



MUSIC HEADS

VÍCTOR BERENGUER MADRID

PROYECTO FINAL DEL CICLO FORMATIVO DE GRADO SUPERIOR EN
DESARROLLO DE APLICACIONES MULTIPLATAFORMA

IES SEVERO OCHOA

2024

Contenido

1. Introducción	4
2. Descripción del proyecto	¡Error! Marcador no definido.
2.1. Ejemplo de Título 2	¡Error! Marcador no definido.
2.1.1. Ejemplo de Título 3	¡Error! Marcador no definido.
2.1.2. Ejemplo Actualización del índice.....	¡Error! Marcador no definido.
3. Herramientas utilizadas.....	6
4. Desarrollo del Prototipo	9
5. Dificultades encontradas	14
6. Evaluación/pruebas del prototipo	15
7. Conclusiones.....	16
8. Bibliografía y recursos utilizados	17
Anexo A. Ejemplo de anexo	18
A.1. Instalación	18
A.2. Configuración	18
Anexo B. Ejemplo de anexo 2	19

1. Introducción

La música, como es obvio, le gusta a muchas personas, incluso les resulta imprescindible para el día a día. Además, muchas de ellas, se interesan por descubrir nueva música, estar al tanto sobre las novedades de sus artistas favoritos u opinar sobre ella con más gente.

Todo esto resulta algo complicado de realizar ya que no existen plataformas concretas que den soporte a estas opciones y siempre suele ser por medio de extras en aplicaciones o webs de forma indirecta. Existen plataformas para valorar música, se pueden utilizar las redes sociales para opinar sobre ella, descubrir las últimas novedades, pero no de forma personalizada.

Con *Music Heads*, se busca que los usuarios puedan buscar personalizar su experiencia musical. Establecer un diario de los álbumes escuchados, añadir nuevos artistas o álbumes, escribir reseñas, lista de proyectos que se quieren escuchar, ranking de discos con mejor valoración, además de lo anteriormente mencionado, entre otras opciones son las características que hacen de *Music Heads* una aplicación musical completa.

Music Heads está desarrollada en *Kotlin* ya que está pensada para dispositivos móviles, además, utiliza la plataforma *FireBase* para almacenar los datos.

2. Objetivos

Principalmente, la aplicación quiere facilitar a los usuarios el camino para desarrollar sus gustos por la música, ya sea encontrando nuevos o desarrollando los ya obtenidos.

2.1. Objetivos específicos

- Registro y *Login* de usuarios para poder acceder o cerrar sesión de la aplicación.
- Visualización de diversas listas tanto globales, como las de cada usuario.
- Adición, edición y eliminación de artistas/álbumes.
- Artistas/Álbumes relacionados por género.
- Calificar el álbum.
- Búsqueda de artista por nombre o de álbum por título.

3. Requisitos de la aplicación

3.1. Requisitos funcionales

Se han establecido una serie de funciones que la aplicación dispondrá. Entre estas funciones, se incluyen tanto aquellas que se han podido alcanzar como las que se tiene la intención de alcanzar en un futuro. Se listarán a continuación:

3.1.1. Inicio de sesión

Al ejecutar la aplicación, lo primero que aparecerá será la pantalla de inicio de sesión con la cual se accederá a los datos del usuario.

- Botón de acceso al registro de usuarios.
- Botón de recuperación de contraseña.

3.1.2. Pantalla principal

Cuando el usuario accede a su sesión, entrará en la pantalla principal de la aplicación.

- Listado de novedades
- Últimas reseñas de proyectos destacables.
- Botón para añadir un nuevo *ítem*.
- Botón de búsqueda.

3.1.3. Barra lateral

Si el usuario desliza de izquierda a derecha en su pantalla, se mostrarán el resto de secciones de la aplicación.

- Sección de inicio.
- Perfil del usuario.
- Listas.
- Lista para escuchar.
- Diario.
- Ajustes.
- Cerrar sesión.

3.1.4. Perfil

Aquí se muestran los artistas o álbumes favoritos seleccionados por el usuario, además de acceder a distintas listas como el diario, la lista de escucha u otros. Es decir, aquello que verían el resto de usuarios.

- Acceso a la lista de álbumes marcados como escuchados.
- Acceso a la lista de álbumes escuchados en forma de diario.
- Lista de proyectos para escuchar.

3.1.5. Listas

Acceso a las distintas listas generales de la aplicación.

- Álbumes mejor valorados.
- Listas por géneros musicales.

3.1.6. Lista para escuchar

La lista de proyectos musicales individual que cada usuario ha marcado para escuchar en un futuro.

3.1.7. Diario

Lista en forma de calendario en la que se muestra el proyecto marcado como escuchado, junto con la nota adjudicada por el usuario y acompañado por el día en el que se ha escuchado.

3.1.8. Ítems

Cuando se accede a cualquier artista o álbum aparecerá la siguiente información sobre dicho ítem:

- Imagen del ítem.
- Título o nombre, fecha de nacimiento o creación, duración y número de canciones en caso de álbum.
- En caso de artista o grupo:
 - Breve biografía.
 - Proyectos musicales en los que ha trabajado.
- En caso de álbum:
 - Listado de canciones dentro del álbum.
 - Nota media dada por los usuarios.
 - Reseñas de los usuarios.
 - Participantes del proyecto (cantantes, productores, escritores, etc...)

- Género musical del proyecto.
- Botón con diferentes funciones como marcar como escuchado, adjudicar una nota, añadir al diario o marcar para escuchar más tarde.

3.1.9. Ajustes de la aplicación

Diversos ajustes de accesibilidad, privacidad, notificaciones o dar de baja la cuenta de usuario.

3.2. Requisitos no funcionales

Estos requisitos son aquellos que no realizan una función directa en la aplicación si no que se encuentran implícitos en la aplicación. Su principal objetivo es mejorar la experiencia del usuario.

3.2.1. Accesibilidad

La aplicación deberá ser intuitiva para los usuarios de forma que resulte sencillo navegar por la misma, al igual que las distintas funciones deberán ser sencillas de usar para el usuario medio.

3.2.2. Seguridad

Aunque la aplicación no requiera de datos de suma importancia, deberá mantener la seguridad del usuario mediante un *Login*.

3.2.3. Compatibilidad

La aplicación no tendrá una complejidad importante, por lo que se buscará la compatibilidad también con los dispositivos menos actuales.

4. Herramientas utilizadas

A continuación, se tratarán las herramientas utilizadas, es decir, los recursos software utilizados. Entre estos se incluyen *IDEs* (Entornos de Desarrollo), sistemas gestores de bases de datos o lenguajes de programación utilizados.

4.1. Software

En este apartado se incluyen las herramientas que se utilizarán para la creación del prototipo. Es decir, los entornos empleados para manipular el código del programa.

4.1.1. Android Studio

Utilizado como Entorno de Desarrollo para el desarrollo de la aplicación. Es una especialización de *IntelliJ IDEA* dedicada a la creación de aplicaciones *Android*. Contiene soporte para el sistema de automatización de construcción de código *Gradle*, plantillas para la creación de diseños de interfaces *Android* y un emulador de un dispositivo *Android* en el que ejecutar las aplicaciones creadas, entre otras muchas características.

4.1.2. Firebase

Se ha utilizado *Firebase* para el almacenamiento de datos y autenticación de la aplicación. *Firebase* proporciona una serie de servicios como almacenamiento de bases de datos, autenticación, además de la integración de diversas aplicaciones como *Android*, *iOS*, *JavaScript*, entre otros. La plataforma está ubicada en la nube incorporada por *Google Cloud Platform*.

4.2. Tecnologías

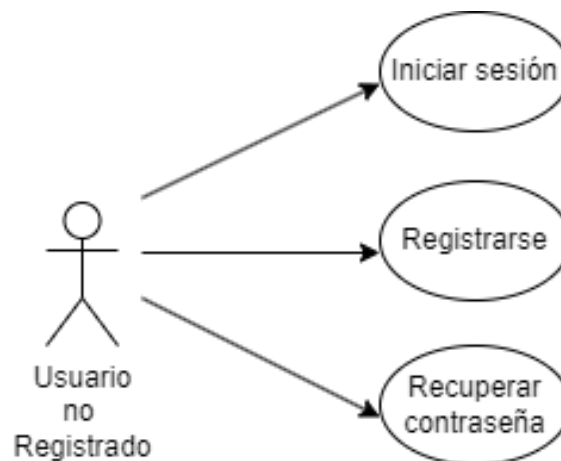
En cuanto a las tecnologías, se incluyen principalmente aquellos lenguajes de programación utilizados, así como las librerías que se emplean para la construcción de la aplicación.

4.2.1. Kotlin

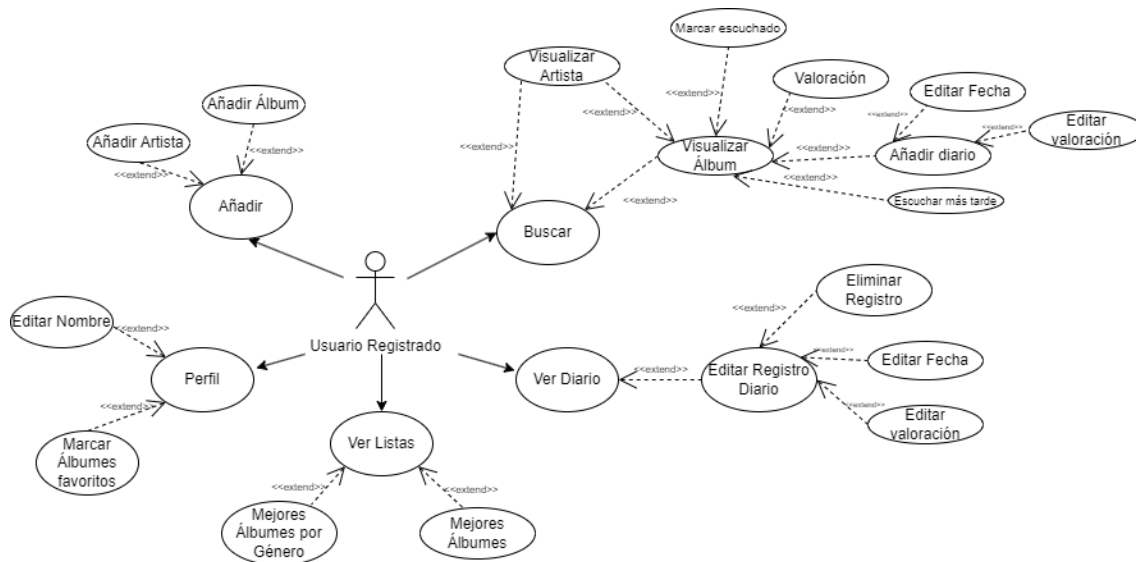
Es el lenguaje de programación elegido para el diseño y la creación del código que dará vida a la aplicación. Según *Google*, es el lenguaje preferido para los desarrolladores de aplicaciones *Android* y está diseñado para integrar de forma total *Java*. Es multiplataforma, estáticamente tipado y con interferencia de tipos.

5. Casos de uso

Los diagramas de casos de uso, proporciona una visión general de las interacciones entre los usuarios o actores y el sistema, ayudando a los desarrolladores a comprender los requisitos funcionales del sistema y a identificar las diferentes acciones que el sistema debe ser capaz de realizar para satisfacer las necesidades de los usuarios. En el caso de este proyecto, se pueden dividir los casos de uso en dos situaciones diferentes:



La primera situación, es cuando un usuario, que no está registrado en la base de datos de la aplicación, accede a ella. Las acciones que podrá realizar están relacionadas con el *Login* a la aplicación, es decir, iniciar su sesión, en caso de estar registrado, registrarse, en caso de no estarlo y reestablecer su contraseña en caso de no recordarla.



Una vez el usuario está registrado y ha accedido a la aplicación, puede interactuar con ella de muchas formas:

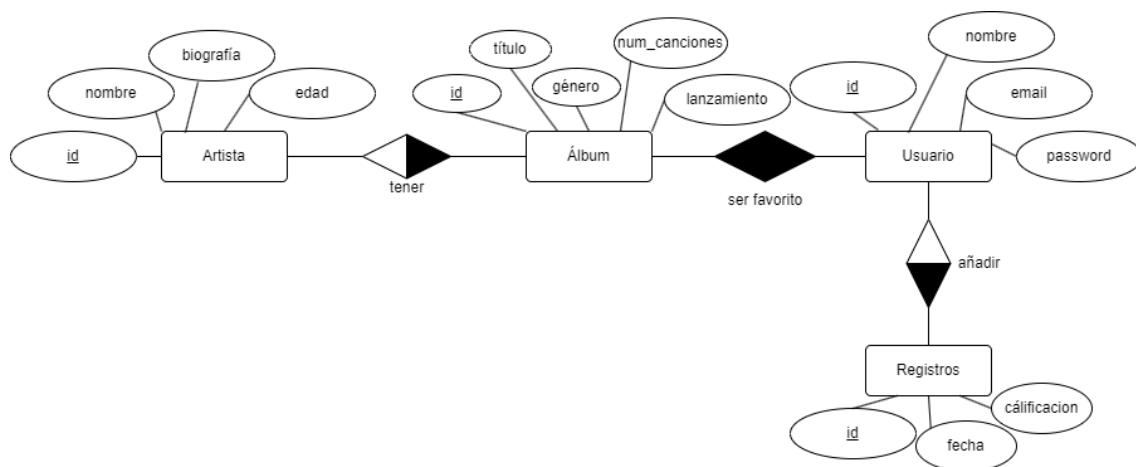
- Buscar un artista o álbum, el cual visualizará posteriormente, además de –en caso de álbum– poder realizar diferentes acciones.
- Visualizar el Diario en forma de lista de registros para, a continuación, editarlos.
- Añadir un Artista o Álbum.
- Acceder a su perfil y realizar diferentes ajustes como Editar su nombre en la aplicación o marcar sus álbumes favoritos en forma de “vitrina”.
- Visualizar diferentes listas anteriormente nombradas.

6. Base de datos

Como se ha mencionado anteriormente, para almacenar los datos de la aplicación, se usará la herramienta de Google, Firebase, más concretamente Cloud Firestore. Esta herramienta hace uso de una base de datos NoSQL, la cual, está principalmente destacada por la flexibilidad y escalabilidad en la nube en la que se almacenarán y sincronizarán los datos necesarios para el posterior desarrollo de aplicaciones web, móviles y de servidor de manera escalable sencilla de usar.

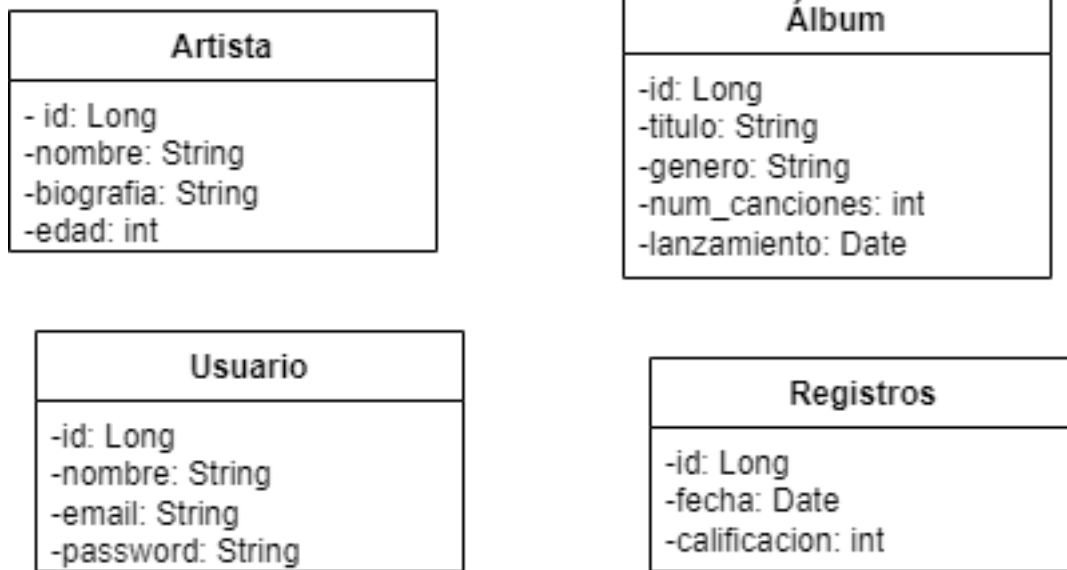
Dado que la base de datos está gestionada por Google, no requiere de un mantenimiento, un servidor externo o inactividad a causa de reinicio. Todo esto, sincronizando los datos en tiempo real entre clientes y servidores.

6.1. Diagrama Entidad – Relación



Aunque Firebase utiliza un modelo de datos NoSQL, lo que significa que no sigue una estructura tabular rígida con filas y columnas, sino que permite almacenar datos en documentos que se organizan en colecciones, representar la base de datos con el modelo Entidad-Relación, queda reflejada de forma más clara.

6.2. Diagrama UML



7. Desarrollo del Prototipo

8. Dificultades encontradas

9. Evaluación/pruebas del prototipo

10. Conclusiones

11. Bibliografía y recursos utilizados

La forma de referenciar un libro es la siguiente: Apellido autor, Iniciales nombre autor, (Año), Título en cursiva, Ciudad y país, Editorial.

La forma de referenciar una web es la siguiente: nombre web, título del artículo, fecha de consulta DDMMAA, URL.

Anexo A. Ejemplo de anexo

A.1. Instalación

A.2. Configuración

Anexo B. Ejemplo de anexo 2