



# **Proxecto FCT: GameTracker**

## **Memoria**

Rubén Agra Casal

*2º Desarrollo de aplicaciones multiplataforma*

## Índice

1. Introducción.....	3
1.1 Objetivo principal de la app.....	3
1.2 ¿Por qué es útil? ¿Qué problema resuelve?.....	3
2. Objetivos.....	3
2.1 Objetivos prioritarios.....	3
2.2 Objetivos no tan prioritarios o para un futuro.....	3
3. Análisis de requisitos.....	3
3.1 Requisitos funcionales.....	3
3.2 Requisitos no funcionales.....	4
4. Diseño.....	4
4.1 Arquitectura técnica.....	4
4.2 Diagramas.....	4
4.3 Bases de datos.....	4
5. Desarrollo.....	4
5.1 Elección de arquitectura.....	4
6. Pruebas.....	4
7. Manual de usuario.....	4
8. Conclusiones.....	4
8.1 Tareas realizadas.....	4
8.2 Tareas próximas.....	4



## 1. Introducción

### 1.1 Objetivo principal de la app

El objetivo principal de la aplicación sería permitir a los usuarios que puedan gestionar su colección de videojuegos de la forma más sencilla posible, haciendo posible también que puedan compartir su actividad con otros usuarios y proponiendo objetivos/retos para que esto sea más divertido.

### 1.2 ¿Por qué es útil? ¿Qué problema resuelve?

Actualmente este tipo de aplicaciones se están volviendo bastante populares, tenemos de diferentes hobbies como películas (letterboxd), libros (GoodReads) o incluso música (Musicboard) pero no hay ninguna aplicación para la gente que disfruta los videojuegos o por lo menos una aplicación que contente a estos.

## 2. Objetivos

### 2.1 Objetivos prioritarios

- Usuario normal y usuario administrador con funcionalidades específicas
- Creación de lista/colección de juegos (implementar CRUD)
- Buscar información individual de algún juego
- Poder ver listas de otros jugadores
- Mostrar datos individuales (Géneros más jugados, porcentaje de juegos completados, etc.)
- Diseño intuitivo y fácil para el usuario con opciones de personalización

### 2.2 Objetivos no tan prioritarios o para un futuro

- Poner una línea de tiempo en la página principal con la actividad de tus amigos (Ideal que se pudiese sincronizar con cuenta de Steam)
- Poner la opción de chat
- Recomendaciones personalizadas
- Sistema de logros

## 3. Análisis de requisitos

### 3.1 Requisitos funcionales

- Registro de usuarios
- Añadir juegos a la biblioteca
- Eliminar juegos de la biblioteca
- Poner valoración con nota y un texto



- Marcar progreso (porcentaje de completado, trofeos, etc.)

### 3.2 Requisitos no funcionales

---

- Buena estructura de código (MVVM)
- Seguridad

## 4. Diseño

---

### 4.1 Arquitectura técnica

---

- Frontend: Jetpack Compose
- Lenguaje de programación: Kotlin

### 4.2 Diagramas

---

A implementar

### 4.3 Bases de datos

---

A implementar

## 5. Desarrollo

---

### 5.1 Elección de arquitectura

---

El patrón de diseño escogido ha sido de MVVM, ya que es algo parecido a MVC, un patrón con el que estoy ya familiarizado. También he decidido utilizar Jetpack Compose a la hora de hacer las pantallas en vez de XML, ya que es lo que se está utilizando cada vez más.

## 6. Pruebas

---

## 7. Manual de usuario

---

## 8. Conclusiones

---

### 8.1 Tareas realizadas

---

- Definición de idea y priorizar necesidades
- Definición de cantidad de pantallas a necesitar (5)
- Colores y estética general que tendrá la aplicación (Posibilidad de cambiar)

### 8.2 Tareas próximas

---

- Creación de base de datos



- Integrar API externa con información de los juegos
- Utilizar firebase para autenticación de usuarios