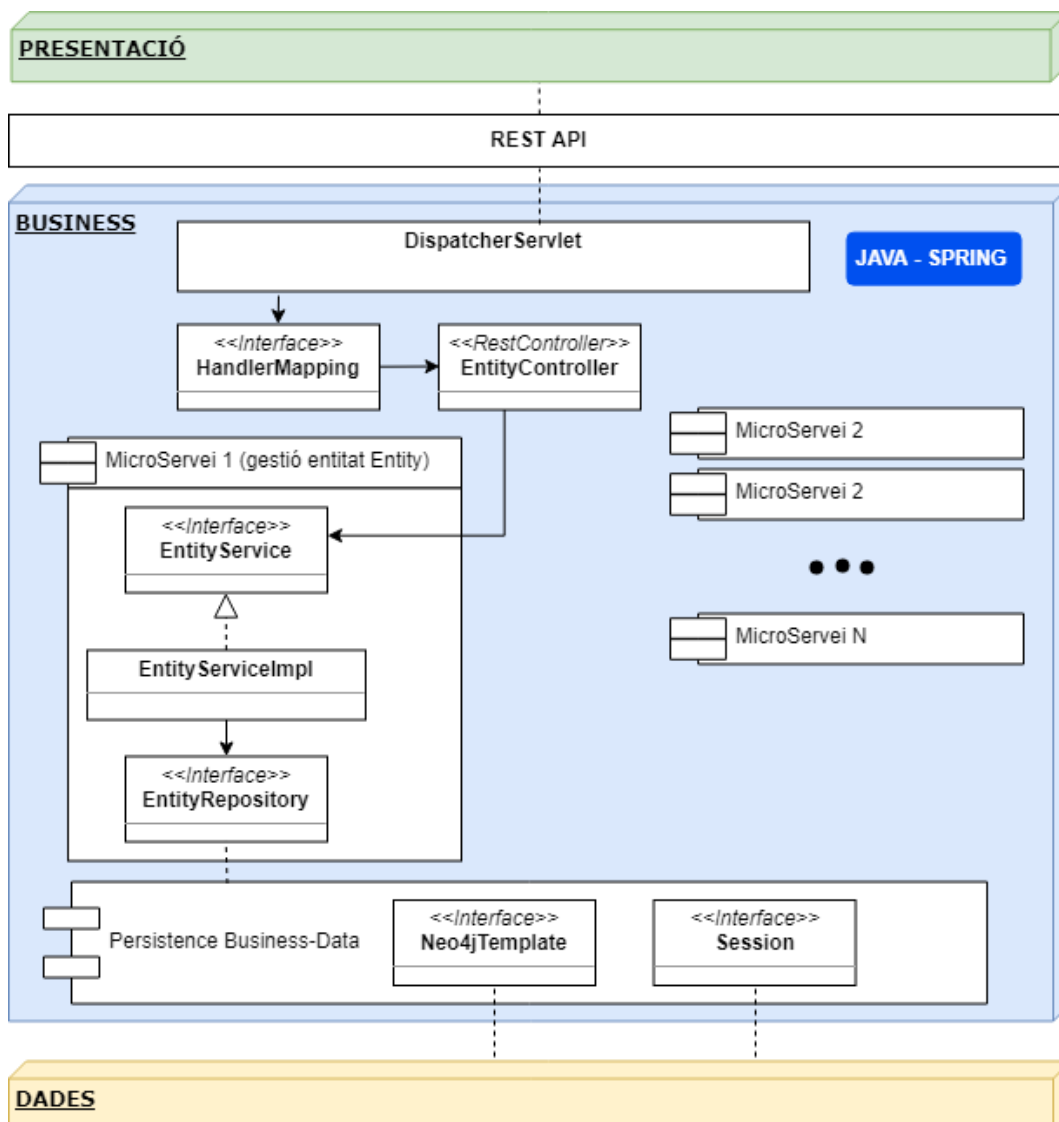


6.3.3. Arquitectura back (capa de negoci)

Tal com hem indicat, tot el back està desenvolupat íntegrament en Java 1.8 sobre l'ecosistema spring seguint una arquitectura orientada a microserveis (si ho traslладem al nostre diagrama d'arquitectura, cada component de la capa de negoci (*AdService*, *ReferenceService*, etc) equivaldria a un microservei java-spring)

- Spring-core: Per la gestió dels microserveis.
- Spring-mvc: Per exposar la *API REST JSON* amb les funcionalitats necessàries perquè el front pugui visualitzar i mantenir les dades, així com la API destinada als usuaris webservice.
- Spring-data-neo4j: Per la persistència amb la base de dades de grafs Neo4j

El back també segueix una arquitectura MVC (model-vista-controlador) a l'hora de publicar les APIS REST JSON.



7. Implementació

7.1 Projectes implementats

Amb la finalitat de tenir la vista totalment desacoblada de la resta de capes i fer el projecte totalment moduable, s'ha optat per crear dos projectes totalment independents. Així, l'aplicatiu consta dels projectes:

- **tfg-front:** Capa de presentació. Client web desenvolupat íntegrament en *AngularJS* amb tota la interfície d'usuari de la plataforma mobile-first. Dialoga amb el back a través d'una *API REST JSON*.
- **tfg-back:** Capa de negoci. Back desenvolupat en *Java i Spring* que conté tota la lògica de negoci. Per la persistència amb la base de dades Neo4j fem ús del framework open source *spring-data-neo4j*. El back exposa una API REST amb les diferents funcionalitats necessàries perquè des del front es puguin visualitzar i mantenir les dades. Per la publicació de l'API es fa ús del framework *spring-mvc*. Internament, aquest mòdul s'estructura seguint una arquitectura orientada a *microserveis*, de manera que cada servei està aïllat de la resta.
- **BBDD de grafs Neo4j:** Capa de dades. Conté la informació de totes les entitats i relacions gestionades.

A continuació, anem a aprofundir una mica més en com s'estructura internament cadascun dels mòduls.

7.1.1 tfg-front

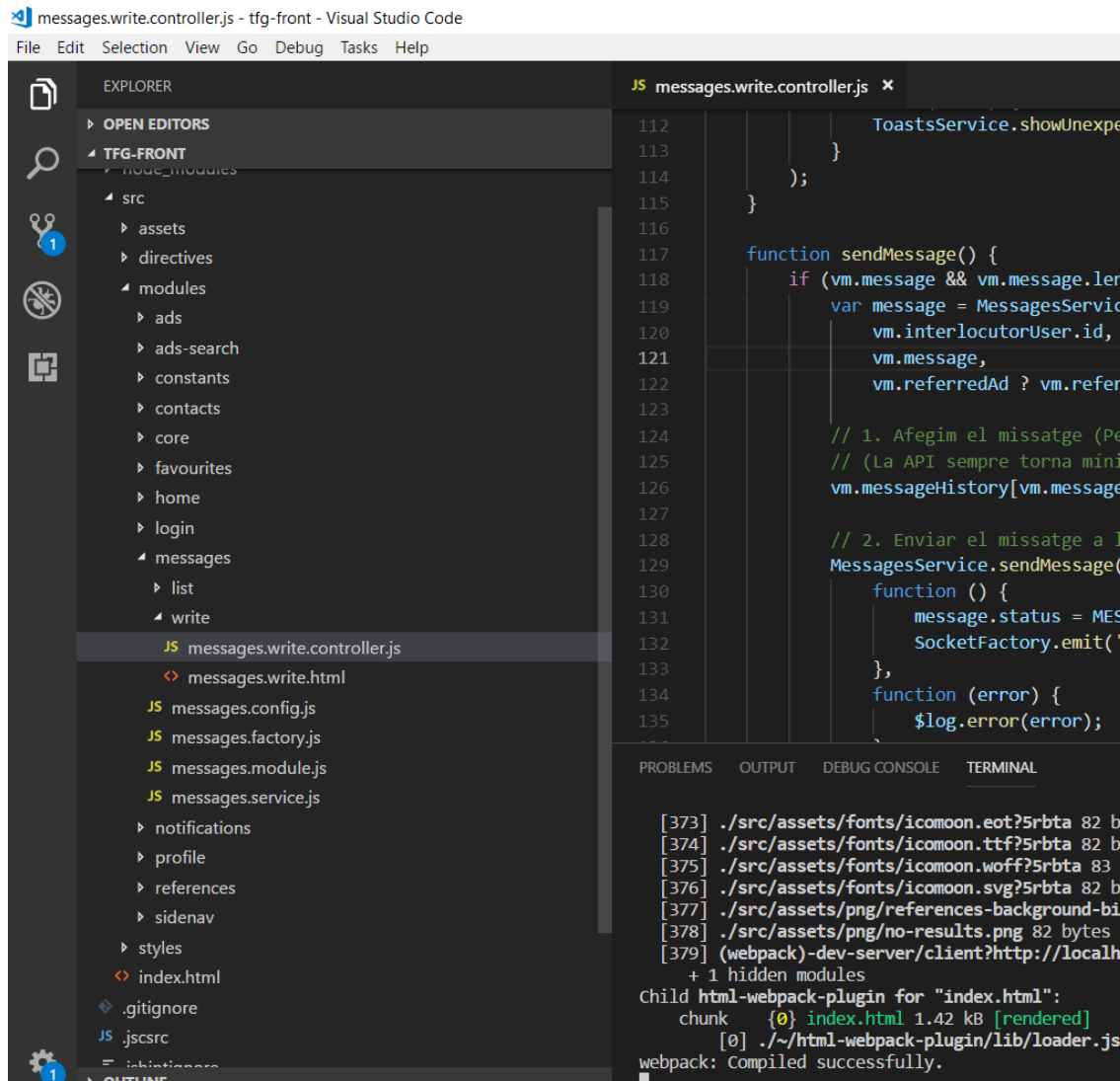
Codi font complet disponible a:

- <https://bitbucket.org/troig-uoc/tfg-front/src/master>

A nivell estructural, el projecte es divideix en els següents blocs:

- **/assets:** Recursos estàtics tals com imatges, fonts, svg, etc.
- **/directives:** Directori on agrupem les diferents directives angular emprades en el projecte. Podem entendre una directiva angular com un component web independent i per tant reaprofitable en diferents mòduls. Per exemple, la directiva *avatar* és la que utilitzem en tots els mòduls de l'aplicatiu per visualitzar l'avatar de l'usuari amb la seva foto de perfil en miniatura.
- **/modules:** Directori que engloba els diferents mòduls de la plataforma. Un dels principals avantatges de AngularJS és la modularitat. Hem aprofitat aquesta característica i tenim diferents mòduls per les seccions del frontal o per la gestió de diferents entitats. Exemple: ads, contacts, messages, profile...
- **/styles:** Fulles d'estil CSS.

Pel desenvolupament del projecte tfg-front he emprat l'editor de codi obert **Visual Studio Code**.



No entrarem en detall en el codi font generat (es pot consultar online en la seva totalitat), però sí que cal fer menció a com s'estructura cadascun dels mòduls a nivell general. Si agafem com a model el mòdul *contacts*, tenim que els fitxers més rellevants, i comuns a la resta de mòduls, són:

- **contacts.config.js**: Configuració del mòdul, on es defineixen les rutes i el flux de navegació del mateix.
- **contacts.module.js**: Fitxer javascript que centralitza la creació del mòdul, indicant el seu fitxer de configuració, controladors, serveis i factories utilitzats.
- **contacts.controler.js**: Controlador encarregat d'actualitzar la vista a partir del model en l'arquitectura MVC que segueix AngularJS.
- **contacts.service.js**: Servei amb les diferents funcionalitats exportables de mòdul. AngularJS suporta injecció de dependències, fet que faria possible injectar el servei de contactes en un altre mòdul i així poder reaprofitar tota la seva lògica.

- **contacts.factory.js**: Capa encarregada de dialogar amb les APIs que exposa el tfg-back.
- **contacts.html**: Html amb la vista.

7.1.2 tfg-back

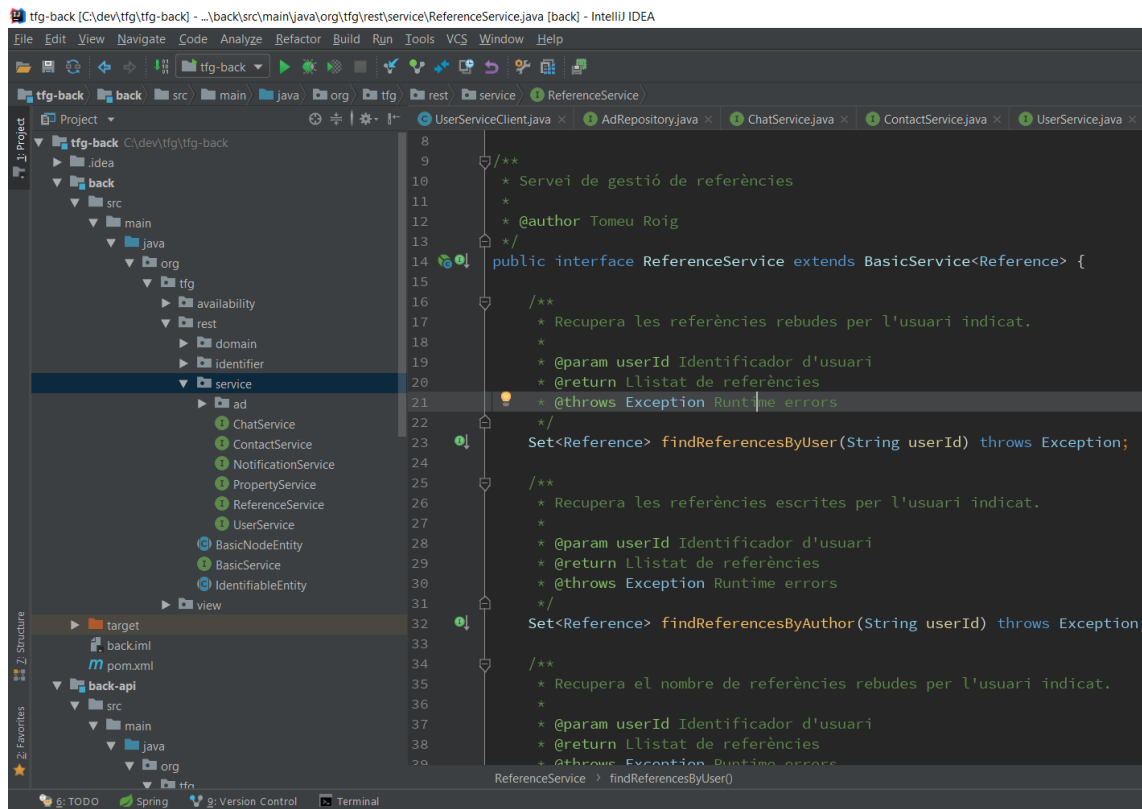
Codi font complet disponible a:

- <https://bitbucket.org/troig-uoc/tfg-back/src/master>

A nivell estructural, el projecte es divideix en els següents blocs principals:

- **/back**: Mòdul únicament d'especificacions on s'inclouen les interfaces dels diferents serveis del projecte, les classes que defineixen el domini de dades així com les classes que defineixen el domini de vista.
 - **/rest/domain**: Classes del model de dades: `Ad`, `Address`, `Chat`, `Contact`, etc.
 - **/rest/service**: Interfaces dels diferents serveis amb la lògica de negoci: `ChatService`, `ContactService`, `AdService`..
 - **/view**: Classes del domini adaptat a la vista. `ChatView`, `MessageView`, etc.
- **/back-api**: Implementació dels diferents serveis i controladors per exposar una API REST JSON a fi de que les diferents entitats es puguin gestionar a través del front i del webservice.
 - **/rest/controller**: Controladors que exposen els diferents endpoints per accedir a l'API de recursos: `AdController`, `ContactController`, etc.
 - **/rest/repository**: Repositoris d'accés a la capa de dades. Emprem `spring-data-neo4j` per accedir i persistir la capa de dades. És per això que tots els repositoris estenen de la classe `GraphRepository<T>`
 - **/rest/service**: Lògica de negoci amb la implementació de les interfaces des diferents serveis definides a `/back/rest/service`: `ChatServiceImpl`, `ContactServiceImpl`, etc.
 - **/view**: Lògica amb l'adaptació de les dades a la vista.
 - **/controller**: Controladors específics per la vista.
 - **/transformar**: Classes amb la lògica de transformació d'entitats del model de dades al domini de vista i viceversa.
- **/back-auth**: Mòdul encarregat del procés d'autenticació a través de `spring-security`. Exposa els endpoint necessaris per l'obtenció del token d'autenticació a partir de les credencials d'usuari.

Pel desenvolupament del projecte he emprat el potent IDE **IntelliJ Idea**:



A mode il·lustratiu, a continuació adjuntem un annex amb alguns exemples d'endpoints de l'API REST que publica el projecte tfg-back.

- Mòdul **back-auth**:

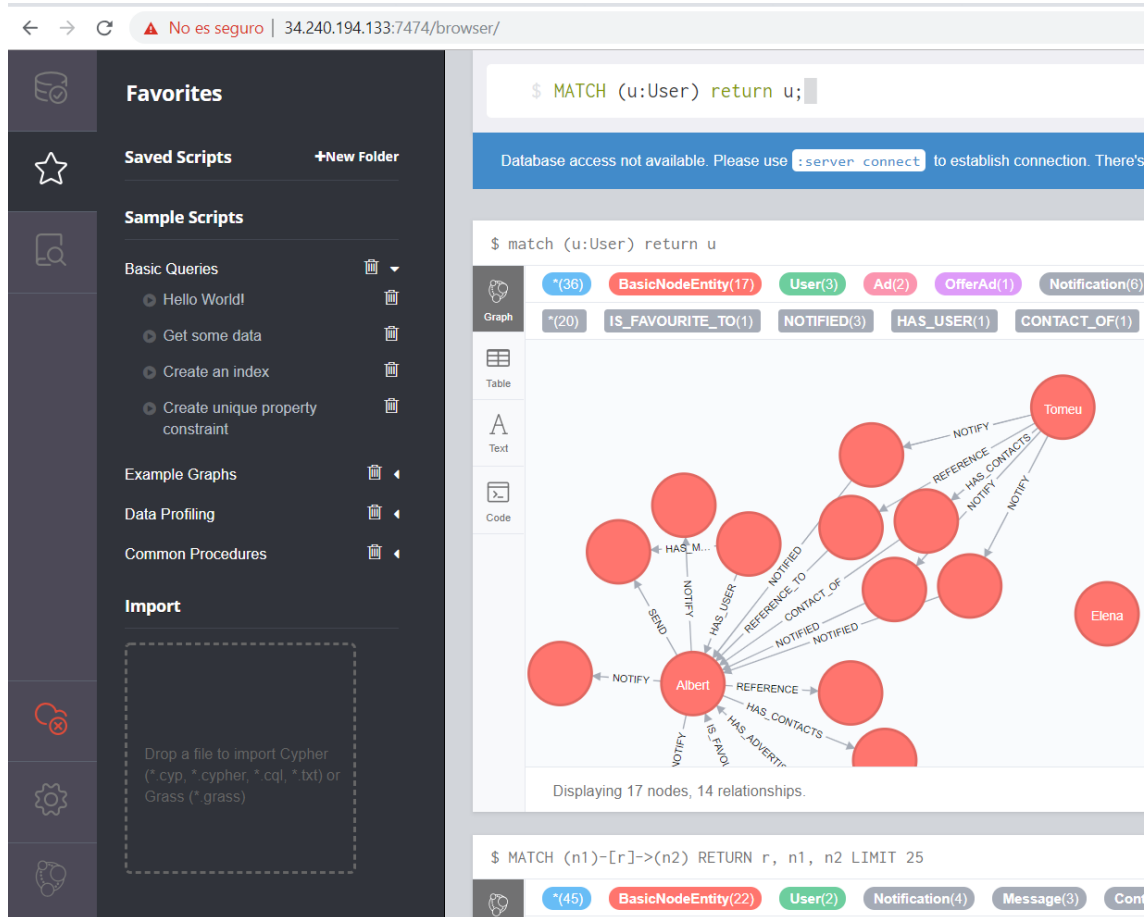
Mètode	Endpoint	Descripció
POST	/oauth/token	Endpoint per obtenir el token a partir de les credencials d'autenticació. El token resultant necessita ser inclòs com un header en totes les peticions de l'API que requereixen autenticació

- Mòdul **back-api**:

Mètode	Endpoint	Descripció
POST	/ads/offer/	Creació d'un anunci d'oferta
DELETE	/ads/offer/{adId}	Elimina l'anunci d'oferta indicat
GET	/ads/offer/{adId}	Recupera l'anunci d'oferta indicat
POST	/ads/{id}/favourite/{userId}	L'usuari afegeix un anunci a favorits
DELETE	/ads/{id}/favourite/{userId}	L'usuari esborra un anunci dels favorits
POST	/ads/{id}/disable	Desactiva l'anunci indicat
POST	/ads/{id}/enable	Activa l'anunci indicat
...

7.1.3 Base de dades de grafs Neo4j

Com ja s'ha indicat, en la capa de dades utilitzem la base de dades de grafs **Neo4j**. Neo4J disposa tant d'un client web com una aplicació d'escriptori multiplataforma per poder gestionar la base de dades. Dit aplicatiu ha estat de gran utilitat a l'hora de comprovar que les dades des de la capa de negoci a la base de dades es persistien correctament.



En la capa de persistència, hem fet ús del framework **spring-data-neo4j**, el qual ens ha facilitat enormement aquesta tasca. Destaquem la possibilitat de incloure consultes i instruccions Cypher com simples anotacions en els repositoris que estenen `GraphRepository<T>`. Així, les consultes a Neo4j s'han simplificat i aïllat de la resta de la lògica de negoci. [10]

Exemple:

```
@Query("MATCH (r:Reference)-[reference_to:REFERENCE_TO]->(u:User{id:{0}}) " +
    "MATCH (author:User)-[reference:REFERENCE]->(r) " +
    "MATCH (r)-[refers:REFERS]->(a:Address) " +
    "RETURN r, u, reference, author, refers, a")
Set<Reference> findReferenceByUser(String userId);
```

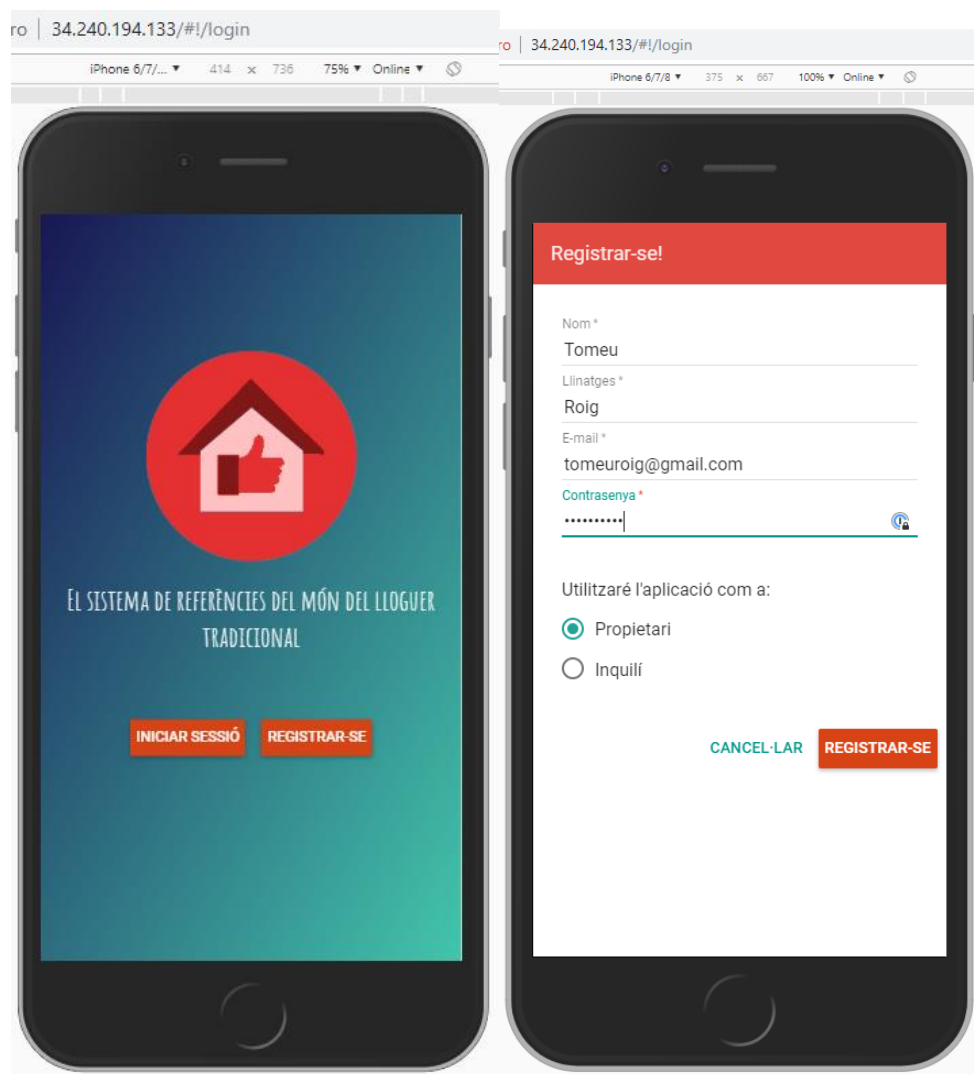
7.2 Plataforma web

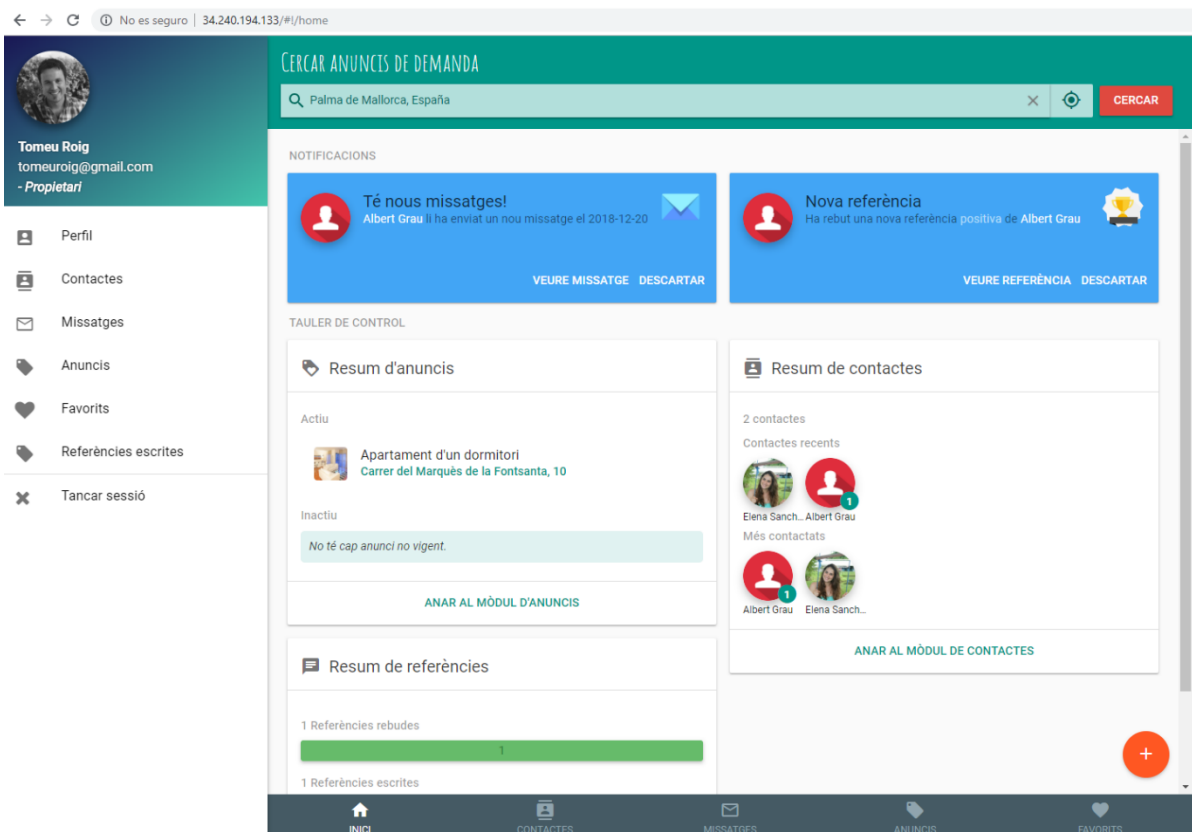
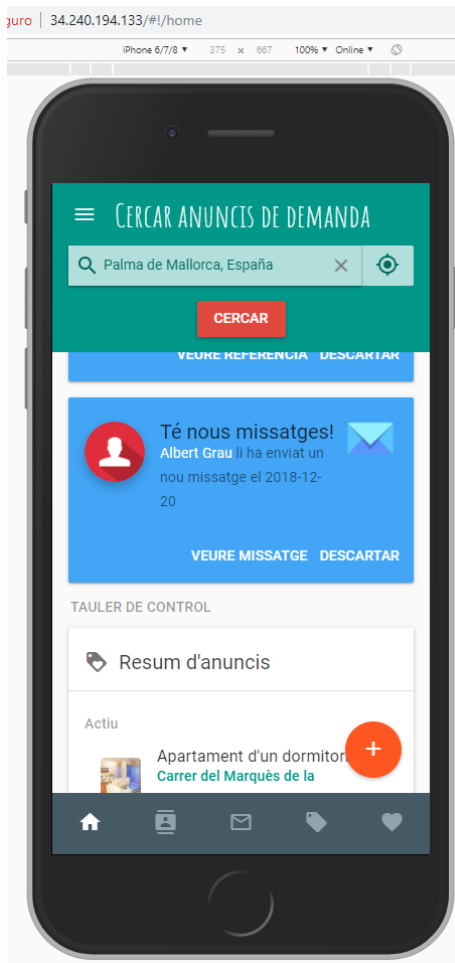
El principal producte obtingut és una plataforma web totalment funcional. Es pot accedir a l'aplicatiu en les URL:

- DNS: <http://www.troig-uoc.tk>
- IP: <http://34.240.194.133>

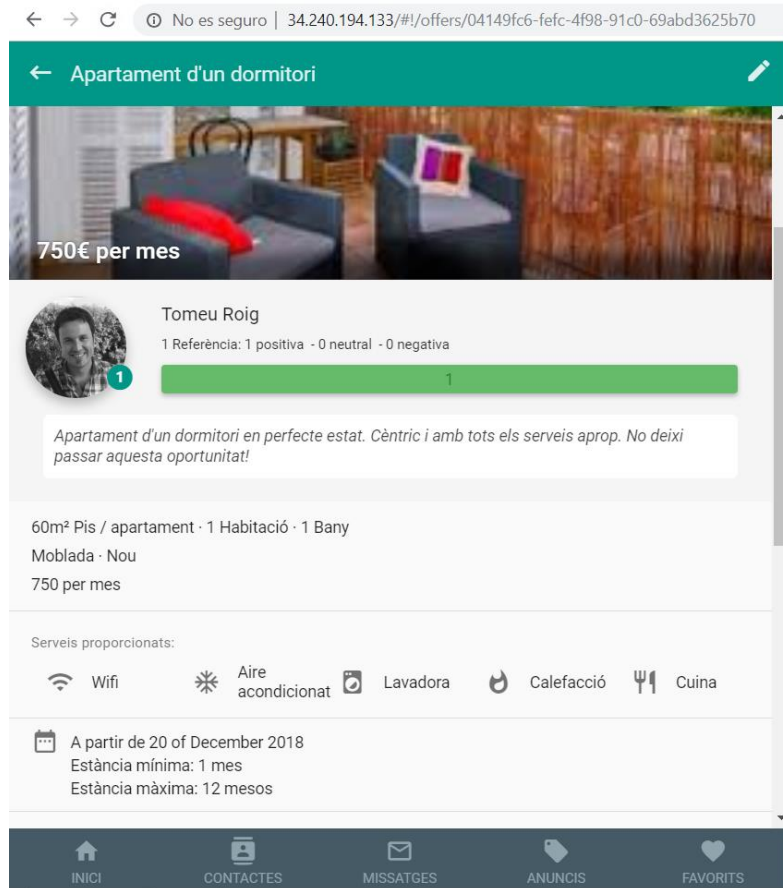
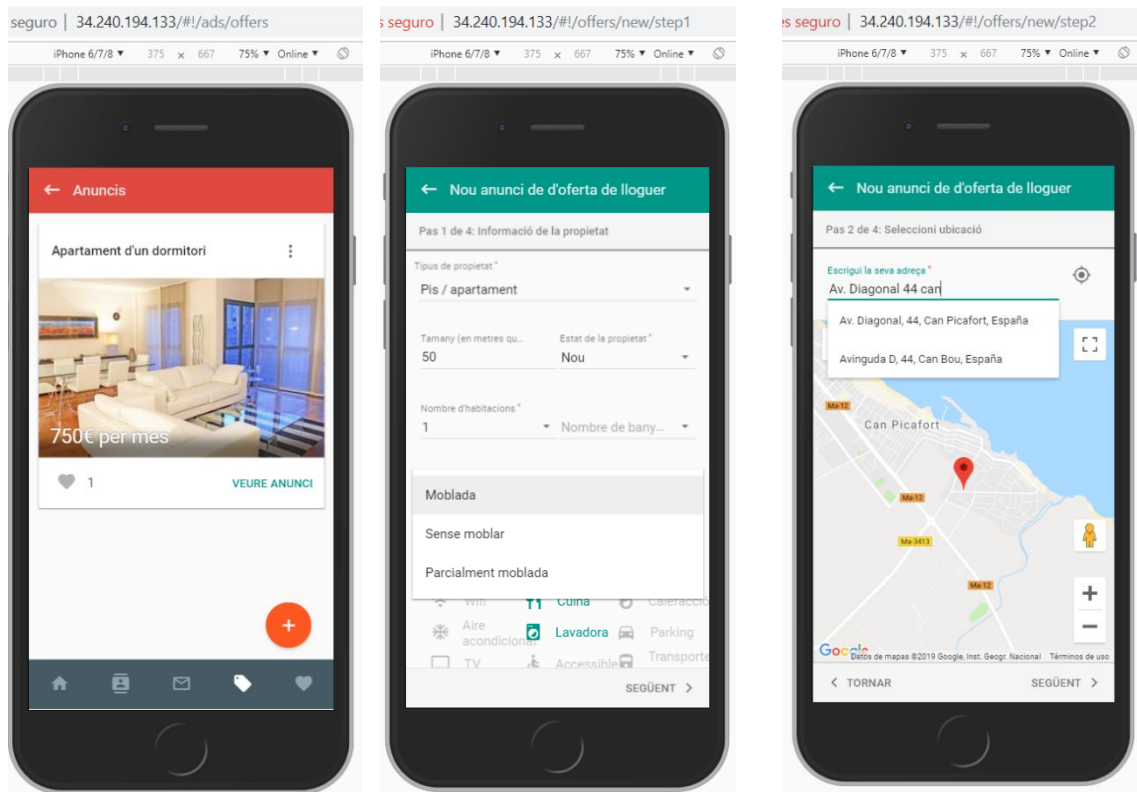
A continuació adjuntem captures de pantalla dels principals mòduls. A l'igual que en la fase de prototipat, la majoria de captures s'han fet amb la resolució de mòbil, simplement perquè al ser un aplicatiu Mobile-first, la vista s'adapta millor a resolucions d'aquest dispositiu.

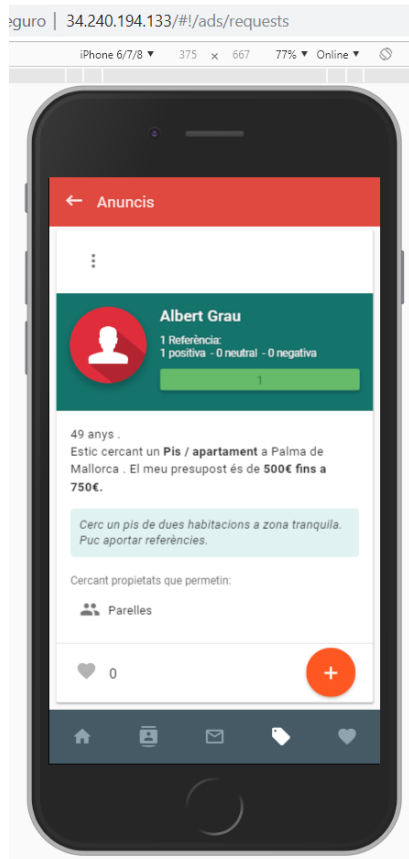
7.2.1. Registre i autenticació



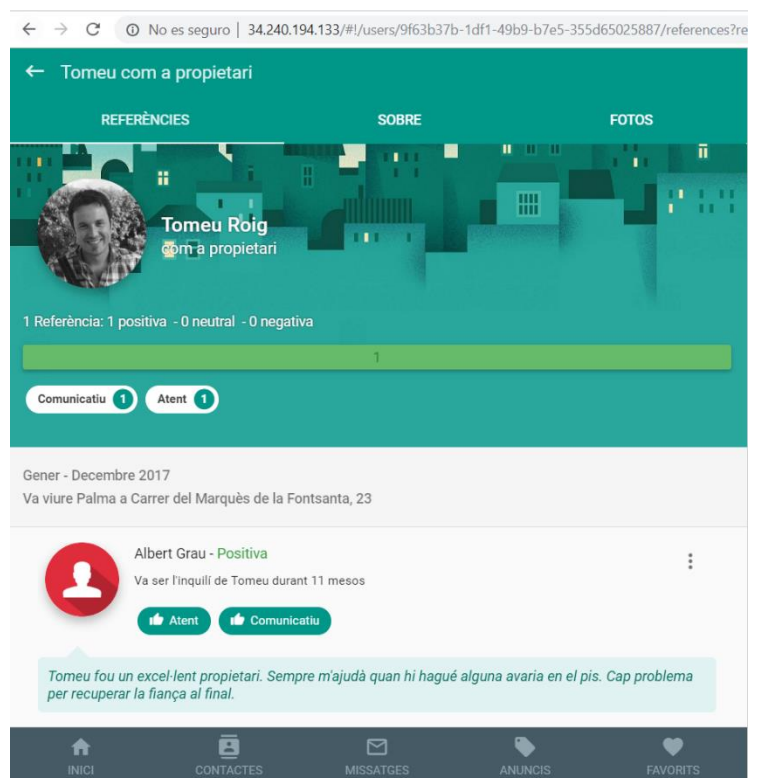


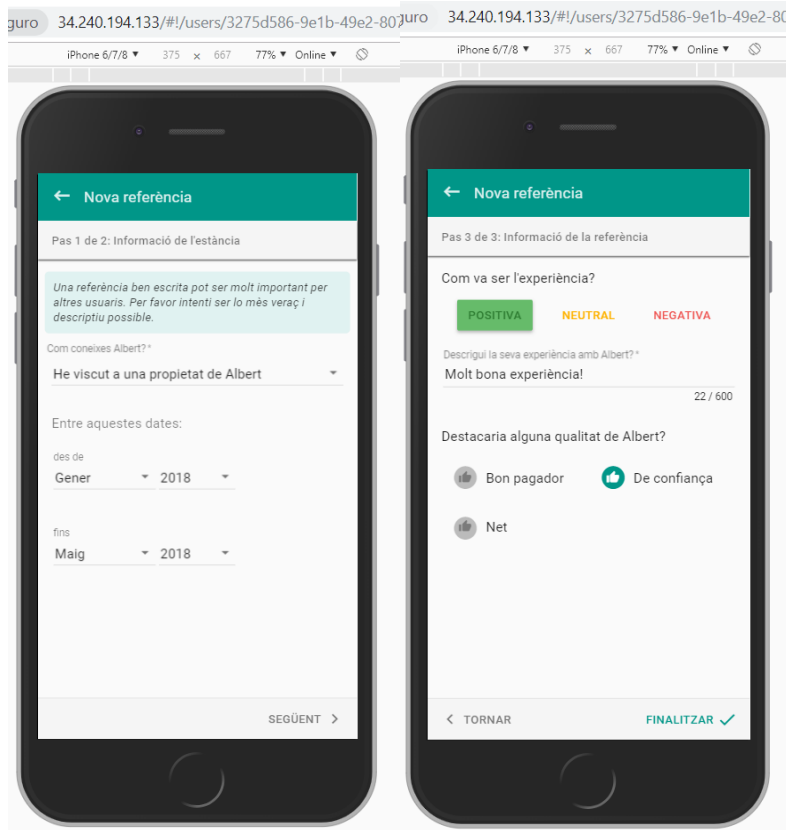
7.2.2. Gestió d'anuncis



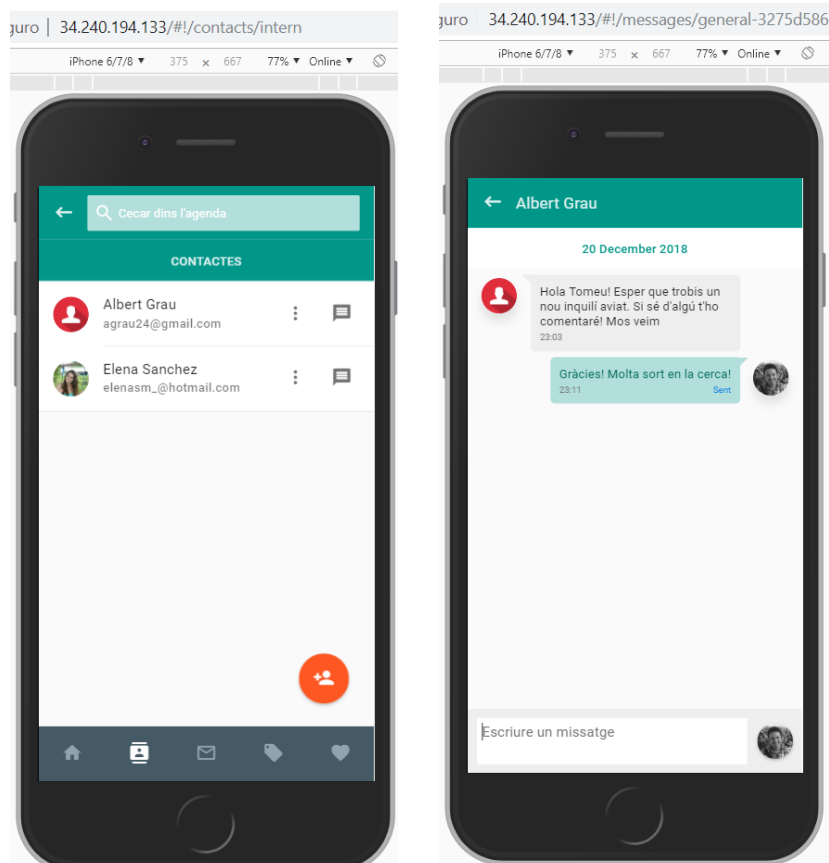


7.2.3. Gestió de referències i reputació

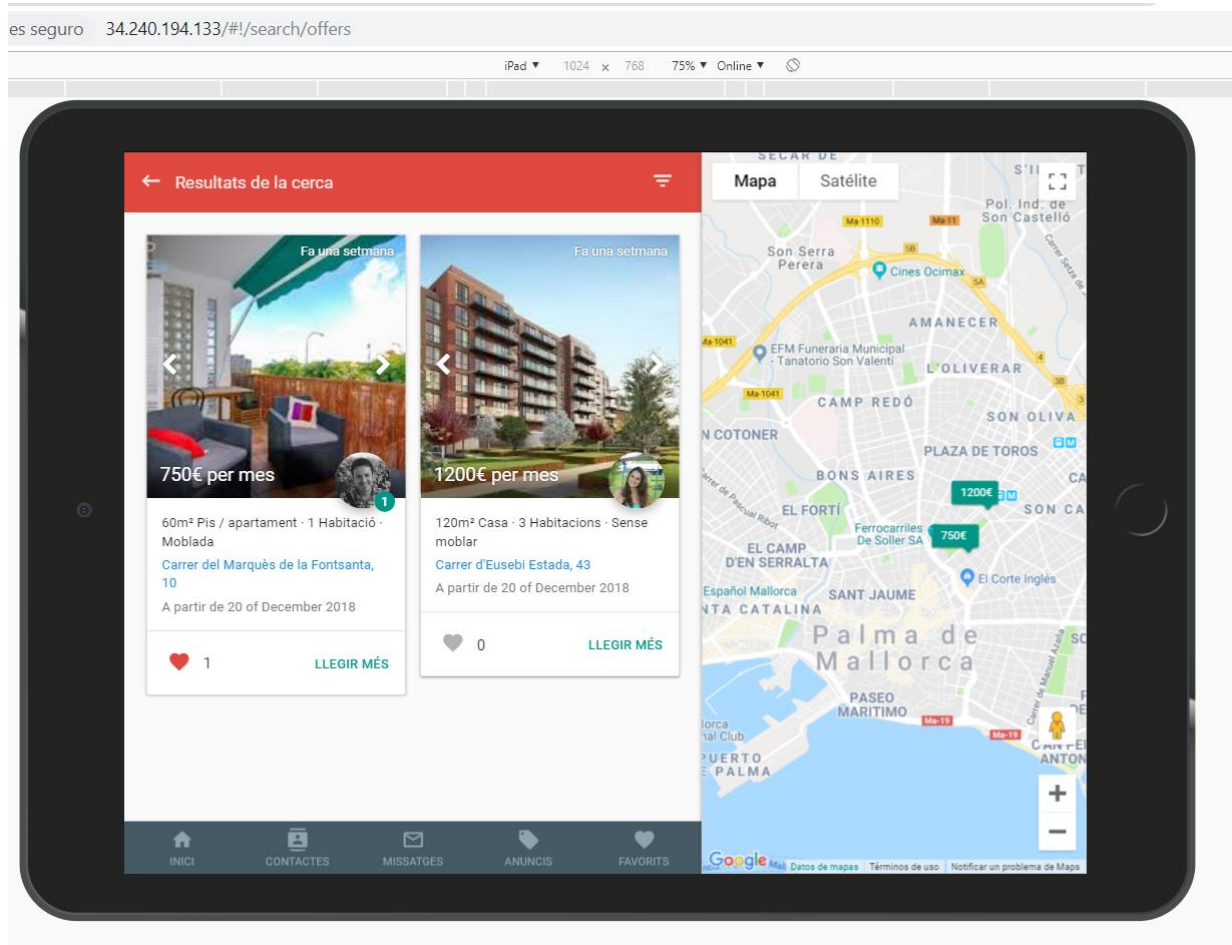




7.2.4. Gestió de contactes i missatgeria



7.2.5. Motor de disponibilitat



7.3 Webservice de consulta de referències

S'ha publicat un webservice a través del qual es poden recuperar les referències d'un usuari (de moment únicament per identificador d'usuari, encara que seria interessant poder fer la consulta directament per mail).

Al igual que les peticions del front, es requereix autenticació, per la qual cosa seran necessàries dues invocacions:

- Una a l'endpoint `/oauth/token` del projecte back-auth
- Una a l'endpoint `/users/{id}/referencesView` (passant el token d'autenticació obtingut en la invocació anterior com un header http)

Per provar el correcte funcionament del webservice hem fet ús de l'eina **Postman**:

Petició d'obtenció de token d'autenticació:

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://34.240.194.133:9080/oauth/token?client_id=ftg-front&client_secret=frontadmin&grant_type=password...
- Status:** 200 OK
- Time:** 448 ms
- Size:** 1.44 KB
- Headers (1):** Authorization: Basic dG...
- Body:** A JSON object containing an access token and a refresh token.

```
1 - {
2   "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
   .eyJ1c2VyX25hbWUiOiJ0b211dXJvaWdAZ21haWwvY29tIiwic2Nv
   cGUiOiJ1dG9zZCJdLCJleHAiOiJlNDY1NTM4NDksInVzZXJZCjE6I
   j1mNjMzZjE1TFkzJzE1NDY1NTM4NDksInVzZXJZCjE6Ij1mNjMz
   F1dGhvcml0aWVzIjpbIj1JPTFVZmVvbnVlIj0uLj1mNjMzZjE1
   iNi1kNTVlLQ4YTMtOWI0My11MmQ1N2IxM2Vkn2I1LCJybG11bnRf
   aWQiOiJ0ZmctZnJvbnQifQ
   .auc19UZpmtE7malugsPo10A50AkLjvrLH8b9at05MI",
3   "token_type": "bearer",
4   "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
   .eyJ1c2VyX25hbWUiOiJ0b211dXJvaWdAZ21haWwvY29tIiwic2Nv
   cGUiOiJ1dG9zZCJdLCJleHAiOiJlNDY1NTM4NDksInVzZXJZCjE6I
   WI0My11MmQ1N2IxM2Vkn2I1LCJleHAiOiJlNDY1NTM4NDksInVzZX
   JZCjE6Ij1mNjMzZjE1TFkzJzE1NDY1NTM4NDksInVzZXJZCjE6I
   j1mNjMzZjE1TFkzJzE1NDY1NTM4NDksInVzZXJZCjE6Ij1mNjMz
   4NyIsImF1dGhvcml0aWVzIjpbIj1JPTFVZmVvbnVlIj0uLj1mNjMz
   MzRhMmEyZS0xZjE2LTQ2YVwtOTY1YS1iYmY1NWVjYmMwZGEiLCJj
   bG11bnRfYWQiOiJ0ZmctZnJvbnQifQ
   .62PxTGSdE4aKxpTy7tMWRqCm0Yk4uYPdm2oHB-OggKE",
5   "expires_in": 1799,
6   "scope": "test",
7   "user_id": "9f63b37b-1df1-49b9-b7e5-355d65025887",
8   "jti": "c74072b6-d55b-48a3-9b43-e2d57b13ed7b"
9 }
```

Petició de consulta de referències:

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://34.240.194.133:8080/users/9f63b37b-1df1-49b9-b7e5-355d65025887/referencesView
- Status:** 200 OK
- Time:** 413 ms
- Size:** 1.65 KB
- Headers (1):** Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX25hbWUiOiJ0b211dXJvaWdAZ21haWwvY29tIiwic2Nv...
- Body:** A JSON array containing a list of references.

```
17   },
18   "references": [
19     {
20       "id": "470ca050-cc98-4295-9c59-4e28d236b8c3",
21       "years": 0,
22       "months": 11,
23       "weeks": 0,
24       "freeText": "Tomeu fou un excel·lent propietari
   . Sempre m'ajudà quan hi hagué alguna
   avaria en el pis. Cap problema per
   recuperar la fiança al final.",
25       "rating": "Positive",
26       "trusts": [
27         "Helpful",
28         "Communicative"
29       ],
30       "author": {
31         "id": "3275d586-9e1b-49e2-8077
   -258497b8e005",
32         "firstName": "Albert",
33         "lastName": "Grau",
34         "userType": "Tenant",
35         "profilePicture": "DEFAULT_PROFILE_PICTURE",
36         "backgroundPicture": "DEFAULT_BACKGROUND_PICTURE",

```

7. Conclusions

Com ja he introduït en capítols anteriors, el principal objectiu d'aquest TFG ha estat desenvolupar un projecte IT complex totalment des de zero, començant per la fase de ideació i planificació fins al desplegament final. Crec que participar activament en totes les fases del cicle de vida d'un projecte de software és una tasca que, encara que no impossible, difícilment podem dur a terme al llarg de la nostra carrera professional, ja que un cop entrem en el món laboral solem especialitzar-nos en una determinada àrea. Personalment, sempre he estat lligat a tasques d'arquitectura i desenvolupament, però tinc poca experiència per exemple en tot el que fa referència a desplegament i anàlisi formal. En quant a aquest darrer punt, destacar que assignatures cursades a la UOC com són "Enginyeria de requeriments" i "Enginyeria de programari i components distribuïts" m'han servit de gran ajuda.

Crec que aquesta mateixa reflexió és si cap més aplicable a la fase de desenvolupament. En la majoria de projectes que segueixen una típica arquitectura de tres capes (vista, negoci i dades) també és habitual que un desenvolupador s'especialitzi més en alguna de les capes. Tornant al meu cas personal, sempre he estat més enfocat a la capa de negoci, tocant la capa de dades per damunt, però pràcticament sense entrar en la vista. Aquest TFG m'ha servit per aprendre noves tecnologies i bones pràctiques en àrees de desenvolupament en les quals pràcticament no tenia experiència així com per aprofundir i especialitzar-me en aquelles que sí ja en tenia.

Muntar el projecte des de zero m'ha obligat a tenir que prendre decisions, i per tal de prendre les decisions correctes he hagut de recopilar informació i avaluar pros i contres de cada alternativa. En la capa de dades, utilitzar una base de dades relacional, documental o de grafs? Quin framework utilitzar en la capa de presentació: JSF, React o Angular? A l'hora de desplegar en el cloud: Heroku, OpenShift o Amazon EC2? I un llarg etcètera.

Crec que un cop entregat el producte final, puc afirmar que s'han assolit els objectius que m'havia plantejat abans de començar el TFG. Com a resultat tenim publicada una plataforma web totalment usable amb pràcticament totes les funcionalitats que s'havien proposat en la planificació inicial disponibles. A més, estic orgullós d'haver emprat un stack tecnològic actual i molt potent, i d'haver après algunes tecnologies desconegudes per jo fins al moment, com per exemple la base de dades de grafs Neo4J.

En quant al seguiment i la metodologia aplicada durant tot el semestre, s'han respectat les fites i dates d'entrega. La comunicació amb el consultor ha estat molt fluïda i aquest fet ha ajudat a complir el planning i timing proposats. Tot i així, al final de cada fita important ha estat necessari un esforç extra per poder arribar a temps. Penso que per desenvolupar un projecte des de zero, un semestre és molt ajustat, sobretot si compaginem els estudis amb feina a jornada completa.

Ja per acabar, afegir que l'experiència ha estat molt positiva, m'ha ajudat a créixer com a professional i que tinc la intenció de depurar el projecte i afegir-hi

noves funcionalitats en el futur. Segurament, el següent pas serà intentar empaquetar la aplicatiu amb algun framework tipus Cordova o Phonegap per poder-la fer servir com aplicació mòbil Android i IOS i no només com una plataforma web que és el que s'ha desenvolupat fins a la data.

8. Glossari

Angular	Framework per aplicacions web desenvolupat en TypeScript, de codi obert i mantingut per Google, que s'utilitza per crear i mantenir aplicacions web d'una sola pàgina.
Apache Maven	Eina de software de codi obert, mantinguda per la fundació Apache, per la gestió i construcció de projectes java, basat en un format XML.
Cordova	Framework open-source, mantingut per Apache, que permet emprar tecnologies web estàndard tals com HTML5, Javascript, etc. pel desenvolupament multi plataforma, evitant així el desenvolupament natiu de cada plataforma mòbil.
Endpoint	En el context de APIS REST, un endpoint fa referència a una URL relativa o absoluta que permet l'accés a un punt d'entrada de la API.
IDE	Acrònim de Entorn de desenvolupament.
JSON	Acrònim de JavaScript Object Notation. És un format de fitxer estàndard que empra text "human-readable" per transmetre objectes de dades.
Material Design	Llenguatge visual que sintetitza els principis clàssics del bon disseny amb la innovació de tecnologia i ciència. Desenvolupat i mantingut per Google.
Mobile First	Metodologia de disseny d'interfícies d'usuari que consisteix en dissenyar els aplicatius web amb resolució d'escriptori començant per la resolució mòbil.
Neo4j	Software lliure de base de dades orientada a grafs implementada en java.
Npm	Gestor de paquets i llibreries enfocat a projectes Javascript.
Oauth2	Protocol d'autenticació que permet a les aplicacions obtenir accés limitat a comptes d'usuari en un servei HTTP, com per exemple Facebook o GitHub.
Postman	Eina especialitzada en el consum de RestFul APIs i webservicess. Ofereix una interfície d'usuari que facilita el procés de peticions HTTP amb diferents formats tals com XML, JSON, etc.
React	Biblioteca javascript de codi obert, mantinguda per Facebook, orientada a la creació d'interfícies web amb l'objectiu d'animar el desenvolupament d'aplicacions en una sola pàgina.
Rad-only	API només de consulta. Permetrà recuperar informació però no fer cap tipus d'inserció o edició de recursos a través d'ella.
Responsive	Tècnica de disseny web adaptatiu que cerca la correcta visualització de una mateixa pàgina en diferents tipus de dispositiu. Generalment: mòbil, Tablet i escriptori.

Spring	Framework pel desenvolupament d'aplicacions i contenidor de inversió de control, de codi obert per la plataforma Java.
State-less	Una API serà State-less sempre i quan cada petició contingui tota la informació necessària per poder ser processada pel servidor, de forma que el servidor no emmagatzema informació de cap petició rebuda.
Webpack	Paquet de mòduls Javascript de codi obert, l'objectiu principal del qual és agrupar els arxius pel seu ús en un navegador, però també capaç de transformar i empaquetar qualsevol recurs o asset com HTML, CSS o imatges.
Websockets	Tecnologia que proporciona un canal de comunicació bidireccional y full-duplex sobre un únic socket TCP. Està dissenyada per ser implementada en navegadors i servidors web.

9. Bibliografia

1. Base de datos orientada a grafos. Wikipedia
 - *URL:* https://es.wikipedia.org/wiki/Base_de_datos_orientada_a_grafos
 - *Data:* 28/10/2018
2. Concepts: Relational to graph. Neo4j
 - *URL:* <https://neo4j.com/developer/graph-db-vs-rdbms>
 - *Data:* 28/10/2018
3. Git Documentation
 - *URL:* <https://git-scm.com/doc>
 - *Data:* 15/10/2018
4. Graph Databases
 - *Autor:* Ian Robinson, Jim Webber, and Emil Eifrem
 - *Títol:* Graph Databases
 - *Edició:* O'Reilley, segona edició.
 - *Ciutat:* United States of America.
 - *Any:* 2016
5. Neo4j
 - *URL:* <https://neo4j.com>
 - *Data:* 21/10/2018
6. Material-Design
 - *URL:* <https://material.io>
 - *Data:* 18/10/2018
7. Restangular documentation
 - *URL:* <https://github.com/mgonto/restangular>
 - *Data:* 15/10/2018
8. Spring framework documentation:
 - *URL:* <https://spring.io>
 - *Data:* 15/10/2018
9. Spring-boot documentation:
 - *URL:* <http://spring.io/projects/spring-boot>
 - *Data:* 17/10/2018
10. Spring-data project documentation:
 - *URL:* <http://projects.spring.io/spring-data/>
 - *Data:* 17/10/2018

11. Spring Data Neo4j reference documentation. Chapter 10.4.5

- *URL:* <https://docs.spring.io/spring-data/neo4j/docs/current/reference/html/#reference:annotating-entities:graph-id>
- *Data:* 29/10/2018

12. Spring-security documentation:

- *URL:* <https://spring.io/projects/spring-security>
- *Data:* 18/10/2018

13. Stackoverflow

- *URL:* <http://www.stackoverflow.com>
- *Data:* 01/11/2018 – 20/12/2018

14. Two-way data binding. AngularJS

- *URL:* https://docs.angularjs.org/tutorial/step_06
- *Data:* 31/10/2018

15. Websockets, Wikipedia:

- *URL:* <https://es.wikipedia.org/wiki/WebSocket>
- *Data:* 19/10/2018

10. Annexos

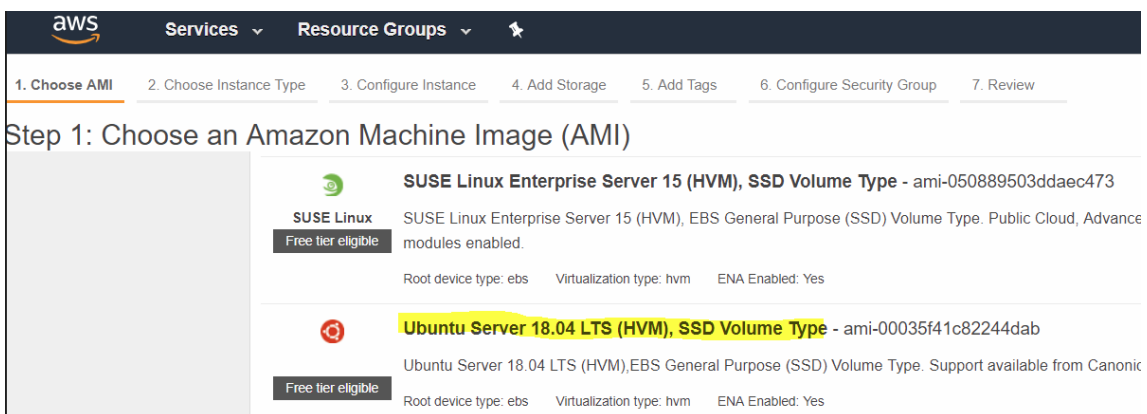
10.1 Guia de instal·lació a Amazon EC2

S'ha muntat una instància Ubuntu 18.04 a Amazon EC2, per tal de poder desplegar l'aplicatiu en el cloud. El primer pas és crear una conta a Amazon AWS. Es pot fer fàcilment seguint les instruccions de la pàgina web:

➤ <https://portal.aws.amazon.com/billing/signup#/start>

10.1.1 Configuració de la instància.

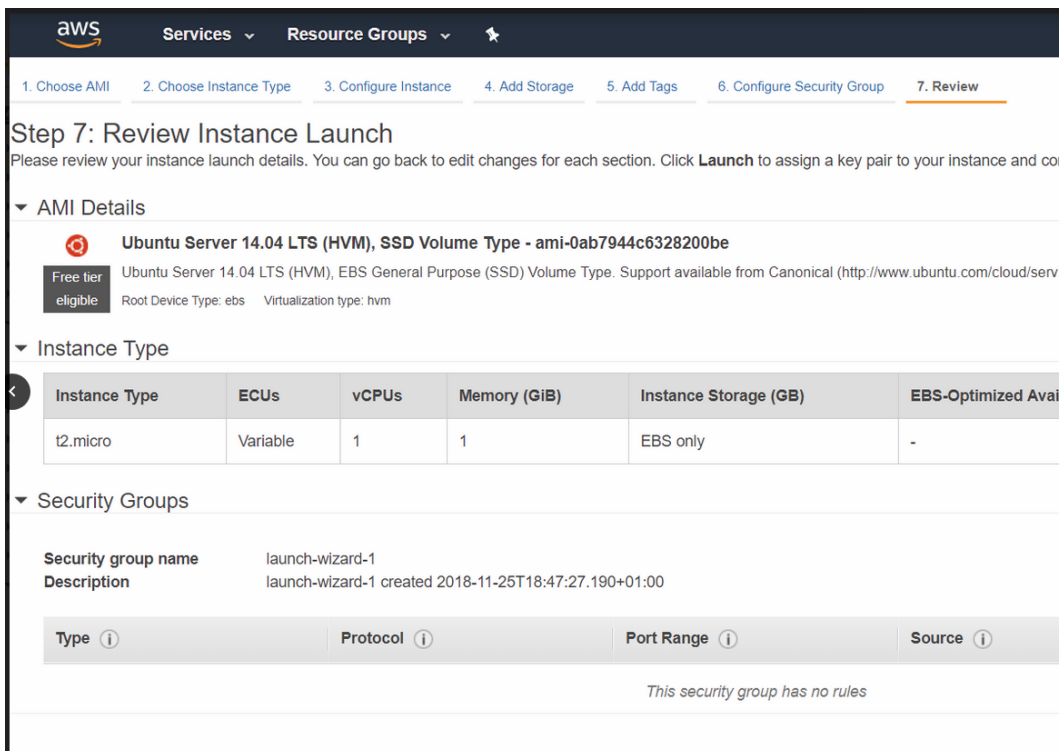
1. Creació de la instància. Escollim Ubuntu Server 18.04 LTS



The screenshot shows the AWS console interface for creating an EC2 instance. The top navigation bar includes 'Services' and 'Resource Groups'. Below the navigation bar, a progress bar indicates the current step: '1. Choose AMI'. The main content area is titled 'Step 1: Choose an Amazon Machine Image (AMI)'. Two AMI options are listed:

- SUSE Linux Enterprise Server 15 (HVM), SSD Volume Type** - ami-050889503ddaec473
SUSE Linux Enterprise Server 15 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advance modules enabled.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
- Ubuntu Server 18.04 LTS (HVM), SSD Volume Type** - ami-00035f41c82244dab
Ubuntu Server 18.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

2. Podem disposar d'una instància gratuïta 12 mesos, del tipus t2.micro.



The screenshot shows the AWS console interface for reviewing the instance launch. The top navigation bar includes 'Services' and 'Resource Groups'. Below the navigation bar, a progress bar indicates the current step: '7. Review'. The main content area is titled 'Step 7: Review Instance Launch'. The page shows the details of the instance launch, including AMI Details, Instance Type, and Security Groups.

AMI Details

- Ubuntu Server 14.04 LTS (HVM), SSD Volume Type** - ami-0ab7944c6328200be
Free tier eligible
Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/server>)
Root Device Type: ebs Virtualization type: hvm

Instance Type

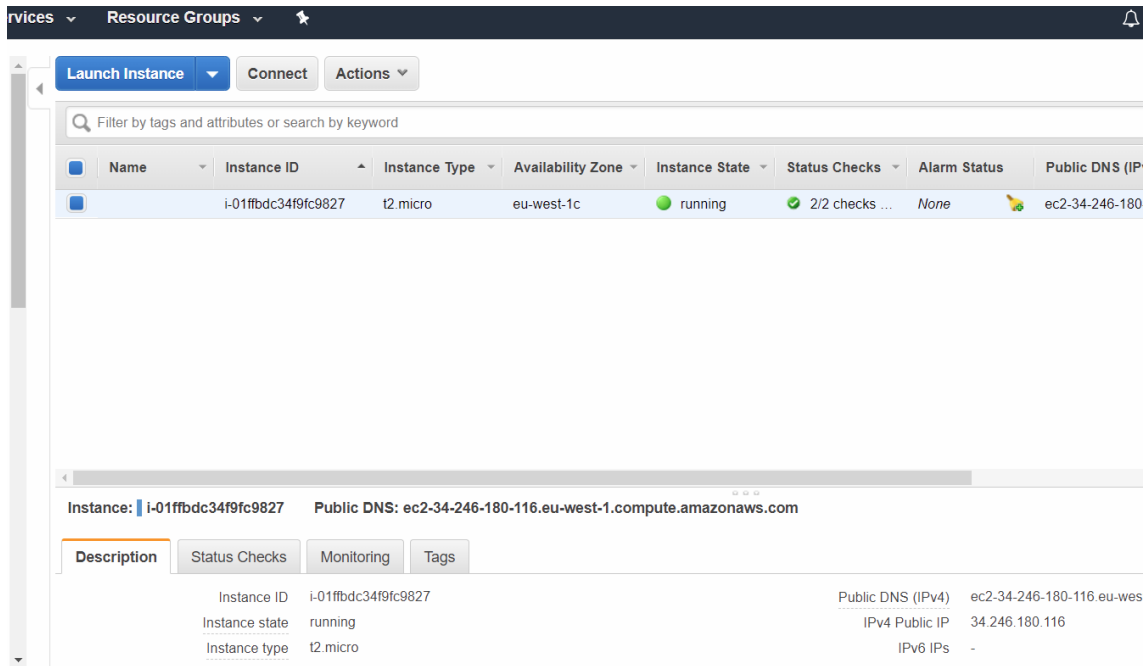
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Availability
t2.micro	Variable	1	1	EBS only	-

Security Groups

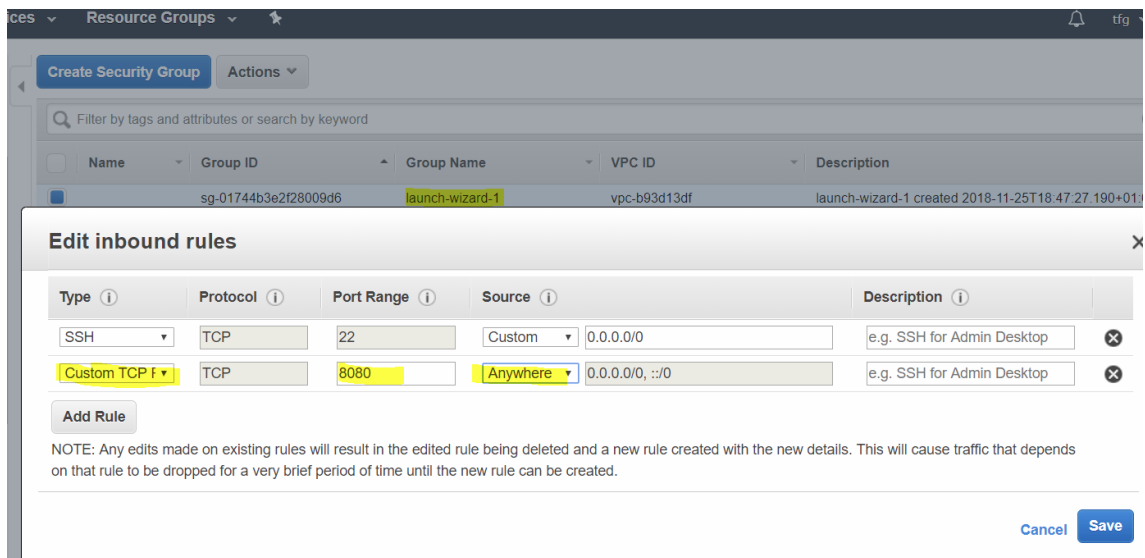
Security group name: launch-wizard-1
Description: launch-wizard-1 created 2018-11-25T18:47:27.190+01:00

Type	Protocol	Port Range	Source
This security group has no rules			

- Després de fer click en el botó “Launch”, ja disposem de l’instància aixecada i a punt per ser utilitzada



- Per acabar, ens falta modificar les regles de seguretat per obrir els ports que volem que siguin visibles des de fora. Es fàcilment configurable a través de en la pestanya ‘Security Groups’, opció ‘Edit Inbound Rules’



10.1.2 Instal·lació software en la instància.

➤ Java 8

```
$ sudo add-apt-repository ppa:webupd8team/java
$ sudo apt update
$ sudo apt install oracle-java8-set-default
```

Ruta per defecte: /usr/lib/jvm/java-8-oracle/

Set JAVA_HOME

```
$ export JAVA_HOME=/usr/lib/jvm/java-8-oracle
```

➤ Apache

```
$ sudo apt-get install apache2
$ sudo service apache2 [start | stop | status]
```

➤ Neo4j

```
$ sudo su
$ wget --no-check-certificate -O
- https://debian.neo4j.org/neotechnology.gpg.key
| sudo apt-key add -
$ apt update
$ apt install neo4j
```

Directories in use:

```
home:          /var/lib/neo4j
config:        /etc/neo4j
logs:          /var/log/neo4j
plugins:       /var/lib/neo4j/plugins
import:        /var/lib/neo4j/import
data:          /var/lib/neo4j/data
certificates:  /var/lib/neo4j/certificates
run:           /var/run/neo4j
```

```
$ sudo service neo4j [status | stop | start]
```

El ports 7474 (http) i 7687 (bold) no són visibles des de fora. Fa falta modificar la configuració de neo4j indicant

```
$ sudo chmod 777 -R /etc/neo4j
$ vim neo4j.conf
```

I afegint els ports a les regles inbound de Amazon EC2.

Descomentar les línies:

```
# dbms.connectors.default_listen_address=0.0.0.0
```

Reiniciar neo4j.

Per crear una bbdd. Si només feim feina amb una, és suficient amb indicar-ho a la config en el paràmetre dbms.active_database=

```
$ vim neo4j.conf
```

Descomentar la línia i indicar el nom corresponent:
dbms.active_database=neo4j

Es primer cop que ens conectam hem de canviar el password. Usuari i password per defecte: neo4j / neo4/
Hem hagut d'establir el password fent la connexió via bolt. Per http no hi ha hagut manera.

➤ **Problema CORS amb apache:**

```
$ cd /etc/apache2  
$ vim apache.conf
```

I afegir:

```
<Directory /var/www/html>  
    Order Allow,Deny  
    Allow from all  
    AllowOverride all  
    Header set Access-Control-Allow-Origin "*"   
</Directory>
```

Activar el mòdul:

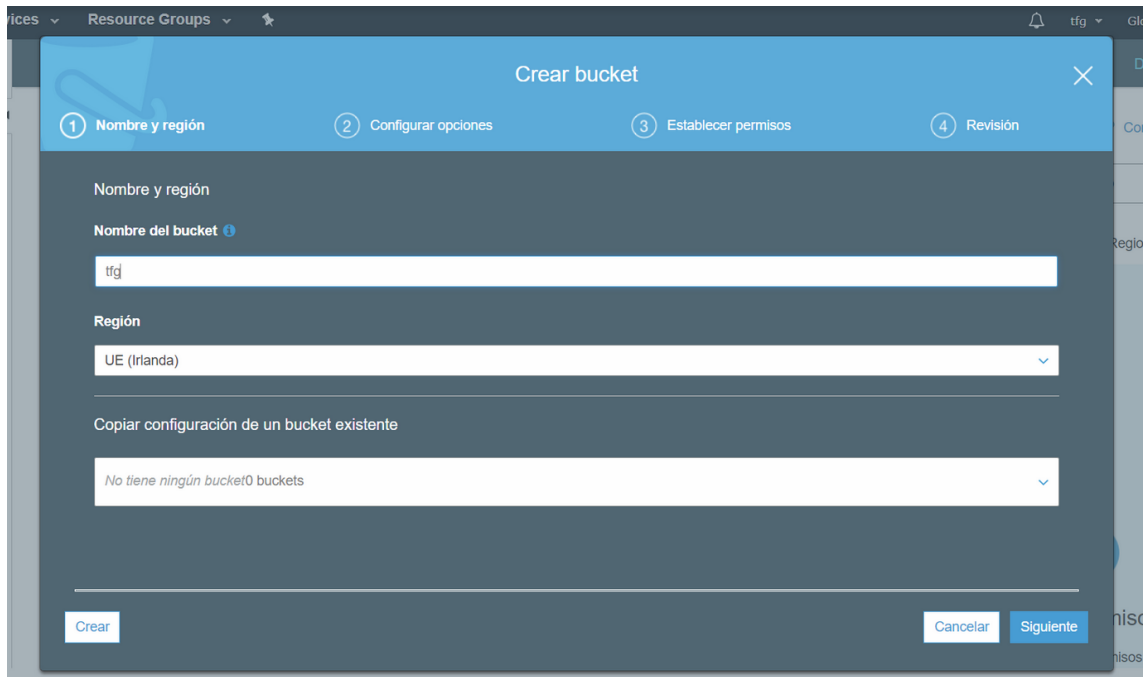
```
$ a2enmod headers
```

10.2 Guia de configuració Amazon S3

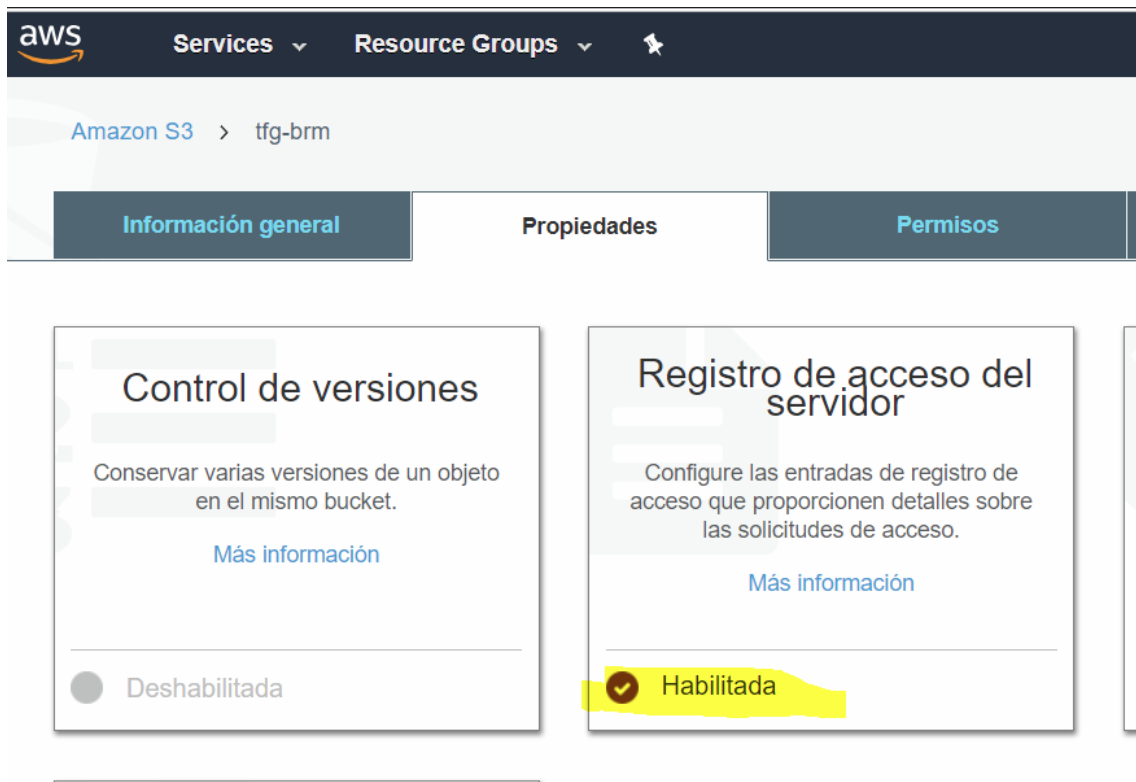
Hem utilitzat el servei cloud Amazon S3 per emmagatzematge de les imatges que es pugen a través de la plataforma web: fotos de perfil dels usuaris, imatges associades als anuncis, etc.

Reaprofitem el mateix compte Amazon AWS que hem creat per aixecar la instància, ja que Amazon centralitza la gestió de tots els seus serveis sota el mateix compte.

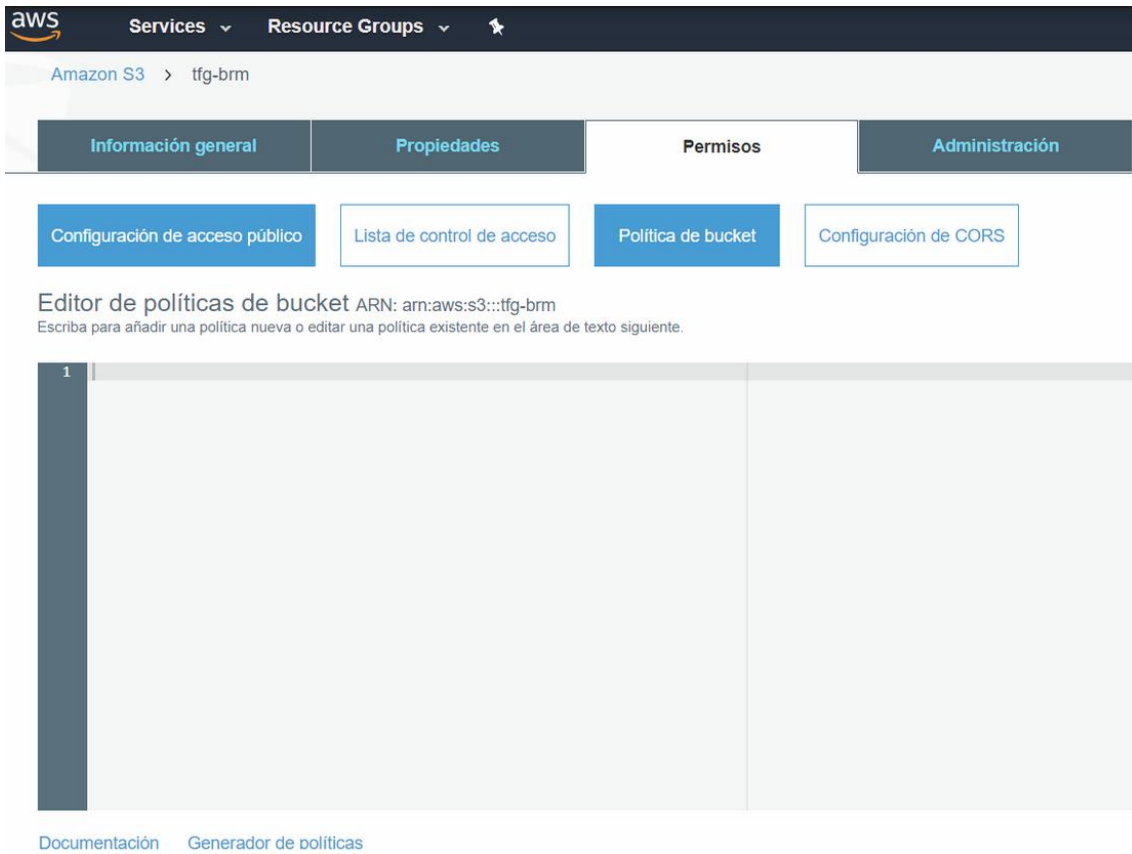
Seleccionem el servei S3 i el primer pas és crear un bucket d'emmagatzematge.



Habilitem el registre d'accés al servidor, necessari per accedir públicament als recursos del bucket.



Per restringir l'accés als recursos (només lectura, lectura-escritura, etc) necessitem crear una política de Bucket. Ho fem des de la pestanya "Permisos" del bucket, opció "Política de bucket"



Amazon posa al serveis dels usuaris un generador de polítiques, per facilitar la tasca. En fem ús:

AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#).

Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an IAM Policy, an S3 Bucket Policy, an SNS Topic Policy, a VPC Endpoint Policy, and an SQS Queue Policy.

Select Type of Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See [a description of elements that you can use in statements](#).

Effect Allow Deny

Principal

Use a comma to separate multiple values.

AWS Service

All Services (**)

Use multiple statements to add permissions for more than one service.

Actions All Actions (**)

Amazon Resource Name (ARN)

ARN should follow the following format: `arn:aws:s3:::<bucket_name>/<key_name>`.
Use a comma to separate multiple values.

Add Conditions (Optional)

Resource Groups ✦ tfgr

Add user

1 2 3 4 **5**

✓ **Success**

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://062793652651.signin.aws.amazon.com/console>

[Download .csv](#)

User	Access key ID	Secret access key
▶ ✓ tfg-front	AKIAJ2PMZWB76ZVE3SQQ	dthz5ZC/+rC+cgpT3sQxHXEs+R7jqQerEKT9tIL Hide

Seguint els passos que marca l'assistent, en la política final tenim que les lectures dels recursos del bucket són públiques mentre que les escriptures només és poden fer a través de tfg-front:

Amazon S3 > tfg-brm

Información general Propiedades Permisos **Administración**

Configuración de acceso público Lista de control de acceso Política de bucket Configuración de CORS

Editor de políticas de bucket ARN: arn:aws:s3::tfg-brm

Escriba para añadir una política nueva o editar una política existente en el área de texto siguiente.

```

1  {
2  "Version": "2012-10-17",
3  "Id": "Policy1543959383287",
4  "Statement": [
5    {
6      "Sid": "Stmt1543959349889",
7      "Effect": "Allow",
8      "Principal": {
9        "AWS": "arn:aws:iam::062793652651:user/tfg-front"
10     },
11     "Action": [
12       "s3:DeleteObject",
13       "s3:GetObject",
14       "s3:PutObject"
15     ],
16     "Resource": "arn:aws:s3::tfg-brm/*"
17   }
18 ]
19 }
```

[Documentación](#) [Generador de políticas](#)

10.3 Configuració projecte en local

10.3.1 tfg-front

S'utilitza **npm** com a gestor de les dependències del projecte. Les versions emprades durant el desenvolupament són:

- Node: 6.10.2
- Npm: 3.10.10

La instal·lació és pot fer sense problemes seguint les instruccions de la documentació oficial:

- Node: <https://howtonode.org/how-to-install-nodejs>
- Npm: <https://www.npmjs.com/>

Un cop instal·lats node i npm, ja podem arrancar el projecte tfg-front. Des de la ruta arrel del projecte executar:

```
$ npm install
$ npm run start
```

Obrim un navegador i ja tenim la plataforma web :

➤ <http://localhost:9001>

10.3.2 tfg-back

S'utilitza apache Maven com a gestor de les dependències del projecte. Versió emprada durant el desenvolupament: 3.5.2

La instal·lació és pot fer sense problemes seguint les instruccions de la documentació oficial: <https://maven.apache.org/download.cgi>

Un cop instal·lat Maven, ja podem arrancar el projecte tfg-back.

Des de la ruta arrel del projecte executar:

```
$ mvn clean install
```

Bé amb l'editor o per línia de comandes, executar les classes:

- `org.tfg.auth.BackAuth`: Arranca la API d'autenticació, en <http://localhost:9080>
- `org.tfg.api.config.BackResourceApplication`: Arranca la API d'accés als recursos, en <http://localhost:8080>