**TFG**

TRABAJO DE FINAL DE GRADO

I.E.S. BARAJAS

|  |
| --- |
|  |
| Desarrollo de Aplicaciones Multiplataforma |
| Aplicación Android para seguimiento y consulta de información sobre criptomonedas |
|  |
| Android application for tracking and consulting information about cryptocurrencies |
|  |

|  |
| --- |
| Damián Peña-Marín Samaniego  25-11-2024 |

Contenido

[I. Introducción 2](#_Toc178950764)

[II. Abstract 2](#_Toc178950765)

[III. Objetivos del Proyecto 3](#_Toc178950766)

[IV. Metodología de Desarrollo 4](#_Toc178950767)

[V. Análisis del estado del arte 6](#_Toc178950768)

1. Introducción

En el dinámico y volátil mundo de las criptomonedas, la gestión eficiente de las inversiones es crucial para el éxito. Este Trabajo de Fin de Grado presenta una aplicación móvil diseñada para abordar esta necesidad, proporcionando a los usuarios una herramienta integral para el seguimiento y análisis de sus portafolios de criptomonedas.

La aplicación permite a los usuarios registrar sus activos de criptomonedas. A través de la integración con APIs de datos de mercado en tiempo real, la aplicación ofrece una visión actualizada del valor del portafolio, el rendimiento individual de cada criptomoneda y las fluctuaciones del mercado.

Con funcionalidades como gráficos interactivos y noticias del mercado, la aplicación empodera a los usuarios con la información necesaria para tomar decisiones de inversión informadas. Este proyecto no solo aborda un problema real en el creciente mercado de las criptomonedas, sino que también demuestra la aplicación práctica de los principios de desarrollo de software móvil aprendidos durante el grado.

1. Abstract

In the dynamic and volatile world of cryptocurrencies, efficient investment management is crucial to success. This Bachelor´s Degree Final Project presents a mobile application designed to address this need, providing users with a comprehensive tool for tracking and analyzing their cryptocurrency portfolios.

The application allows users to record their cryptocurrency assets. Through integration with real-time market data APIs, the application offers an up-to-date view of the portfolio's value, the individual performance of each cryptocurrency, and market fluctuations.

With functionalities such as interactive charts and market news, the application provides users with the information needed to make informed investment decisions. This project not only addresses a real-world problem in the growing cryptocurrency market, but also demonstrates the practical application of the mobile software development principles learned during the degree.

1. Objetivos del Proyecto

El objetivo principal de este Trabajo de Fin de Grado es desarrollar una aplicación móvil para Android que facilite la gestión y seguimiento de portafolios de criptomonedas, proporcionando a los usuarios herramientas útiles para analizar sus inversiones y tomar decisiones informadas. Para alcanzar este propósito, se establecen los siguientes objetivos específicos:

1. **Registrar y gestionar activos de criptomonedas:** Permitir a los usuarios añadir y gestionar sus inversiones en diversas criptomonedas, a partir de una lista actualizada obtenida mediante la API de CoinGecko.
2. **Integrar datos de mercado en tiempo real:** Implementar la integración con APIs de datos de criptomonedas, como CoinGecko y CoinMarketCap, para ofrecer información actualizada sobre el valor de mercado y las fluctuaciones en tiempo real.
3. **Proporcionar visualización clara de datos:** Ofrecer herramientas de visualización como gráficos interactivos que permitan a los usuarios analizar el rendimiento de sus activos y detectar tendencias en el mercado de criptomonedas.
4. **Facilitar el seguimiento personalizado del portafolio:** Permitir a los usuarios hacer seguimiento del valor de su portafolio en euros, actualizando la información de sus activos de forma fácil y dinámica.
5. **Implementar funcionalidades de autenticación y gestión de cuentas:** Incluir un sistema de inicio de sesión que permita a los usuarios crear y actualizar sus datos de cuenta de manera segura.
6. **Implementar funcionalidades guardado “en la nube”:** Disponer de un almacenamiento en la nube asociado a la cuenta de usuario almacenar y recuperar sus datos en cualquier momento y desde cualquier lugar.
7. **Desarrollar una aplicación intuitiva y accesible:** Diseñar una interfaz de usuario fácil de usar, que permita una navegación fluida y una experiencia positiva para usuarios con distintos niveles de conocimiento sobre criptomonedas.
8. Metodología de Desarrollo

El desarrollo de la aplicación **Cripto Utilidades Damian** se llevó a cabo siguiendo una metodología iterativa e incremental, centrada en la creación de funcionalidades clave de forma progresiva y asegurando la integración continua de componentes. Se empleó un enfoque modular para asegurar que cada parte de la aplicación fuese desarrollada, probada e integrada de manera eficiente, lo que permitió ajustes rápidos en cada iteración. Los aspectos clave de esta metodología incluyen:

1. **Análisis de Requisitos y Definición de Historias de Usuario:** El desarrollo comenzó con la identificación clara de los requisitos, a partir de los cuales se definieron las historias de usuario que guiarían la implementación de las funcionalidades. Cada historia de usuario cubría una funcionalidad específica de la aplicación, como el registro de usuarios, la visualización de la lista de criptomonedas o el almacenamiento de criptomonedas en el monedero. Esto permitió descomponer el proyecto en pequeños componentes manejables.
2. **Diseño e Implementación Modular:** Cada funcionalidad de la aplicación fue desarrollada de manera independiente, siguiendo un enfoque modular. Esto permitió trabajar en diferentes aspectos de la aplicación, como la conexión con APIs externas, la gestión de bases de datos locales y en la nube, y la creación de la interfaz de usuario. Para la implementación, se emplearon patrones de diseño como el Modelo-Vista-ViewModel (MVVM), que separa la lógica de negocio de la interfaz gráfica, facilitando la mantenibilidad del código.
3. **Integración de APIs y Servicios Externos:** La aplicación se conecta a APIs externas como CoinGecko para obtener información en tiempo real sobre criptomonedas, y utiliza CoinMarketCap para proporcionar datos detallados a través de una WebView. Se empleó la librería **Retrofit** para realizar las peticiones dinámicas a la API, lo que permitió cargar y visualizar listas de criptomonedas de forma eficiente en un RecyclerView, combinando con **Glide** para la carga de imágenes.
4. **Gestión de Datos Locales y Remotos:** Para el almacenamiento de datos, se utilizaron dos sistemas complementarios:
   * **SQLite** para gestionar el almacenamiento local de las criptomonedas que los usuarios agregan a su monedero, permitiendo un seguimiento offline.
   * **MySQL** para la gestión de cuentas de usuarios, con una base de datos en la nube que permite la creación, edición y autenticación de usuarios, así como el respaldo de la información de las criptomonedas del monedero.
5. **Diseño y Usabilidad de la Interfaz de Usuario:** Se implementó un diseño de interfaz sencillo e intuitivo, asegurando una experiencia fluida para los usuarios. Se empleó **Data Binding** para vincular los datos observables con los elementos de la interfaz de usuario de forma declarativa. Además, se utilizó un RecyclerView con carga dinámica de elementos para permitir la navegación por grandes volúmenes de datos.
6. **Autenticación y Funcionalidades de Cuenta:** Se desarrolló un sistema de autenticación que permite a los usuarios registrarse, iniciar sesión y editar su información básica. Además, se incorporó la opción de recordar las credenciales (email y contraseña) mediante el almacenamiento seguro de la información utilizando SQLite. Los datos de las criptomonedas almacenadas en el monedero también pueden respaldarse en la nube, lo que permite a los usuarios restaurar su información en diferentes dispositivos.
7. **Pruebas y Validación:** Durante el desarrollo, se realizaron pruebas continuas en dispositivos Android para validar el funcionamiento de cada funcionalidad. Las pruebas incluyeron tanto la verificación de las historias de usuario como la validación del rendimiento y la correcta integración de los datos obtenidos de las APIs. Las pruebas unitarias y de integración se realizaron especialmente en áreas críticas como la autenticación y la gestión de datos.
8. **Uso de Librerías Externas:** El proyecto utilizó diversas librerías de terceros para optimizar el desarrollo, tales como:
   * **Volley** para realizar peticiones HTTP.
   * **Glide** para la carga eficiente de imágenes.
   * **Retrofit** para la gestión de peticiones a la API de CoinGecko. Estas herramientas facilitaron la implementación de funcionalidades clave de la aplicación.
9. Análisis del estado del arte

asdas