

---

This is the **published version** of the bachelor thesis:

Ram Ram, Bhupinder Kumar; Otazu Porter, Xavier, dir. Aplicació per trobar company de joc. 2022. (958 Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/264156>

under the terms of the  license

# Aplicació per trobar company de joc

Bhupinder Kumar Ram Ram

## Resum

**Resum**— Cada dia augmenta el nombre de videojocs en línia al mercat, cadascun amb una jugabilitat diferent i única, on la necessitat de trobar un o més companys és crítica per poder gaudir i tenir una bona experiència. Moltes vegades es perd l'interès en un videojoc per la frustració de no poder aconseguir cap amic o amiga amb el qual poder passar-ho bé jugant, ja sigui perquè no hi ha un interès comú o per diferents horaris de disponibilitat. Per aquests motius sorgeix la idea per una aplicació que et permet descobrir companys de joc, on el principal objectiu és posar en contacte dues persones o més que tinguin algunes preferències en comú.

**Paraules clau**— Aplicació mòbil, Frontend, Backend, React Native, Django, Django Channels, Django Rest Framework, Firebase Auth i Firebase Storage.

## Abstract

The amount of online video games is increasing every day. Each game with a different and unique game-play, where the need to find one or more partners is critical. In order to enjoy and have a good experience, many gamers find of essence, the need to have friends to play with. Without these connections, gamers often loose interest in the game. When brainstorming a simple solution to make sure that gamers can find similar interests in games and schedule availability, the idea of this project came to life. The main goal is to create an application that allows players to find team mates to connect with, who have common preferences.

**Keywords**— Mobile application, Frontend, Backend, React Native, Django, Django Channels, Django Rest Framework, Firebase Auth and Firebase Storage.



manera millorant les habilitats d'Enginyer del Software, i guanyar experiència per resoldre problemes del món real.

## 1 MOTIVACIÓ

**A**VUI en dia hi ha una gran varietat de jocs multijugador en línia que ens poden cridar l'atenció, on per poder tenir una bona experiència de jugabilitat, una submersió a la història i sobretot gaudir d'aquest és necessari l'ajuda d'altres persones, amb els quals cooperes per continuar endavant en el videojoc.

La motivació d'aquest projecte, neix d'aquesta reflexió. La idea és poder crear una aplicació que permeti interconnectar diferents persones d'una manera ràpida i senzilla, que comparteixen els mateixos característiques, amb l'objectiu d'expandir el cercle de coneguts d'aquestes.

Un cop va sorgir aquesta idea, es va valorar positivament donar una solució per aquest projecte mitjançant una aplicació mòbil, sorgeix l'oportunitat d'aprendre sobre totes les noves tecnologies que han sorgit en els darrers anys. Explorar el món del Frontend, per la interacció amb l'usuari, i també el Backend, l'encarregat d'entregar totes les funcionalitats necessàries als usuaris, d'aquesta

## 2 OBJECTIUS

L'objectiu principal d'aquest projecte és crear una aplicació mòbil que permet els diferents usuaris, en aquest cas jugadors de videojocs en línia, interconnectar entre ells. Els usuaris han de poder trobar altres jugadors amb els mateixos interessos i expectatives similars respecte al videojoc en línia que volen jugar. Addicionalment aprendre les tecnologies necessàries per poder desenvolupar l'aplicació, tant per la part que interactua amb l'usuari final, és a dir, el Frontend, com les tecnologies per realitzar les funcionalitats dels clients, el Backend.

A continuació definiré els objectius principals proposats a l'inici del projecte en més detall.

### 2.1 Objectius principals

#### 2.1.1 Implementar gestió d'usuaris

Un dels punts més clau de l'aplicació, és la implementació dels usuaris, ja que l'aplicació hi ha de proveir a un usuari l'espai necessari per emmagatzemar les seves dades personals, a més, l'aplicació ha de poder identificar-lo de manera única.

- E-mail de contacte: bhupinder.kumar@autonoma.cat
- Menció realitzada: Enginyeria del Software
- Treball tutoritzat per: Xavier Otazu Porter
- Curs 2021/22

Per complir l'objectiu proposat, no cal distingir diferents rols pels usuaris, perquè tots tenen els mateixos privilegis i poden realitzar les mateixes accions al llarg de l'aplicació. Tanmateix, les sessions dels usuaris a l'aplicació han de ser persistents, és a dir, un cop que l'usuari fa l'inici de sessió no ho ha de tornar a fer un altre cop, a no ser que implícitament hagi tancat la sessió.

### 2.1.2 Implementar preferències d'usuari

Amb l'objectiu de poder interconnectar usuaris, cal saber les seves preferències respecte els videojocs en línia, quines són les seves expectatives a l'hora de jugar i en quin idioma prefereixen comunicar-se, segons les preferències dels usuaris, aquests podran filtrar i trobar el company més adequat.

Per poder assolir aquest objectiu, per a cada usuari de l'aplicació s'ha de guardar els diferents videojocs d'interès, les seves expectatives, i un idioma per la comunicació de la seva preferència.

### 2.1.3 Implementar comunicació

La part més important de l'aplicació és la comunicació entre els diferents usuaris que s'han interconnectat. L'aplicació té tres maneres diferents per aconseguir que dos usuaris és interconnectin:

- La forma principal per poder connectar amb altres usuaris és mitjançant les sales públiques, ja sigui creades pel mateix usuari o per altres.
- Invitar a un altre usuari a una sala, que pot ser pública o privada, on la participació d'altres usuaris pot ser possible.
- Invitar a un altre usuari a una sala només dedicada per l'usuari invitat i el que invita, sense la possibilitat de participació de nous usuaris.

Els usuaris poden crear un nombre il·limitat de sales, on també es pot especificar les seves preferències de manera informativa pels usuaris que es vulguin unir a aquesta.

Cada sala té un xat associat mitjançant el qual els usuaris es poden comunicar a temps real.

## 2.2 Objectius d'aprenentatge

Per aconseguir el desenvolupament adequat d'aquest projecte, és necessari arribar a uns certs nivells de domini tècnic sobre les tecnologies que permeten la creació tant del Frontend com del Backend.

### 2.2.1 Frontend

Per desenvolupar una aplicació mòbil, és essencial el coneixement tècnic necessari per poder complir amb les següents funcions:

- Creació de pantalles visuals amb la qual l'usuari final interactua.
- Desplaçament entre pantalles de l'aplicació.
- Comunicació mitjançant peticions HTTP [17].
- Gestió de l'estat de l'aplicació.

### 2.2.2 Backend

Els requeriments tècnics en l'àmbit de Backend són els més importants, ja que és la base del projecte, i sense aquest no podem aconseguir un producte final, per tant, s'ha de complir amb les següents funcions:

- Creació d'una API REST [1], preparada per respondre les peticions dels usuaris.
- Gestió de bases de dades.
- Servei per un xat en temps real.

## 3 ESTAT DE L'ART

Actualment, a l'abast mercat de les aplicacions mòbils hi existeixen algunes aplicacions que donen un servei per interconnectar diferents persones per assolir diversos tipus de necessitats, però només s'ha centrat en les aplicacions que tenen d'objectiu el públic que juga videojocs en línia.

Les aplicacions que més s'apropen al servei que ofereix aquest projecte, són "WeGamers" i "UNBLND".

"WeGamers" [20], és una comunitat social, més enfocada a la creació d'articles informatius sobre diferents videojocs, en canvi, "UNBLND" [4] és una aplicació exclusivament per xatejar, però sense fer un èmfasi a les preferències de les persones als videojocs. Les dues aplicacions tenen un xat en temps real, però per arribar a aquest punt, s'ha de realitzar molts passos, que fan perdre temps.

Al mercat s'ha vist feble la funcionalitat d'interconnectar diferents usuaris amb unes preferències comunes en els videojocs en línia. En definitiva, aquest projecte creu en la senzillesa per interconnectar jugadors, amb el mínim nombre de passos a executar per part d'una persona que està en cerca de companys de joc, per aquesta raó sorgeix el desenvolupament d'una aplicació fàcil d'utilitzar sense contingut que s'allunya de l'objectiu principal. A més aquesta aplicació està dirigida a un segment concret d'usuaris, que només tenen com a interès jugar videojocs en línia, i no volen perdre el temps en accions i informació innecessària.

## 4 METODOLOGIA

En ser un projecte que només serà desenvolupat per una persona, s'utilitzarà un sistema senzill però efectiu, el Kanban [8]. Això és degut a la seva visualització del flux del treball, a més de la flexibilitat que dona per poder fer canvis en la planificació de manera fàcil i senzilla, sense perdre molt de temps. A més, ens permet tenir una visió global i control sobre les tasques del projecte, i en quin estat està la feina que s'està realitzant. Es farà servir l'eina que ens proporciona Jira Software, per poder gestionar les fases i el Kanban.

Amb el suport del Kanban es marquen unes fases, on cada una d'aquestes dividirà el treball en quatre segments principals.

## 4.1 Fase inicial

Un cop s'ha aconseguit fixar l'ideal del projecte, toca afrontar el problema que és pretén resoldre, i per això s'ha de desglossar el problema inicial, és a dir, realitzar una recerca dels requisits necessaris per complir l'objectiu [Taula 1].

També és important investigar les tecnologies que es faran servir per desenvolupar el projecte, per saber els avantatges que ens proporcionen, i sobretot les limitacions que ens poden suposar.

S'han d'establir les tasques necessàries per poder realitzar els requisits trobats, a més a més, de fer una planificació per dur a terme el projecte en un termini de temps establert.

## 4.2 Fase de disseny

Seguidament de la fase anterior, tenint els requisits necessaris i les tecnologies fixades pel desenvolupament del projecte, la segona fase, tracta de definir les estructures necessàries per poder desenvolupar el projecte.

En aquesta fase s'ha de fer un disseny de la base de dades [Fig. 11], amb els atributs necessaris per poder complir amb els requisits definits, a més d'establir les peticions per la comunicació entre el Frontend i el Backend. Per la part del Frontend, establir les pantalles amb les quals l'usuari interactua mitjançant l'aplicació.

## 4.3 Fase de desenvolupament

És la fase més important i la més llarga del projecte, on la càrrega del treball és més elevada comparant amb les altres fases, ja que l'objectiu d'aquesta fase és desenvolupar tant el Backend, com el Frontend.

Amb el suport dels dissenys que s'han realitzat a la fase anterior, en aquesta fase s'ha d'aconseguir fer un MVP (Mínim producte viable). L'objectiu que hi ha d'assolir aquest MVP és la realització dels requisits: R-1, R-2, R-3, R-4, R-5, R-6, R-7, R-8, R-9, R-10 i R-11 [Taula 1].

## 4.4 Fase d'afinament

L'última fase també consisteix en el desenvolupament del producte, per continuar millorant-ho, afegint les funcionalitats restants, i corregir errors en el disseny que es puguin trobar al llarg del desenvolupament.

## Execució

Un dels grans problemes a enfrontar durant la tercera fase, utilitzant Kanban com a metodologia, que ha creat un impediment del flux de treball adequat, és la creació de tasques grans i poc detallades, sense objectius fàcils i raonables. Això creava una incertesa a l'hora de començar a tractar amb una nova tasca, cosa que causava una pèrdua de temps. Per evitar aquest comportament durant l'execució de la quarta fase i part de la tercera, es van actualitzar les tasques del Jira, on el focus és més petit, amb més detalls i ràpid de visualitzar.

Aquest canvi va aportar un impacte molt positiu a la productivitat, fent que el projecte es pugui desenvolupar d'una manera progressiva i més constant.

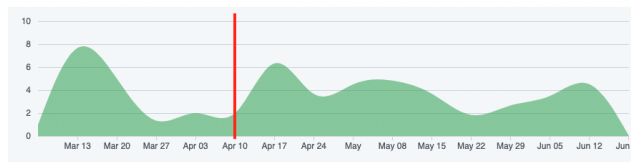


Fig. 1: Addicions de codi

## 5 EINES

En aquest apartat parlem de totes les eines utilitzades pel desenvolupament, tant pel Frontend com pel Backend, i la raó darrera d'aquestes.

### 5.1 Eines de desenvolupament per Frontend

#### React Native

És un framework de codi obert creat per Facebook, més conegut com Meta Platforms, que permet el desenvolupament d'aplicacions per Android, Android TV, iOS, macOS, tvOS, Web, Windows i UWP (Universal Windows Platform).

El seu lema és "Learn once, write anywhere" [12], i és la principal raó per la qual s'ha escollit React Native, l'avantatge que et dona en desenvolupar una aplicació només un cop i poder utilitzar-la en totes les plataformes mencionades anteriorment.

React Native permet desenvolupar aplicacions en JavaScript o TypeScript [11], per aquest projecte s'ha escollit TypeScript [5], ja que és un llenguatge de tipus fort o tipus segur.

#### Redux Toolkit

És una llibreria de JavaScript de codi obert [19], que permet tenir un control sobre l'estat de l'aplicació, és a dir, et permet guardar, actualitzar i visualitzar informació d'un estat per tota l'aplicació sense tenir passar manualment aquest estat des d'una pantalla a l'altre, facilitat el desenvolupament i reduint la complexitat del codi.

#### React Navigation

L'aplicació requereix múltiples pantalles, amb navegació entre elles, però React Native per defecte no té la funcionalitat de navegació entre pantalles, per aquesta raó la necessitat d'aquesta llibreria. React Navigation és una llibreria [10], escrita en JavaScript, independent, que permet implementar la navegació.

#### Axios

És un client HTTP, basat en promeses [16], per l'entorn de Node.js i navegadors, amb la propietat de ser isomorf, és a dir, que pot executar-se tant en navegadors com en aplicacions que utilitzen Node.js amb el mateix codi. S'utilitza Axios a l'aplicació per la comunicació entre el Backend, mitjançant una API REST.

## 5.2 Eines de desenvolupament per Backend

### Django Rest Framework

És una llibreria de Python amb un conjunt d'eines, pel Framework de Django, que redueix la quantitat de codi per crear una aplicació REST. Aquesta llibreria és utilitzada per realitzar la majoria de les funcions necessàries d'aquest projecte. En estar en una capa superior del Framework Django, es pot aprofitar de totes les funcions que ofereix Django, per la creació d'una base de dades i tota la gestió que comporta aquest.

### Django Channels

Aquesta llibreria de Python [13], agafa el projecte de Django i fa una extensió per donar-li encara més funcionalitats. Django per defecte només dona suport al protocol HTTP, però amb aquesta llibreria afegeix la utilització de més protocols com: WebSockets, chat protocols, IoT protocols i alguns més.

Per la necessitat del projecte de realitzar un xat en temps real, aquesta llibreria ens ofereix el servei ideal per poder aconseguir-ho, utilitzant WebSockets.

### Redis

Redis és una base de dades en memòria, ràpida, de codi obert, on les dades són emmagatzemades com a clau-valor [18], aquesta també es pot utilitzar com una base de dades persistent si cal.

Redis s'integra amb Django Channels utilitzant Websockets, per poder donar resposta ràpida a tots els usuaris connectats en un xat.

### Firebase

Firebase és una plataforma pel desenvolupament d'aplicacions tant a mòbils com a webs, allotjada al núvol, que pertany a Google. Aquesta plataforma ens proporciona accés a una gran varietat de recursos amb una implementació relativament ràpida.

El servei que ens interessa és d'autenticació d'usuaris, Firebase Auth [14], ja que aquest servei permet als usuaris registrar-se amb correu electrònic i contrasenya convencional, a més de registrar-se amb Google, Apple Id, Twitter, i altres proveïdors rellevants. També ens interessa el servei que proporciona per l'emmagatzematge, Firebase Storage [9], per guardar fitxers, com són les imatges del perfil d'usuari.

## 5.3 Eines de suport

Les eines que han sigut de gran suport per al desenvolupament d'aquest projecte, però no afecten el producte final.

Els dissenys s'han realitzat mitjançant un software [3] que permet fer dibuxos gràfics des de diferents plataformes, o utilitzant-ho per web. Per escriure codi del Frontend s'ha utilitzat l'editor de text IntelliJ IDEA [15], ja que aquest facilita la generació de codi repetitiu, a més té un bon suport per TypeScript. Pel desenvolupament del Backend s'ha fet servir l'editor de text PyCharm [6], que s'especialitza únicament en entorns de Python.

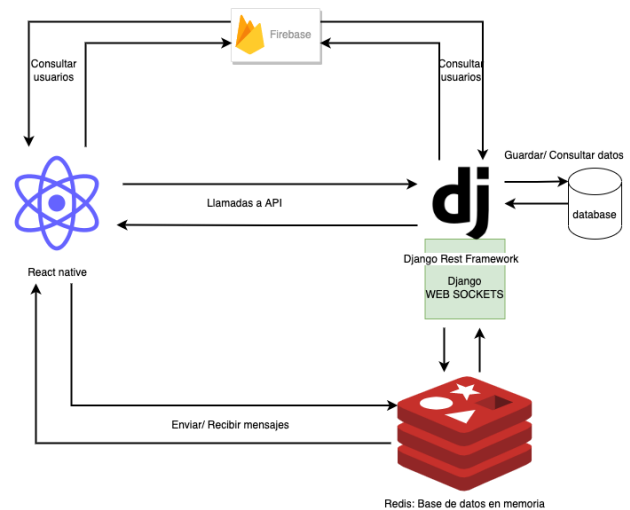


Fig. 2: Estructura del projecte

Per provar les peticions REST desenvolupades, l'eina Postman ha sigut de gran ajuda.

## 6 PLANIFICACIÓ

Tenir una proposta de la planificació ben estructurada és crítica pel bon desenvolupament del projecte, però en tractar-se d'un projecte de Software a mesura que va avançant el projecte, la creació de noves tasques o modificacions d'algunes creades anteriorment és inevitable per aquesta raó la flexibilitat en una planificació és vital per poder crear un projecte amb qualitat. Per tenir una visualització de la planificació més detallada veure Diagrama de Gantt [7] [Fig. 11].

### Seguiment

Tant la Fase Inicial com la Fase del disseny s'han executat sense cap mena de problemes ni retards, ja que aquest tractaven majoritàriament a fer recerca sobre el producte, les eines a utilitzar, i la creació de models a seguir per poder realitzar l'aplicació d'una manera satisfactòria.

A l'hora d'executar la Fase de Desenvolupament, què tenia d'objectiu realitzar un MVP, amb les funcionalitats bàsiques de l'aplicació, però durant l'execució d'aquesta fase es van trobar diverses dificultats, el resultat de les quals va ser un retard en la planificació inicial. Això és degut a diferents factors, dels quals en destaquen dos:

Un canvi introduït en els requisits inicials, aquest canvi tenia l'objectiu d'aconseguir un inici de sessió amb Google, que permet a l'usuari registrar-se a l'aplicació d'una manera ràpida i sense passos extres.

A més, un factor que no s'havia tingut en compte a la Fase Inicial del projecte a l'hora de fer la planificació era l'aprenentatge d'algunes tecnologies requerides pel desenvolupament d'una aplicació multiplataforma, fet que ha comportat un retard a la planificació, no obstant això, aquest procés era necessari.

## 7 DESENVOLUPAMENT I RESULTATS

A continuació s'exposa els resultats tant del desenvolupament del Frontend, com del Backend. Analitzant totes les

pantalles disponibles pels usuaris i les diferents accions que es poden realitzar en aquestes, també es comenta les restriccions imposades en algunes pantalles.

## Inici de sessió

L'inici de sessió de l'usuari, està dividida en dues pantalles, una per donar-se d'alta a l'aplicació, i l'altre fer l'inici de sessió. Comencem parlant de la pantalla de registre, ja que aquesta està conformada per 3 camps, un camp per introduir el correu electrònic, i els restants per la contrasenya, només es demana informació necessària per donar d'alta l'usuari. Aquests camps tenen una verificació tant per part del client, Frontend, com el servidor, Firebase Auth. Com a client abans d'enviar el formulari, es confirma que l'usuari ha introduït un correu amb el format correcte i una contrasenya de mínim sis dígit. Firebase també hi dona un servei per verificar el correu dels usuaris, però no s'utilitza, ja que no s'ha vist la necessitat.

La pantalla d'inici de sessió [Fig.3], compte amb dues seccions, la primera per fer un inici de sessió amb correu electrònic, i l'altre fent-ho amb un compte de Google. A més a més, té una acció que permet la navegació a la pantalla per enregistrar-se amb correu electrònic en cas de la necessitat. Si l'usuari és la primera vegada que fa l'inici de sessió aquest té un comportament diferent per realitzar la navegació, ja que, aquest és redirigit a una seqüència de pantalles per demanar-li les seves preferències, en cas contrari, és redirigit a la pantalla principal de l'aplicació.

Per fer la gestió d'usuaris, tant el Frontend, com el Backend utilitzen llibreries de Firebase Auth per encarregar-se de l'emmagatzematge dels usuaris, també ens proporciona una manera fàcil de poder enregistrar usuaris amb comptes de Google.



Fig. 3: Pantalla inici sessió

## Preferències d'usuari

Saber les preferències dels usuaris és una part essencial d'aquesta aplicació, per aquesta raó, s'han fet 3 pantalles diferents per poder seleccionar la informació més important, per després poder interconnectar amb els altres usuaris.

A la primera pantalla [Fig. 4], es demana a l'usuari de seleccionar tots els videojocs d'interès, aquest té diferents opcions a realitzar en aquesta situació: La primera i la més senzilla, però aporta poc valor a l'aplicació, és saltar la selecció de videojocs, en aquest cas, es redirigeix a l'usuari a la pantalla 3 [Fig. 5]. La segona situació que pot realitzar l'usuari és seleccionar els videojocs d'interès, i continuar a una pantalla, on proporciona informació més detallada sobre aquests videojocs seleccionats, finalment arribant a l'última pantalla on informa sobre l'idioma de preferència i l'horari ideal per jugar.

Per realitzar les preferències dels usuaris des del Backend, només s'ha fet servir Django Rest Framework per fer les crides REST.

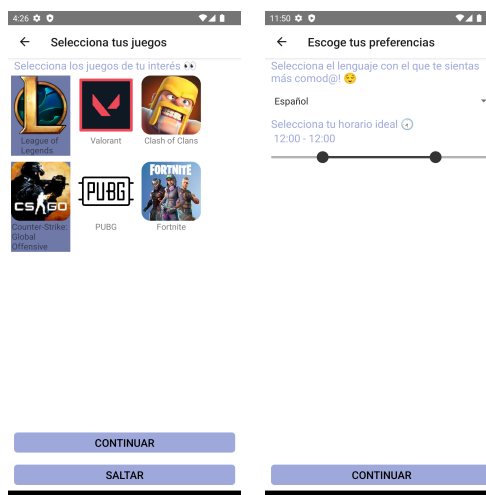


Fig. 4: Pantalla 1 selecció preferències Fig. 5: Pantalla 2 selecció preferències

## Pantalla principal

Un cop l'usuari ja s'ha donat d'alta a l'aplicació, havent fet el procés d'indicar les seves preferències, o d'iniciar la sessió, és redirigit a la pantalla principal. Aquesta està composta per un menú inferior que permet navegar entre 4 diferents opcions: Pantalla de sales, pantalla per trobar jugadors, pantalla amb tots els xats i finalment per gestionar el perfil.

### Sales

Es poden visualitzar totes les sales disponibles, creades pels altres usuaris de l'aplicació, on per cada una de les sales es mostra la informació més rellevant, el nom del videojoc, el tipus de jugabilitat, l'horari ideal i el nombre de participants, amb l'opció d'unir a la sala si és possible, ja que aquest pot tenir una limitació de nombre màxim de participants. Adicionalment, amb un botó situat a baix a la dreta [Fig. 6], se li permet crear una sala, indicant sobre el videojoc que es tracta, la jugabilitat que és pretén tenir a la sala, l'horari ideal, el nombre màxim de participants i si la sala és privada o no.

Tant l'acció de crear una nova sala com la d'unir-se a una ja existent, comporten uns efectes secundaris que afecten la pantalla de xat, ja que per cada sala que es va crear o unir-se, apareix una nova entrada de xat d'aquesta.

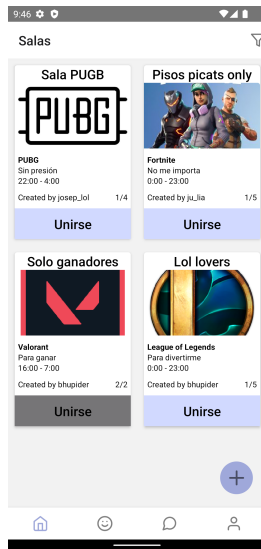


Fig. 6: Pantalla inicial



Fig. 7: Pantalla trobar companys

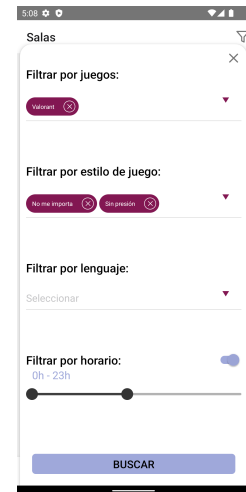


Fig. 8: Pantalla filtres

## Usuaris

Tots els usuaris participants de l'aplicació apareixen a la Pantalla per trobar companys [Fig. 7], mostrant la informació més rellevant i d'interès pels altres usuaris, com són els videojocs, l'idioma preferent i el seu horari per jugar. A més els usuaris poden realitzar dues accions, la primera és obrir un xat privat amb l'usuari, de manera que es crea una sala privada on només estan presents aquests dos, i l'altre és convidar a una sala, l'aplicació només permet convidar a una sala on el convidador és el propietari d'aquesta, a més, validant primer que l'usuari no està participant en aquesta sala.

## Filtratge

Per poder interconnectar amb usuaris que comparteixen els mateixos interessos, l'aplicació ofereix un filtratge tant de sales com d'usuaris, la pantalla per trobar usuaris i de sales hi apareix un botó amunt a la dreta [Fig.6] [Fig.7], permetent als usuaris de l'aplicació obrir una cortina pel davant de la pantalla [Fig.8] amb l'opció de poder filtrar. Aquesta cortina permet realitzar una cerca, videojocs, estils, llenguatge i un rang d'horari de vint-i-quatre hores, totes permeten, excepte el rang d'horari, una selecció múltiple de filtres i són opcionals, creant un abast molt gran de combinació per cercar sales o companys que millor s'adaptin a les necessitats d'un usuari.

## Xat

El xat [Fig.9] és la funcionalitat que permet els usuaris poder interactuar entre ells, i són un efecte secundari a les accions d'unir-se a una sala, obrir un xat privat, o convidar usuaris a sales, tant siguin privades com públiques. El xat permet una comunicació en temps real amb els usuaris connectats. Per poder tenir aquesta funcionalitat, s'ha fet una integració de Django Channels amb Django Rest Framework, per poder connectar cada una de les sales, amb un servei de WebSockets, a més utilitzant la base de dades de Redis per tenir una comunicació més ràpida.

La gestió dels xats és dur a terme individualment, és

a dir, per poder eliminar un xat, eliminar participants o abandonar, no tothom pot realitzar aquestes accions, les primeres dues només són possibles sent propietari d'una sala [Fig. 10] i l'acció restant quan no és el propietari.

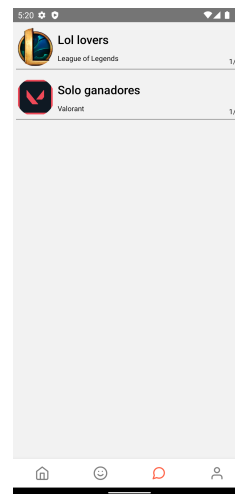


Fig. 9: Pantalla xat

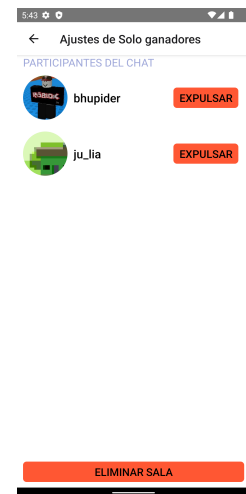


Fig. 10: Gestió xat

## Perfil d'usuari

La pantalla del Perfil d'usuari està conformada per dues seccions, la part superior amb informació de l'usuari, i l'altre restant amb informació de les preferències escollides, a més a més, amb un botó que permet tancar la sessió actual. L'usuari a més de poder veure el seu perfil amb la informació més rellevant, la seva foto de perfil, el correu electrònic amb el qual s'ha donat d'alta, i el nom d'usuari pel qual és vist al llarg de l'aplicació, també té la possibilitat de poder modificar el seu perfil privat, amb l'opció d'actualitzar el seu nom d'usuari, foto de perfil, nom, cognom i la descripció.

Per l'emmagatzematge de les imatges dels usuaris s'ha fet servir el servei que proporciona Firebase Storage, utilitzant-ho com a CDN (Xarxa de distribució de continguts) [2]. De manera que els usuaris carreguen les imatges al servidor de Firebase Storage, i tothom té accés a aquest mitjançant un URL.

La segona secció del perfil d'usuari, és la visualització de les preferències de l'usuari, mostrant el seu llenguatge, l'horari, i els videojocs d'interès. Amb l'opció de poder modificar-les, aprofitant les pantalles ja desenvolupades [Fig.4] [Fig.5], l'usuari modifica les seves preferències mitjançant aquestes.

## 8 CONCLUSIONS

Durant l'execució d'aquest projecte, s'han vist tecnologies, eines i plataformes molt interessants amb diferents serveis d'un gran interès per l'aplicació a desenvolupar, ja que el món del desenvolupament d'aplicacions mòbils és molt gran i amb moltes alternatives, que està evolucionant constantment. Considerant-ho tot, pel desenvolupament d'aquest projecte, a la fase inicial, es va escollir el framework de React Native, que en poques paraules, ens permet desenvolupar una aplicació mòbil multiplataforma. Un cop començat l'aprenentatge d'aquest framework, el detall més important i a la vegada preocupant va ser la seva dependència a les llibreries fetes per la comunitat, perquè aquest no té funcionalitats essencials i necessàries per les aplicacions mòbils, com és la navegació dins l'aplicació.

Per altra banda, un punt important del projecte, el Backend, es va veure afectat per la complexitat d'integració de diferents tecnologies per aconseguir l'objectiu final, com és el cas de treballar amb Firebase, Django Rest Framework i Django Channels, com a conseqüència la planificació es va veure endarrerida, però en tenir una planificació que ens permet ser flexible, per la complexitat d'algunes tasques, l'adaptació a aquest convenient es va poder solucionar correctament.

En conjunt, el problema que s'intenta resoldre amb aquest projecte, de poder interconnectar persones amb un interès similar, s'aconsegueix adequadament amb el grup de tecnologies escollides, sobretot a l'oferir un espai compartit on poden comunicar-se entre ells mitjançant el xat de l'aplicació.

### 8.1 Treball futur

Encara que el projecte té un bon potencial en les funcionalitats de l'aplicació, sempre hi ha espai per noves idees per donar-li una experiència encara més gratificant als usuaris. Una funcionalitat que pot aportar molt de valor al projecte, a més de generar una millora substancial a l'aplicació és la implementació de notificacions en rebre un nou missatge de xat, ja que actualment, l'única manera de saber si existeixen nous missatges és entrant a l'aplicació, a l'apartat dels xats i obrir el xat que ens interessa. L'apartat del filtratge pot tenir una nova secció per fer una cerca encara més precisa, sent aquesta un indicatiu del nivell d'experiència del jugador que és pretén cerca. Actualment, aquesta aplicació no està disponible pel públic general, un treball pel futur, és la publicació de l'aplicació a Google Store.

### 8.2 Agraïments

M'agradaria donar les gràcies al meu tutor Xavier Otazu per tota l'ajuda i l'orientació que m'ha proporcionat per

poder dur a terme aquest projecte. A més m'agradaria agrair també la meua família, perquè sempre m'han donat suport per poder seguir amb la carrera d'Enginyeria, amb recomanacions de la vida i com afrontar els problemes del dia a dia. Finalment, agrair a totes les amistats que he anat fent al llarg d'aquests anys, sobretot a la meua bona amiga Sofia, sempre m'ha donat bons consells.

## REFERÈNCIES

- [1] *¿Qué es una API de REST?* Cast. 8 de maig de 2020. URL: <https://www.redhat.com/es/topics/api/what-is-a-rest-api> (cons. 01-01-2022).
- [2] *¿Qué es una CDN?* Ang. amer. URL: <https://www.cloudflare.com/es-es/learning/cdn/what-is-a-cdn/> (cons. 07-04-2022).
- [3] A. *Draw.io*. Ang. amer. 2 de maig de 2022. URL: <https://drawio-app.com/product/>.
- [4] *About unblnd*. Ang. URL: <https://unblnd.com/about> (cons. 16-04-2022).
- [5] Afaq Arif. *Overview of TypeScript*. Ang. amer. 20 de febr. de 2021. URL: <https://learncodeweb.com/typescript/a-complete-overview-of-typescript/> (cons. 14-02-2022).
- [6] Simran Kaur Arora. *What is PyCharm?* Ang. 2 de març de 2022. URL: <https://hackr.io/blog/what-is-pycharm> (cons. 04-03-2022).
- [7] Atlassian. *Diagrama de Gantt*. Cast. URL: <https://www.atlassian.com/es/agile/project-management/gantt-chart>.
- [8] Atlassian i Radigan. *Kanban: Una breu introducció*. Cast. URL: <https://www.atlassian.com/es/agile/kanban> (cons. 18-02-2022).
- [9] *Cloud Storage for Firebase*. Ang. URL: <https://firebase.google.com/docs/storage> (cons. 22-06-2022).
- [10] Libny Corrales. *React Navigation*. Ang. amer. 6 d'oct. de 2021. URL: <https://www.asapdevelopers.com/react-navigation/> (cons. 22-06-2022).
- [11] Edmund Ekott. *Using TypeScript with React Native*. Ang. amer. 12 de maig de 2021. URL: <https://blog.logrocket.com/using-typescript-with-react-native/> (cons. 22-06-2022).
- [12] Facebook. *React native · learn once, write anywhere*. Ang. URL: <https://reactnative.dev/>.
- [13] Godwin. *Introduction — Channels*. Ang. URL: <https://channels.readthedocs.io/en/stable/introduction.html>.
- [14] Google Developers. *Firebase Authentication | Firebase Documentation*. Ang. URL: <https://firebase.google.com/docs/auth> (cons. 22-06-2022).

[15] JetBrains. *React Native* | *IntelliJ IDEA*. Ang. amer. URL: [https://www.jetbrains.com/help/idea/react-native.html#create\\_new\\_react\\_native\\_app](https://www.jetbrains.com/help/idea/react-native.html#create_new_react_native_app).

[16] Ilya Kantor. *Promise*. Ang. URL: <https://javascript.info/promise-basics> (cons. 24-06-2022).

[17] *Métodos de petición HTTP - HTTP* | *MDN*. Ang. amer. 22 de juny de 2021. URL: <https://developer.mozilla.org/es/docs/Web/HTTP/Methods>.

[18] Alexander Nnkwue. *Understanding Redis*. Ang. amer. 16 de gen. de 2020. URL: <https://blog.logrocket.com/guide-to-fully-understanding-redis/>.

[19] *Redux toolkit with TypeScript*. Ang. 14 de juny de 2022. URL: <https://redux-toolkit.js.org/tutorials/typescript> (cons. 22-06-2022).

[20] *WeGamers - game community on mobile*. Ang. amer. URL: <https://www.wegamers.com/> (cons. 16-04-2022).

## 9 ANNEXOS

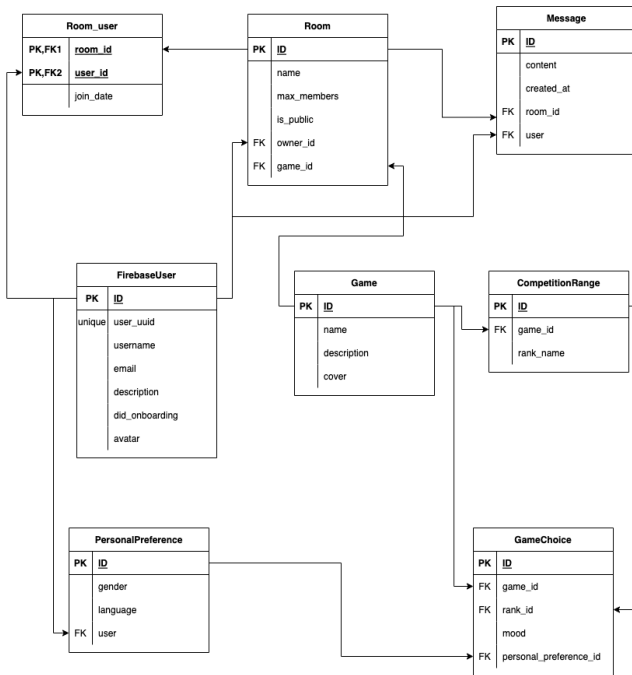


Fig. 11: Base de dades

Identificador del requisit	Descripció del requisit
<b>Requisits funcionals</b>	
R-1	L'usuari ha de poder donar-se d'alta a l'aplicació.
R-2	L'usuari ha de poder iniciar sessió a l'aplicació.
R-3	L'usuari ha de poder tancar sessió a l'aplicació.
R-4	L'usuari ha de poder modificar el seu perfil públic.
R-5	L'usuari pot especificar les seves preferències, és a dir, un o més joc del seu interès, el seu rang, les expectatives, disponibilitat i idioma.
R-6	L'usuari ha de poder modificar les seves preferències.
R-7	L'aplicació permet a l'usuari veure tots els jugadors sense cap filtres.
R-8	L'aplicació permet a l'usuari filtrar usuaris per: Joc que juguen, per les expectatives, per disponibilitat i idioma.
R-9	L'aplicació permet a l'usuari crear sala públiques com privades.
R-10	L'aplicació permet a l'usuari escollir el nombre de persones que poden unir-se a la seva sala.
R-11	L'aplicació permet a l'usuari filtrar les sales per: Joc, per les expectatives, per disponibilitat i idioma
R-12	L'aplicació permet a l'usuari unir-se a una sala pública.
R-13	L'aplicació permet a l'usuari expulsar usuaris de la sala sala.
R-14	L'aplicació permet a l'usuari invitar a altres usuaris a unir-se a una sala privada o pública.
R-15	L'aplicació permet a l'usuari abandonar una sala.
R-16	Els usuaris d'una sala poden xatejar entre ells.
<b>Requisits no funcionals</b>	
R-17	Funcionalitats del sistema han de respondre a l'usuari en menys de 4-5 segons.
R-18	Dades crítiques dels usuaris s'han d'emmagatzemar amb un xifratge.
R-19	L'interfaç de l'usuari ha de ser user-friendly, intuïtiva i ràpida d'aprendre.
R-20	L'aplicació ha d'estar desenvolupat pel sistema operatiu Android.
R-21	L'El backend de l'aplicació ha d'estar amb Django.
R-22	L'aplicació ha d'estar desenvolupada amb React Native.

TAULA 1: TAULA DE REQUISITS

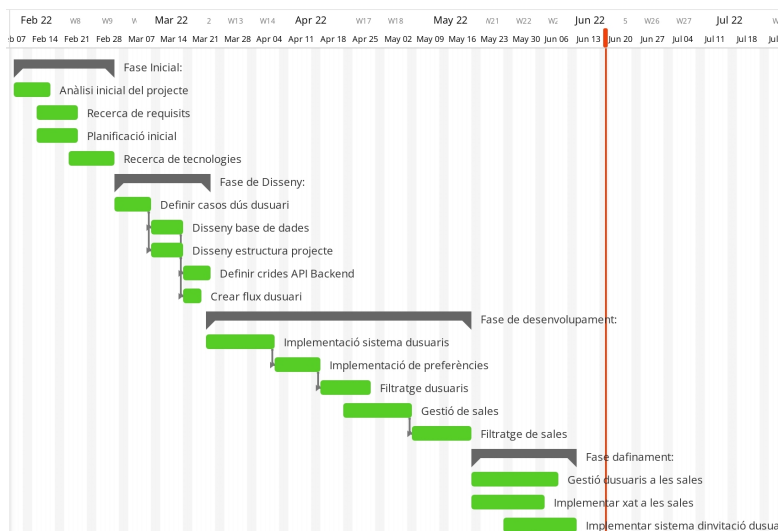


Fig. 12: Diagrama de Gantt

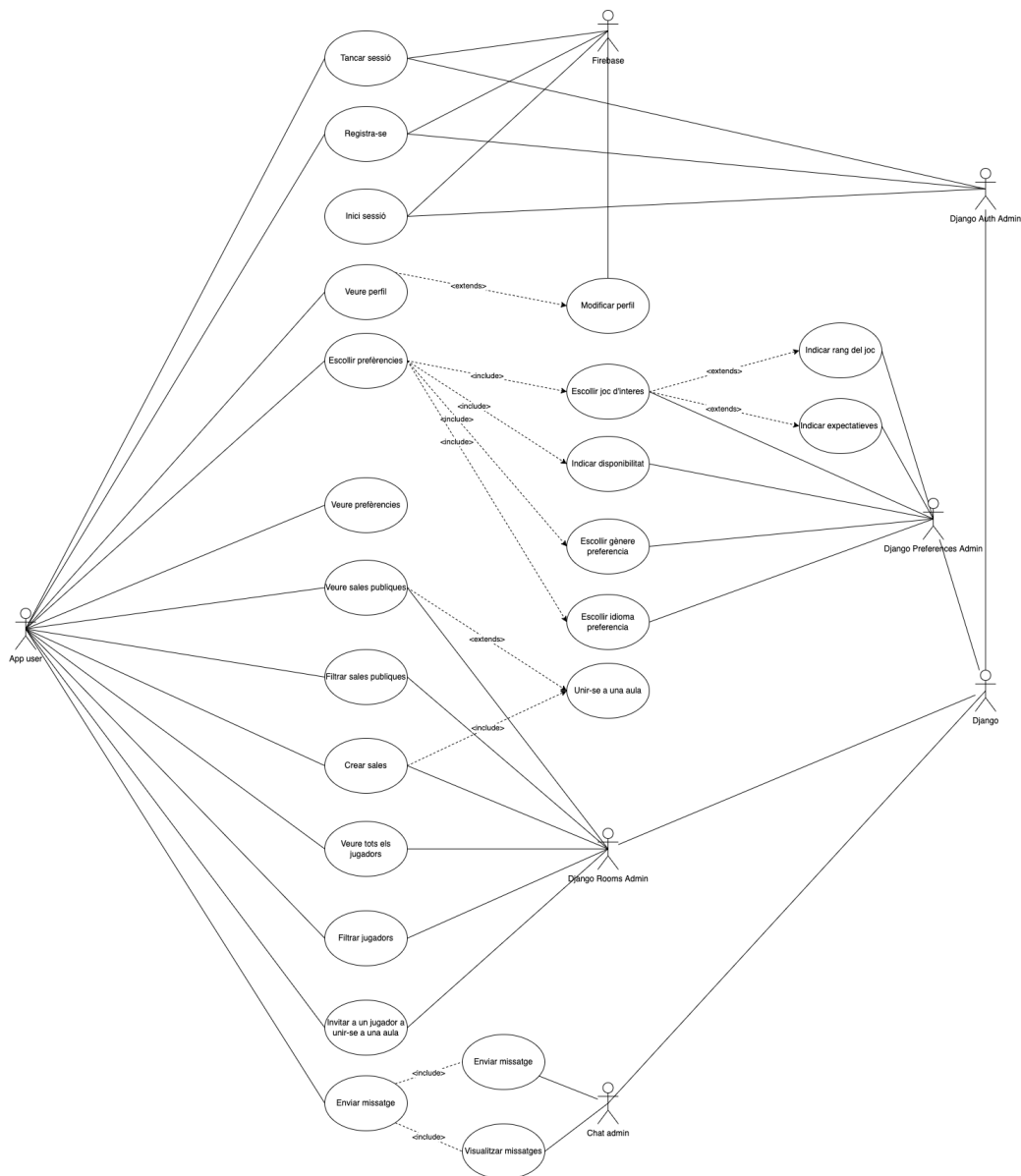


Fig. 13: Diagrama de cas d'ús