

App per recomanar pel·lícules **en grups**

Informe de progrés I

Eugeni Suaña Barranco

Índex:

Planificació:	3
Requisits:	4
Diagrama de casos d'ús:	4
Tecnologies:	4
Metodologia:	5
Etapa de disseny:	6
Disseny base de dades:	6
Prototip:	6
Etapa de desenvolupament:	7
Creació del projecte:	7
Log in i Registre:	8
Home:	9
Perfil i Editar Perfil:	9
Afegir contactes	10
Fonts d'informació i Bibliografia:	11

Introducció:

En aquest document s'explica els avenços realitzats en la primera etapa de desenvolupament del TFG, mostrant els passos que s'han seguit i la metodologia utilitzada. Alhora, s'indica el nivell de seguiment de la planificació proposada en l'informe inicial així com els canvis que s'han aplicat sobre aquesta.

Planificació:

En aquesta segona etapa del projecte, s'han realitzat dos canvis respecte a la planificació que es va proposar en l'informe inicial.

Primerament, un canvi menor que s'ha dut a terme ha sigut corregir un error que hi havia en el diagrama de Gantt original. Aquest error consistia en el fet que les tasques "Perfil i modificar perfil" i "Afegir contactes" no estaven enllaçades quan sí que ho haurien d'estar, fent que les dates d'inici i finalització de les tasques posteriors fossin errònies. Amb aquesta correcció les dates que es mostren en el nou diagrama de Gantt (Figura 1) són les adequades.

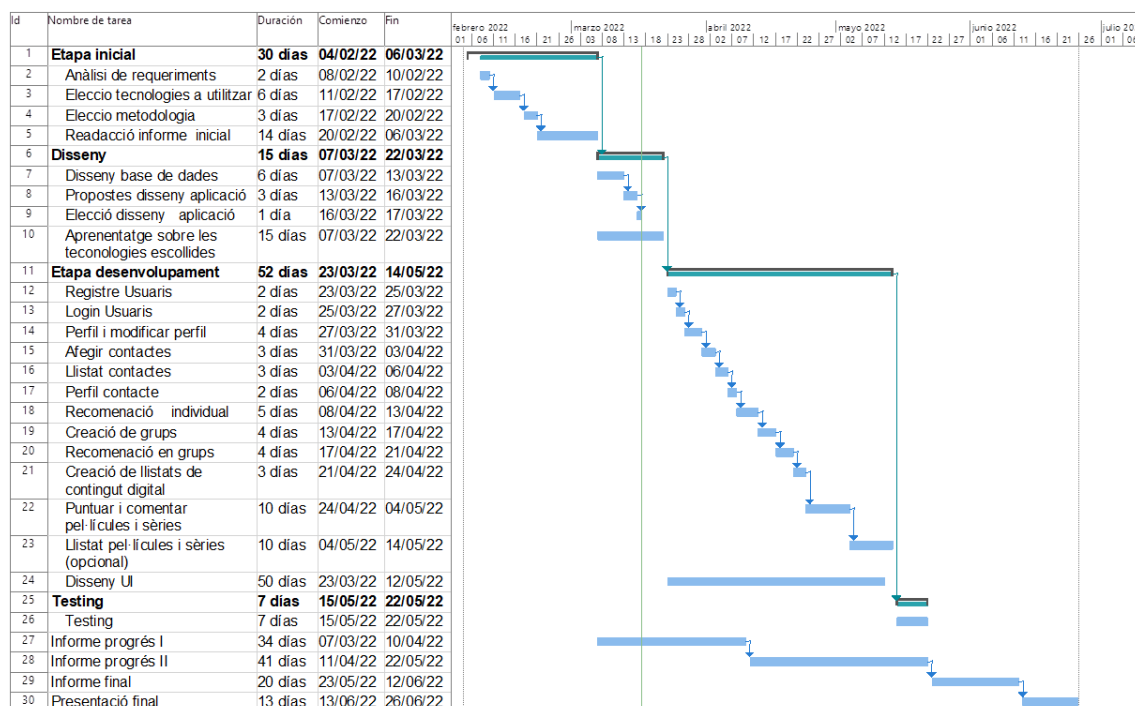


Fig. 1: Diagrama de Gantt corregit

L'altre canvi que s'ha aplicat a la planificació correspon a l'etapa de disseny. Durant el desenvolupament d'aquesta fase, es va optar per ometre les tasques "Propostes disseny aplicació" i "Elecció disseny aplicació" i directament generar un prototip de l'aplicació. Es va escollir aquesta opció perquè es va pensar que d'aquesta manera ja s'establiria el disseny que tindria l'aplicació i, alhora, el prototip també ajudaria a aclarir la funcionalitat que hauria de tenir l'aplicació així com les pantalles que serien necessàries per portar a terme el projecte.

Pel que fa a l'etapa de desenvolupament, de moment s'ha anat seguint la planificació que es tenia proposada, aconseguint els objectius establerts.

Requisits:

El primer que s'ha fet abans de començar amb el desenvolupament de l'aplicació ha sigut fer una anàlisi dels requisits de l'aplicació segons els objectius proposats a l'informe inicial. Aquests requisits es llisten a continuació amb la següent taula:

ID	Requisit
RF-01	L'aplicació ha de permetre registrar usuaris
RF-02	Els usuaris registrats han de poder fer log in i log out a l'aplicació
RF-03	L'usuari ha de poder visitar el seu propi perfil així com el dels seus contactes
RF-04	L'usuari ha de poder modificar el seu perfil
RF-05	L'aplicació ha de permetre agregar o desagregar a altres usuaris com a contactes
RF-06	Al visitar el perfil dels seus contactes, l'usuari ha de poder visualitzar les llistes i opinions creades per aquests
RF-07	L'usuari ha de poder crear, modificar i eliminar llistes de contingut digital
RF-08	L'usuari ha de poder crear, modificar i eliminar opinions sobre contingut digital
RF-09	Els usuaris han de poder recomanar contingut digital als seus contactes o grups de contactes junt amb un missatge descriptiu
RNF-01	L'aplicació serà funcional en el sistema operatiu Android
RNF-02	Per tal de tenir accés a l'aplicació, l'usuari haurà d'estar registrat
RNF-03	Les pantalles de l'aplicació s'han d'adaptar a diferents resolucions
RNF-04	El temps de canvi entre pantalles mai hauria de ser superior a 2 o 3 segons
RNF-05	L'aplicació ha de tenir un temps d'aprenentatge per par de l'usuari menor a 4 hores
RNF-06	L'aplicació ha de proporcionar missatges d'error informatius i orientats a l'usuari
RNF-07	L'actualització de dades a la base de dades no pot tardar més de 2 segons

Taula 1: Requisits de l'aplicació

Diagrama de casos d'ús:

Realitzat l'anàlisi de requisits, s'ha passat a generar el diagrama de casos d'ús (Figura 2) amb l'eina PlantText [1] per mostrar les funcionalitats de l'aplicació basant-se en els objectius i requisits recollits:

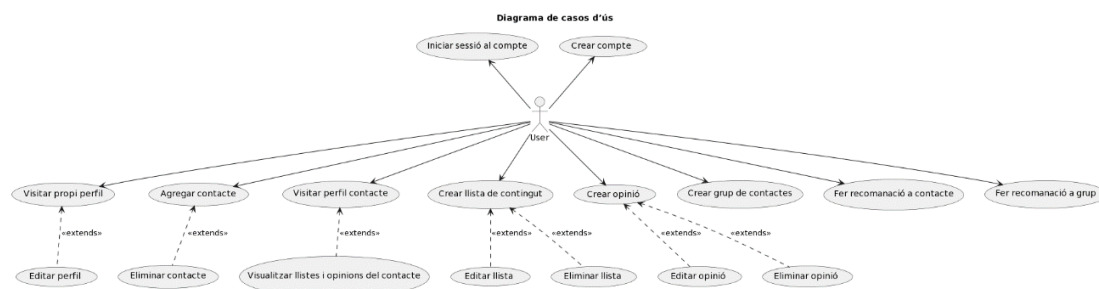


Fig. 2: Diagrama de casos d'ús

Tecnologies:

A l'hora de decidir quines serien les tecnologies, inicialment es va optar per desenvolupar l'aplicació a través de *ReactJS*. Aquesta decisió, però va canviar ràpidament a l'informar-se a través d'internet i veure les moltes facilitats que aportava *Android Studio* a l'hora de

desenvolupar aplicacions per al SO *Android*. És per això que es va acabar decidint que l'aplicació es duria a terme en l'entorn *Android Studio* amb els llenguatges de *Kotlin* i *Java*. [2][3]

Pel que fa al *backend*, es va tenir clar des d'un primer moment que s'utilitzaria *Firebase* de *Google*. Aquesta eina té uns serveis que atorguen grans avantatges, com la facilitat d'implementar un sistema d'autenticació d'usuaris segur o permetre connectar la nostra aplicació amb la base de dades sense necessitat de fer servir una API. A més, *Android Studio* i *Firebase* es complementen molt bé, ja que *Android Studio* ve amb la funcionalitat incorporada de poder connectar amb molta facilitat el teu projecte amb *Firebase*, podent accedir a tots els avantatges que aquesta eina ens proporciona de manera molt senzilla, raó que també va influenciar a l'hora d'escollir *Android Studio* per sobre de *ReactJS*. [4]

L'únic inconvenient que es va trobar amb *Firebase* és que es tracta d'una eina gratuïta fins a cert punt, és a dir, si l'aplicació acabes escalant de manera quantificable (amb més de 10.000 verificacions d'usuari per més o més de 20.000 operacions d'escriptura per dia), segurament s'hauria de pagar per tal de seguir utilitzant els serveis de *Firebase*. [5]

Metodologia:

Com es va comentar a l'informe inicial, la metodologia *agile* escollida va ser la metodologia *Kanban*. Aquesta metodologia s'ha seguit a través del software de gestió de projectes Trello. [6]

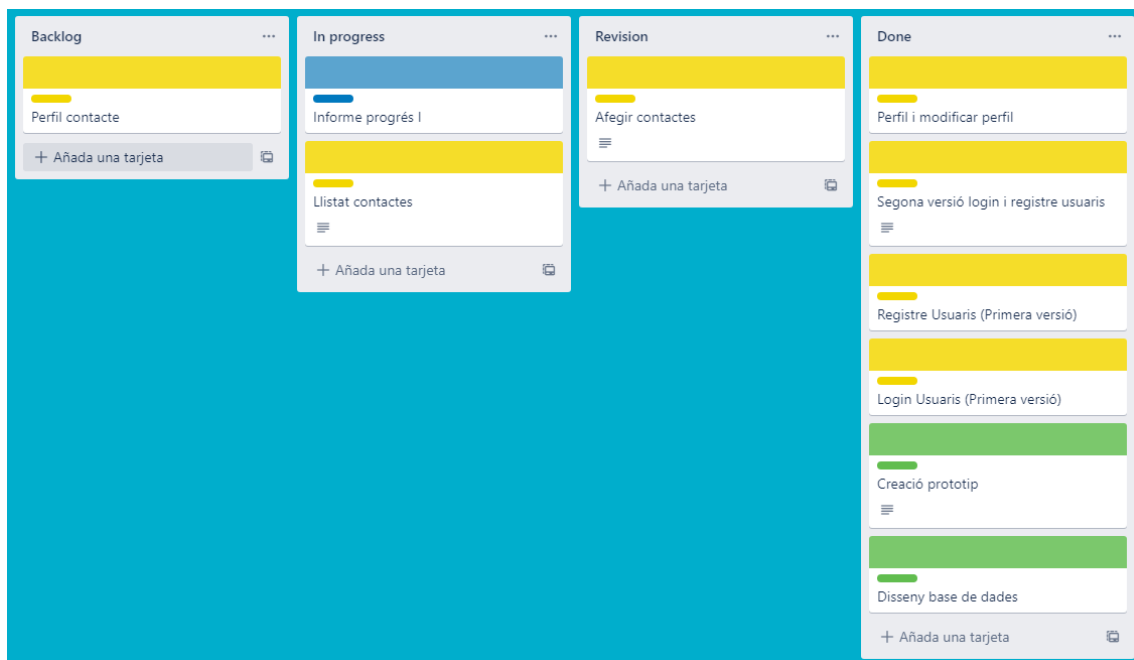


Fig. 3: Tauler Trello

En aquesta eina, com es pot veure a la Figura 3, s'han creat 4 columnes diferents. La primera columna anomenada *Backlog* és la columna on es situen les tasques no començades. A continuació, tenim la columna *In progress* on estan les tasques en les quals s'està treballant actualment. Per acabar, tenim la columna *Revision* on es col·loquen les tasques acabades, però que s'han de revisar per a poder passar-les a la columna *Done* una vegada aprovades.

A part de les columnes, les tasques també s'han classificat per classe. Aquesta classificació s'ha realitzat per colors on el color verd correspon al disseny, el groc al desenvolupament de l'aplicació i el blau a la documentació.

Respecte al control de versions, tal com es va dir a l'informe inicial, a l'eina *GitHub* [7] s'ha anat penjant les noves versions del codi a mesura que s'ha anat desenvolupant, així com la documentació referent al projecte.

Etapa de disseny:

Tenint clars els objectius, requisits i funcionalitats que hauria de tenir l'aplicació, es va procedir a entrar a l'etapa de disseny, on es va especificar el disseny que tindria l'aplicació així com començar a fer les primeres proves amb les tecnologies escollides.

Disseny base de dades:

La base de dades de la nostra aplicació s'encarrega de guardar la informació referent als nostres usuaris, les seves opinions i llistes de continguts, així com els grups i les recomanacions creades pels diferents usuaris. Tenint en compte això es va generar el següent diagrama Entitat Relació (Figura 4) amb l'eina *App Diagrams* [8] que va servir com a esquema per al disseny de la base de dades:

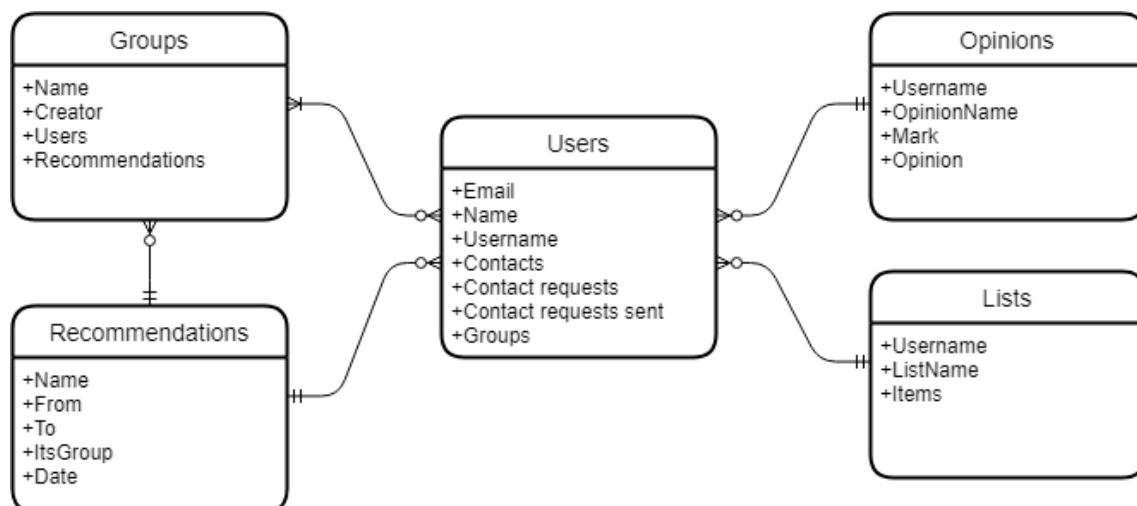


Fig. 4: Disseny de la base de dades

Prototip:

Segons la planificació, al tenir el disseny de la base de dades implementat, tocava realitzar les diferents propostes per al disseny de la base de dades i escollir el que més ens agrada. En canvi, es va optar per directament generar un prototip de l'aplicació amb l'eina *MarvelApp* [9] (Link al prototip: <https://marvelapp.com/prototype/6fb1jh7>) per les raons que ja s'han explicat a l'apartat de Planificació.

A la figura següent es mostren alguns exemples de les pantalles del prototip:

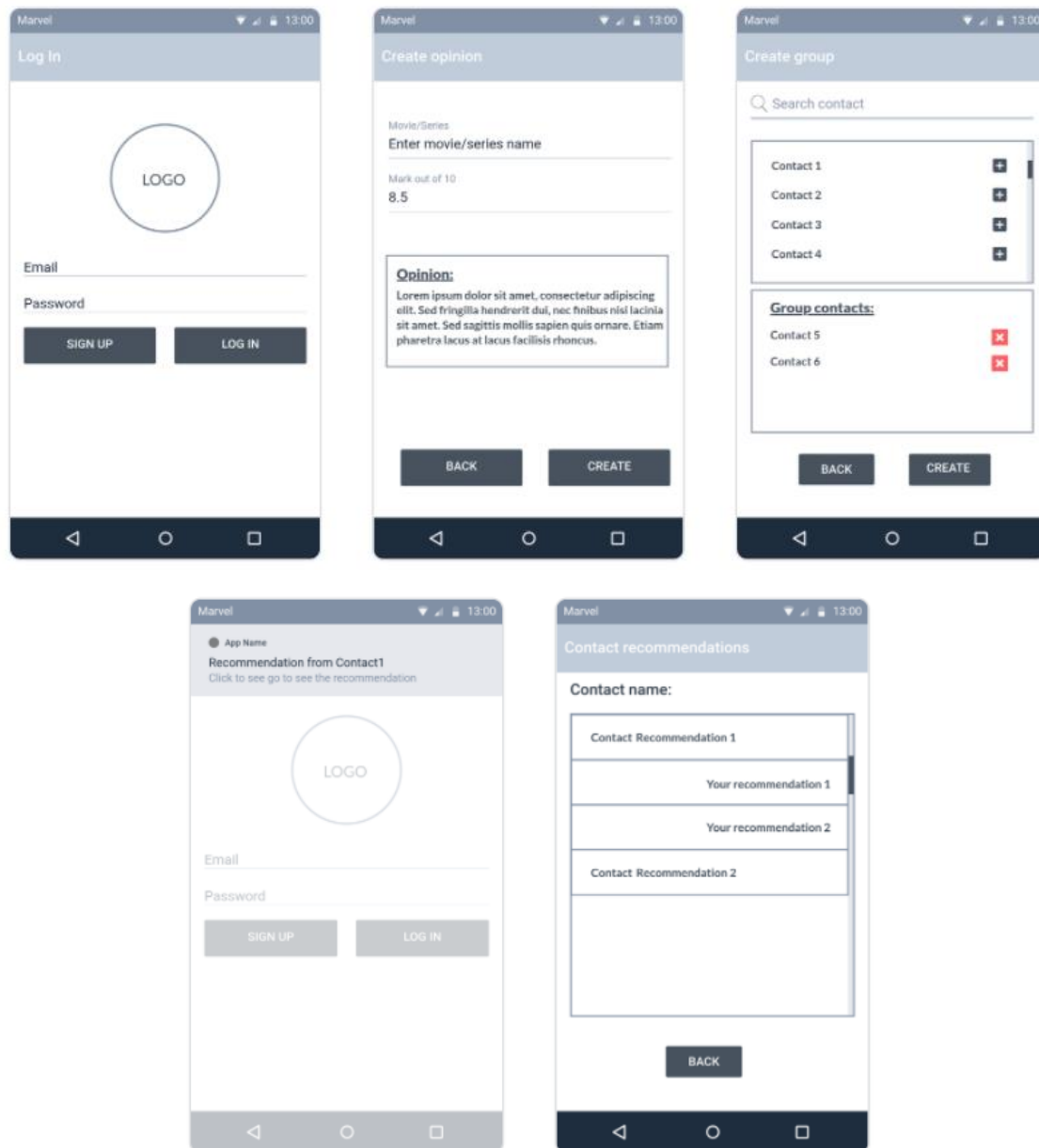


Fig. 5: Pantalles prototip

Etapla de desenvolupament:

Establertes les bases del projecte, tenint el disseny i havent investigat sobre les tecnologies a utilitzar, es va entrar a l'etapa de desenvolupament de l'aplicació.

Creació del projecte:

El primer que es va fer va ser crear el projecte en Android Studio i vincular-lo amb Firebase. Per fer això, simplement es van seguir les instruccions que es troben a la pàgina oficial de Google Firebase. [10]

Connectat el projecte amb Firebase, per tal d'accedir a les funcionalitats de les quals disposa aquesta eina, simplement es va haver d'afegir les dependències corresponents al fitxer Gradle. D'aquesta manera, seguint de nou les instruccions que es troben a la pàgina oficial de Firebase,

es van afegir les dependències necessàries per poder utilitzar la base de dades i el sistema d'autenticació de Firebase.

Tenint ara accés a les eines de Firebase, es va procedir a crear les diferents funcionalitats i pantalles de l'aplicació. Cal mencionar que va ser bastant difícil començar a programar, ja que mai abans s'havia treballat amb Kotlin, però amb els coneixements previs que es tenia de Java, seguint certs tutorials informatius a través de Youtube [11][12], amb l'ajuda de StackOverflow [13] i la documentació oficial de Firebase [4] i Kotlin [14], es va poder començar a desenvolupar les primeres pantalles de manera efectiva.

Log in i Registre:

Primerament, es va començar programant les pantalles de Log in i Registre d'usuaris. Inicialment, va costar implementar aquestes pantalles per les raons mencionades anteriorment, però una vegada obtinguts coneixements sobre el llenguatge Kotlin i l'entorn de programació Andorid Studio, ràpidament es van implementar aquestes dos primeres pantalles de manera efectiva (Figura 6). D'aquesta manera, es va aconseguir que els nous usuaris registrats a l'app es generessin i guardessin de manera correcta a la base de dades.

Per aquesta tasca cal tenir molt en compte l'eina Firebase Authentication. Aquesta eina va proporcionar moltes facilitats i estalviar temps a l'hora de registrar i autenticar usuaris a la nostra aplicació.

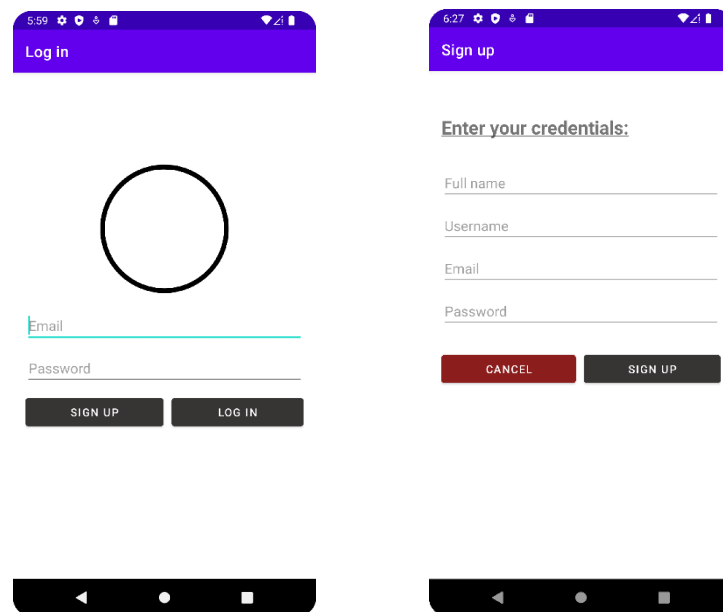


Fig. 6: Pantalles Log In i Registre

S'ha de mencionar també que de moment no s'ha tingut molt en compte el disseny de la UI al desenvolupar les pantalles. El disseny actual és provisional, i es té present canviar-lo i millorar-lo en un futur.

Home:

Una vegada creades aquestes dues pantalles, es va procedir a generar la pantalla Home (Figura 7). Des d'aquesta pantalla, l'usuari té accés a totes les altres pantalles i funcionalitats de l'aplicació, així com fer log out de l'aplicació si ho desitja.

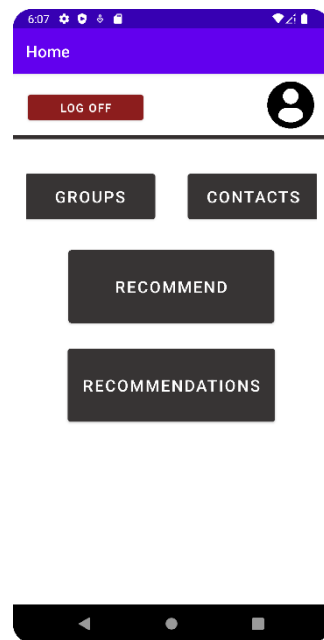


Fig. 7: Pantalla Home

Ara, quan un usuari es registrés per primer cop o fes log in a l'aplicació, seria portat a la pantalla Home.

Perfil i Editar Perfil:

Seguint l'especificat a la planificació, es va procedir a generar les pantalles de Perfil i Modificar Perfil (Figura 8). Inicialment, aquestes van ser senzilles de realitzar amb el que s'havia après creant les anteriors pantalles, però quan es va voler implementar la funcionalitat de que l'usuari tingués la capacitat de penjar una fotografia com a imatge de perfil, va costar més del que es pensava. De nou, amb l'ajuda de StackOverflow, Youtube així com la documentació oficial de Kotlin, es va poder sobrepassar aquests problemes, però va comportar més temps de l'esperat. Per sort, les anteriors pantalles havien requerit menys temps del que s'havia estimat a la planificació inicial i es tenia cert marge.

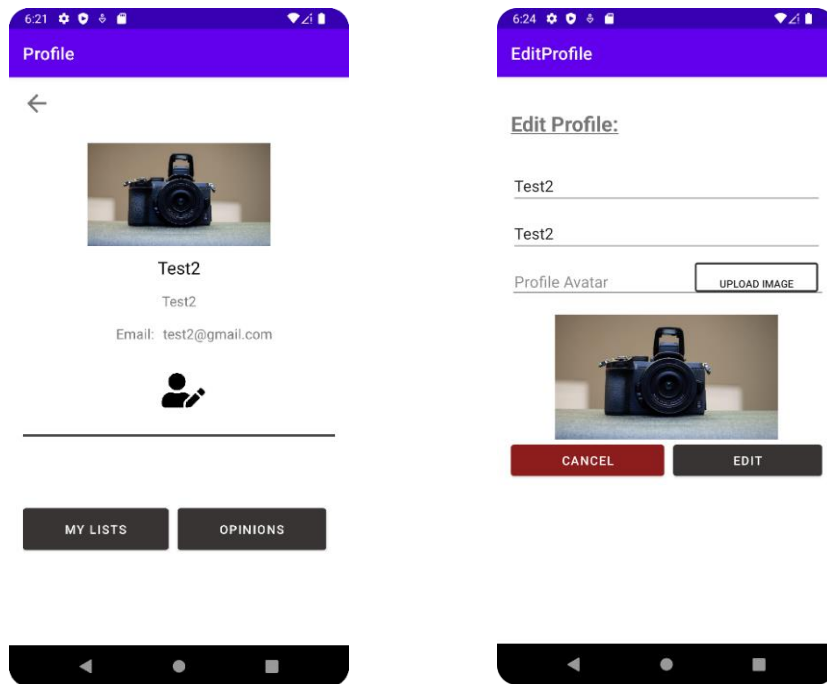


Fig. 8: Pantalles Perfil i Editar Perfil

Afegir contactes

Finalitzades les pantalles del perfil, es va procedir a crear la funcionalitat de poder agregar altres usuaris com a contactes (Figura 9). En aquesta pantalla es va haver d'implementar la funcionalitat d'un cercador on es pogués buscar entre els noms d'usuari dels usuaris registrats a l'aplicació, i poder enviar sol·licituds per tal d'agregar-los a la llista de contactes. Aquesta funcionalitat és la més difícil que s'ha implementat fins al moment perquè, personalment, mai abans s'havia creat un cercador per a una aplicació.

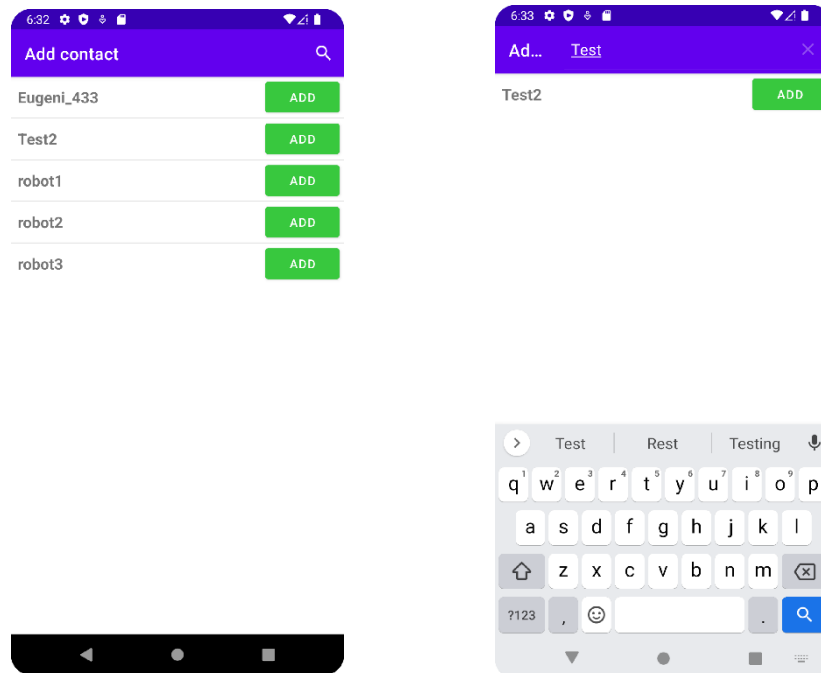


Fig. 9: Pantalla Afegir Contactes

Actualment sembla que aquesta pantalla funciona correctament i s'està treballant en poder mostrar el llistat de contactes.

Fonts d'informació i Bibliografia:

- [1] PlantText, "The expert's design tool", <https://www.planttext.com/> [Accedit: 10-02-2022]
- [2] Platzi (2018) "Cómo crear tu primera app en Android y iOS". [vídeo online] , <https://youtu.be/6MfO1uU8Avk> [Accedit: 12-02-2022]
- [3] Quora, "Should I go with React Native or Java for Android app development?", <https://www.quora.com/Should-I-go-with-React-Native-or-Java-for-Android-app-development> [Accedit: 14-02-2022]
- [4] Firebase, "Documentación", <https://firebase.google.com/docs> [Accedit: 15-02-2022]
- [5] Firebase, "Comienza a usarlo sin costo y luego usa un plan prepagado", <https://firebase.google.com/pricing?hl=es-419> [Accedit: 15-02-2022]
- [6] Trello, "Trabaja mejor, no más duro", <https://www.trello.com/> [Accedit: 08-03-2022]
- [7] GitHub, "Github – Code Repository", <https://github.com/> [Accedit: 23-03-2022]
- [8] App Diagrams, <https://app.diagrams.net/> [Accedit: 09-03-2022]
- [9] Marvelapp, "Rapid prototyping, testing and handoff for modern design teams", <https://marvelapp.com/> [Accedit: 16-03-2022]
- [10] Firebase, "Agrega Firebase al proyecto de Android", <https://firebase.google.com/docs/android/setup> [Accedit: 18-03-2022]
- [11] Youtube, <https://www.youtube.com> [Accedit: 04-02-2022]
- [12] Youtube, "MoureDev by Brais Moure", <https://www.youtube.com/channel/UCxPD7bsocoAMq8Dj18kmGyQ> [Accedit: 15-02-2022]
- [13] Stackoverflow, "A public platform building the definitive collection of coding questions & answers", <https://stackoverflow.com/> [Accedit: 15-03-2022]
- [14] Kotlin, "Kotlin docs", <https://kotlinlang.org/docs/home.html> [Accedit: 15-03-2022]