

Introducción

Accenture es una empresa multinacional de servicios profesionales que ofrece soluciones de tecnología, consultoría y outsourcing. La compañía tiene su sede en Dublín, Irlanda, y cuenta con más de 500.000 empleados en todo el mundo, prestando servicios a clientes en una amplia variedad de sectores, como servicios financieros, tecnología, gobierno, energía, salud y telecomunicaciones, entre otros.

El mercado objetivo de Accenture son empresas de todos los tamaños, desde pequeñas y medianas empresas hasta grandes corporaciones, que buscan soluciones tecnológicas y de negocio para mejorar su eficiencia, aumentar su rentabilidad y mantenerse competitivas en un entorno empresarial cada vez más digitalizado.

El sistema organizativo de Accenture se basa en una estructura jerárquica, con múltiples niveles de gestión. A continuación, se describen las funciones de cada nivel:

Nivel ejecutivo: Este nivel está compuesto por los ejecutivos principales de la empresa, incluyendo el CEO, el CFO, el CIO y otros miembros del equipo de liderazgo. Los ejecutivos de este nivel establecen la visión y estrategia de la empresa.

Nivel de gestión sénior: Este nivel incluye a los líderes de cada unidad de negocio, quienes se encargan de supervisar y dirigir las operaciones diarias y la entrega de servicios a los clientes. También están a cargo de la gestión de los recursos humanos, financieros y tecnológicos, y trabajan en estrecha colaboración con los clientes para garantizar que sus necesidades sean satisfechas.

Nivel de gestión intermedia: Este nivel incluye a los gerentes de proyecto y otros líderes de equipos, quienes trabajan en estrecha colaboración con los gerentes senior para garantizar que los proyectos se entreguen a tiempo y dentro del presupuesto. También se encargan de la gestión de recursos humanos y de la resolución de conflictos.

Nivel de personal técnico: Este nivel incluye a los consultores y otros miembros del personal técnico, quienes proporcionan servicios a los clientes y trabajan en proyectos para mejorar la eficiencia y la rentabilidad de las operaciones empresariales. Los técnicos también son responsables de la investigación y desarrollo de nuevas soluciones y tecnologías, y de la implementación de las soluciones en los proyectos.

En general, Accenture se enfoca en la colaboración y el trabajo en equipo, y fomenta la creación de un ambiente de trabajo inclusivo y diverso. La compañía valora la excelencia, la innovación y la entrega de resultados superiores para sus clientes, y busca atraer y retener a los mejores talentos en la industria.

Como parte del plan de formación de la empresa para el alumnado de prácticas, se nos asignó la tarea de desarrollar una aplicación en la que se pudiese en práctica todos los conocimientos y conceptos adquiridos a través de los cursos audiovisuales facilitados, primero en el entorno de Android Studio, y después a través Flutter. Dicha aplicación será la que tomaré como base y punto de partida para la realización de mi proyecto de PFC.

La aplicación estará destinada a dispositivos móviles y consistirá en una especie de “wikipedia” con información sobre distintos aspectos del universo Harry Potter. La aplicación estaría principalmente destinada a aquellas personas que se estén iniciando en la saga y precisen de un acceso rápido a información que les clarifique parte de los elementos y terminología que la envuelven. Incorpora buscadores y filtros específicos para facilitar la búsqueda de toda la información disponible, que se divide en cuatro grandes secciones: Books (Libros), Characters (Personajes), Spells (Hechizos) y Species (Especies), permitiendo también la confección de listas de favoritos para que el usuario pueda destacar la información que le resulte más valiosa o llamativa.

Tecnologías

La App será desarrollada en Android Studio y se nutrirá de dos APIs para extraer toda la información de la que dispondrá.

Android Studio es un entorno de desarrollo integrado (IDE) utilizado para crear aplicaciones móviles nativas de Android. Es la herramienta oficial de desarrollo de Android, desarrollada por Google y está basada en el popular software IntelliJ IDEA. Como cualquier otro IDE nos proporciona una serie de herramientas para facilitar el desarrollo de aplicaciones, móviles en este caso, tales como la edición de código, el depurado, la emulación de dispositivos, la administración de recursos, la integración con sistemas de control de versiones y muchas otras más. Para el desarrollo de esta aplicación, se utilizará el lenguaje de programación Kotlin para escribir la lógica de la aplicación y XML (lenguaje de marcado extensible) para diseñar la interfaz de usuario.

El log in de la aplicación se implementará a través de Firebase. Firebase es una plataforma de desarrollo móvil y web creada por Google que ofrece servicios en la nube para el desarrollo de aplicaciones móviles y web. Firebase proporciona una amplia variedad de herramientas para el desarrollo de aplicaciones, como autenticación, base de datos en tiempo real, almacenamiento en la nube, notificaciones, análisis, entre otros.

El funcionamiento del login en Firebase es muy sencillo. Firebase Auth es el servicio que proporciona la autenticación de usuarios para las aplicaciones. Para utilizar Firebase Auth en una aplicación, primero debes crear un proyecto en la consola de Firebase y configurar la autenticación para la plataforma que deseas utilizar (por ejemplo, Android, iOS, web). Luego, debes integrar el SDK de Firebase Auth en tu aplicación y seguir las instrucciones para configurar la autenticación.

Firebase Auth admite varios métodos de autenticación, incluidos correo electrónico y contraseña, autenticación con proveedores de identidad social como Google y Facebook, y autenticación con proveedores de identidad empresarial como Microsoft Azure AD. Cuando un usuario intenta iniciar sesión en tu aplicación, Firebase Auth verifica las credenciales del usuario y devuelve un token de autenticación si la autenticación es correcta. Este token se puede usar para identificar al usuario y permitir que acceda a los recursos de la aplicación que requieren autenticación.

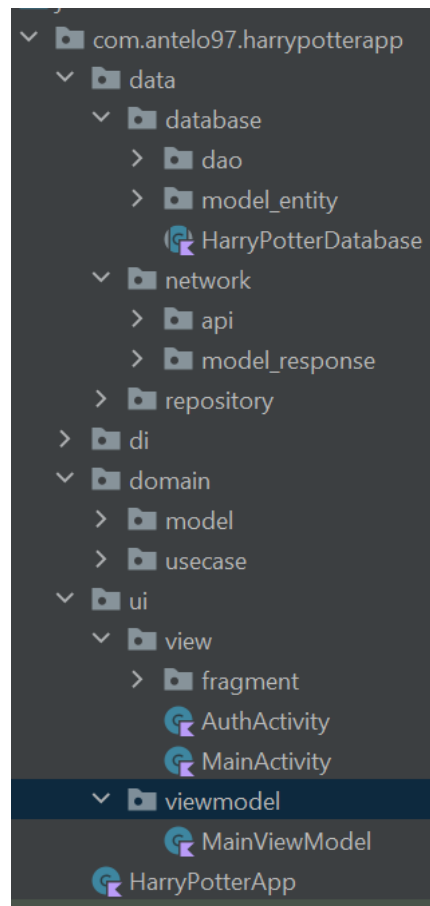
Retrofit es una biblioteca de Android que se utiliza para simplificar el proceso de consumo de API. Retrofit convierte las solicitudes HTTP en objetos Java, lo que facilita su manejo en una aplicación Android. Retrofit utiliza la anotación de Java para definir las solicitudes HTTP, y se encarga de gestionar la comunicación entre la aplicación y el servidor.

Dagger Hilt es un marco de inyección de dependencias para aplicaciones Android. La inyección de dependencias es un patrón de diseño que permite a los desarrolladores separar la lógica de negocio de los componentes de la aplicación. Dagger Hilt proporciona una manera fácil de inyectar dependencias en los objetos de la aplicación, lo que simplifica el código y hace que la aplicación sea más modular y fácil de mantener.

Room es una biblioteca de persistencia de datos de Android que permite a los desarrolladores trabajar con bases de datos SQLite utilizando objetos Java. Room simplifica la gestión de bases de datos en Android, ya que permite a los desarrolladores trabajar con los datos utilizando objetos Java en lugar de SQL. Room también proporciona un alto nivel de abstracción para la gestión de la base de datos, lo que hace que la aplicación sea más fácil de mantener y escalar.

La combinación de estas tres tecnologías puede ser muy poderosa para la construcción de aplicaciones Android. Retrofit puede utilizarse para consumir APIs y proporcionar los datos a la aplicación, Dagger Hilt se utiliza para la inyección de dependencias y Room se utiliza para la gestión de la base de datos de la aplicación.

Arquitectura



MVVM (Model-View-ViewModel) es una arquitectura de software para aplicaciones que se utiliza en el desarrollo de aplicaciones de usuario. Esta arquitectura separa la lógica de presentación de la lógica de negocio, lo que facilita la prueba y el mantenimiento del código. La arquitectura MVVM consta de tres componentes principales:

Model: El modelo representa la lógica de negocio y los datos en la aplicación.

View: La vista representa la interfaz de usuario de la aplicación.

ViewModel: El ViewModel actúa como intermediario entre el modelo y la vista, proporcionando la lógica de presentación y la gestión de los estados.

La meta arquitectura Clean Architecture es una metodología de diseño de software que se enfoca en la separación de las preocupaciones del negocio y la tecnología en la construcción de aplicaciones. Esta metodología se basa en la idea de que el código fuente se divide en capas que representan diferentes niveles de abstracción. Las capas en la arquitectura Clean Architecture son:

Capa de presentación: Esta capa representa la interfaz de usuario y se encarga de la presentación de los datos al usuario.

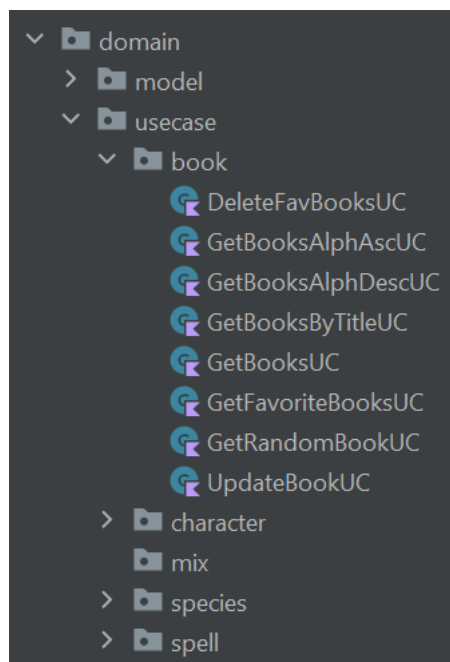
Capa de dominio: Esta capa contiene la lógica de negocio y representa el núcleo de la aplicación.

Capa de infraestructura: Esta capa contiene el código que se encarga de la comunicación con la base de datos, la red y otros servicios externos.

La arquitectura Clean Architecture busca maximizar la independencia del código, facilitando la integración con otras tecnologías y frameworks, además de fomentar el desarrollo guiado por pruebas y la reutilización del código.

Análisis de requisitos

Ejemplo de casos de uso para Book (extensible al resto de secciones)



1. Eliminar libros de favoritos
2. Obtener listado de libros ordenados por título de la A a la Z
3. Obtener listado de libros ordenados por título de la Z a la A
4. Buscar libros por título
5. Obtener el listado de libros completo
6. Obtener el listado de favoritos de libros
7. Obtener un libro aleatorio de los registrados en la BD local
8. Actualizar libros