

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



Índice

I.	Introducción	2
II.	Abstract	2
III.	Objetivos del Proyecto	3
IV.	Metodología de Desarrollo	4
V.	Tecnologías utilizadas.....	6
VI.	Desarrollo	8
VII.	Casos de uso.....	10

I. 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.

II. 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.

III. 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.

IV. 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.

V. Tecnologías utilizadas

1. Lenguajes

1.1. Java

Para la implementación de la aplicación se ha utilizado **Java**.

Java es un lenguaje de programación orientado a objetos, ampliamente utilizado y de código abierto. Comenzó a comercializarse en 1995 por Sun Microsystems, y desde entonces ha sido adoptado para el desarrollo de diversas aplicaciones, especialmente por su robustez, seguridad y capacidad multiplataforma. Java es fiable y rápido, lo que lo convierte en una opción ideal para el desarrollo de aplicaciones Android, donde se requiere rendimiento y escalabilidad.



Logo de Java

2. Software

2.1. Android Studio

Para el desarrollo del proyecto se ha utilizado Android Studio.

Android Studio es el entorno de desarrollo integrado (IDE) oficial para Android, basado en IntelliJ IDEA. Se seleccionó debido a sus numerosas ventajas, tales como:

- Un editor de código potente y fácil de usar.
- Herramientas de desarrollo de IntelliJ integradas.
- Compilación flexible basada en Gradle, que permite configurar y personalizar las builds.
- Plantillas y herramientas que simplifican la creación de aplicaciones Android.
- Soporte para pruebas y debugging en una variedad de marcos de trabajo.



Logo de Android Studio

2.2. MySQL

Para la persistencia de datos y la autenticación de usuarios se ha utilizado una **base de datos MySQL alojada en la nube**. MySQL es un sistema de gestión de bases de datos relacional basado en SQL, ampliamente utilizado en el desarrollo de aplicaciones web y móviles debido a su eficiencia, fiabilidad y capacidad para manejar grandes volúmenes de datos. Esta solución se ha implementado en el proyecto para gestionar el almacenamiento remoto de la información de los usuarios y sus portafolios de criptomonedas.

Las principales ventajas de utilizar **MySQL en la nube** son:

- **Escalabilidad:** MySQL es capaz de manejar grandes cantidades de datos y escalar según la demanda.
- **Seguridad:** Proporciona múltiples capas de seguridad para la gestión de usuarios y el acceso a la base de datos.
- **Interoperabilidad:** Se puede integrar fácilmente con diferentes tecnologías y plataformas de desarrollo.

- **Disponibilidad:** Al estar alojado en la nube, garantiza que los datos estén disponibles en todo momento y desde cualquier lugar.



Logo de MySQL

Para la autenticación y la gestión de usuarios, se ha implementado un sistema basado en **MySQL**, lo que permite a los usuarios registrarse, iniciar sesión y editar su información. Los datos se almacenan en una tabla específica de usuarios, permitiendo la recuperación y actualización de los mismos. A diferencia de soluciones NoSQL, MySQL permite realizar consultas complejas que optimizan la gestión de grandes volúmenes de información.

VI. Desarrollo

Requisitos

A continuación, se define una lista de los requisitos funcionales que debe cubrir la aplicación:

1. Usuario

1.1. Creación de un usuario

La aplicación permite a los usuarios crear una cuenta nueva ingresando su correo electrónico y una contraseña segura.

1.2. Inicio de sesión

La aplicación permite a los usuarios iniciar sesión utilizando las credenciales creadas previamente (correo electrónico y contraseña).

1.3. Cierre de sesión

La aplicación permite cerrar la sesión del usuario en cualquier momento.



1.4. Edición de datos del usuario

El usuario puede modificar ciertos datos personales, como el correo electrónico o contraseña, desde su perfil, así como poder mostrar la contraseña en texto plano con un “switch”.

1.5. Recordar credenciales

La aplicación permite recordar las credenciales del usuario si selecciona la opción "recordar contraseña" en la pantalla de inicio de sesión.

2. Criptomonedas

2.1. Listar criptomonedas

La aplicación permite al usuario visualizar una lista actualizada de criptomonedas obtenida a través de la API de CoinGecko.

2.2. Visualización de información detallada

La aplicación permite acceder a la información detallada de cada criptomoneda.

2.3. Añadir criptomonedas al monedero

El usuario puede agregar criptomonedas a su monedero personal para hacer un seguimiento de su valor en tiempo real.

2.4. Eliminar criptomonedas del monedero

La aplicación permite eliminar criptomonedas del monedero del usuario.

2.5. Seguimiento del valor de las criptomonedas

El usuario puede consultar el valor actual de las criptomonedas añadidas a su monedero, actualizado en tiempo real.

3. Sincronización y almacenamiento de datos

3.1. Almacenamiento local de criptomonedas

La aplicación almacena los datos del monedero del usuario localmente utilizando SQLite, permitiendo el acceso a la información sin conexión.

3.2. Respaldo en la nube.

El usuario puede respaldar su monedero en la nube, utilizando una base de datos MySQL, para garantizar que los datos no se pierdan.

3.3. Restauración desde la nube

La aplicación permite restaurar los datos del monedero desde la nube en caso de cambio de dispositivo o reinstalación de la aplicación.

4. Funcionalidades adicionales



4.1. Contador de días hasta la entrega del TFG

La aplicación muestra en la pantalla principal un contador con el número de días restantes hasta el 1 de septiembre (fecha de entrega del TFG) (esta característica fue mandada hacer obligatoriamente por la anterior profesora).

4.2. Visualización de criptomonedas en la pantalla principal

La aplicación muestra una vista resumida de criptomonedas seleccionadas en la pantalla principal, facilitando su seguimiento.

VII. Casos de uso

En esta sección se definen los casos de uso de la aplicación.

1. Un usuario se registra en el sistema

El usuario accede a la aplicación y selecciona la opción de "registrarse".

Deberá ingresar su correo electrónico y crear una contraseña segura.

La aplicación valida los datos y registra la cuenta del usuario en la base de datos MySQL.

Si el proceso es exitoso, se muestra un mensaje de confirmación y el usuario es redirigido a la pantalla de inicio de sesión.

2. Un usuario inicia sesión

El usuario abre la aplicación e ingresa su correo electrónico y contraseña en el panel de inicio de sesión.

La aplicación verifica las credenciales con los datos almacenados en la base de datos MySQL.

Si la autenticación es correcta, el usuario accede a la pantalla principal de la aplicación, donde puede visualizar su monedero y los datos del mercado.

3. Un usuario edita los datos de su perfil

El usuario accede al apartado de "Cuenta" en el menú de la aplicación.

En esta sección, se despliega un formulario con los datos actuales del usuario, como el correo electrónico y la contraseña.

Presionará el botón "Actualizar datos" para empezar la edición.

El usuario puede modificar uno o más campos, como cambiar su dirección de correo



o actualizar su contraseña.

Al finalizar, selecciona la opción "Actualizar datos".

La aplicación valida los datos y actualiza la información del usuario en la base de datos MySQL en la nube.

Una vez completada la actualización, se muestra un mensaje de confirmación indicando que los cambios han sido guardados correctamente.

4. Un usuario cierra sesión

El usuario accede al menú principal de la aplicación.

Selecciona la opción de "cerrar sesión".

La aplicación elimina las credenciales almacenadas en la sesión activa y redirige al usuario a la pantalla de inicio de sesión.

5. Un usuario añade una criptomoneda al monedero

El usuario accede al apartado de "Lista de Criptomonedas".

Selecciona una criptomoneda de la lista obtenida a través de la API de CoinGecko.

La aplicación muestra un formulario donde el usuario puede ingresar la cantidad de criptomonedas que posee.

Al confirmar, la criptomoneda se agrega al monedero del usuario y los datos se guardan en la base de datos SQLite local.

6. Un usuario edita una criptomoneda en su monedero

El usuario accede al apartado "Monedero" donde puede ver las criptomonedas que ha añadido.

Selecciona la criptomoneda que desea modificar.

La aplicación despliega un formulario donde el usuario puede cambiar la cantidad de la criptomoneda en su monedero.

Al guardar los cambios, la aplicación actualiza los datos tanto en la base de datos SQLite local como en la base de datos MySQL en la nube si el usuario está sincronizado.

7. Un usuario elimina una criptomoneda del monedero

El usuario accede al apartado "Monedero" y selecciona una criptomoneda que desea eliminar.

La aplicación solicita confirmación antes de proceder.

Una vez confirmada, la criptomoneda se elimina del monedero del usuario, y los



datos se actualizan en la base de datos local y en la nube si la sincronización está habilitada.

8. Un usuario respalda su monedero en la nube

El usuario accede a la opción de sincronización en el menú de la aplicación.

Selecciona la opción "Respalda en la nube".

La aplicación sube los datos del monedero a la base de datos MySQL en la nube.

Una vez completado, se muestra un mensaje de éxito indicando que los datos han sido respaldados correctamente.

9. Un usuario restaura su monedero desde la nube

El usuario accede al menú de la aplicación y selecciona "Restaurar desde la nube".

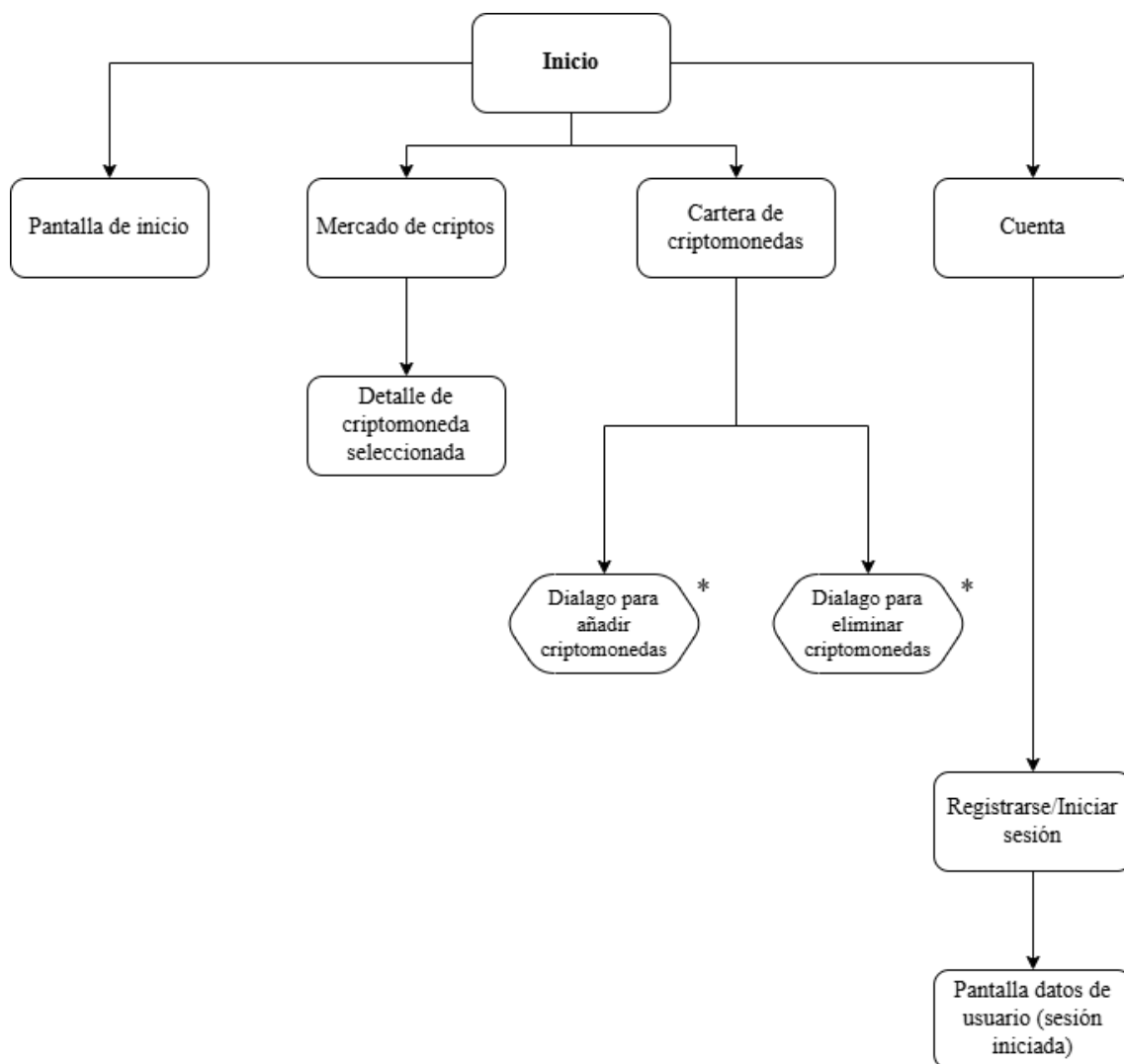
La aplicación verifica la existencia de un respaldo previo y descarga los datos almacenados en la base de datos MySQL.

Los datos del monedero se actualizan en la base de datos SQLite local y se muestra un mensaje confirmando la restauración exitosa.

VIII. Diseño de pantallas

1. Diagrama de flujo de pantallas

A continuación, se muestran las diferentes pantallas de la aplicación y una breve explicación de su funcionamiento. En primer lugar, se muestra un diagrama en el que se define el flujo de las pantallas.

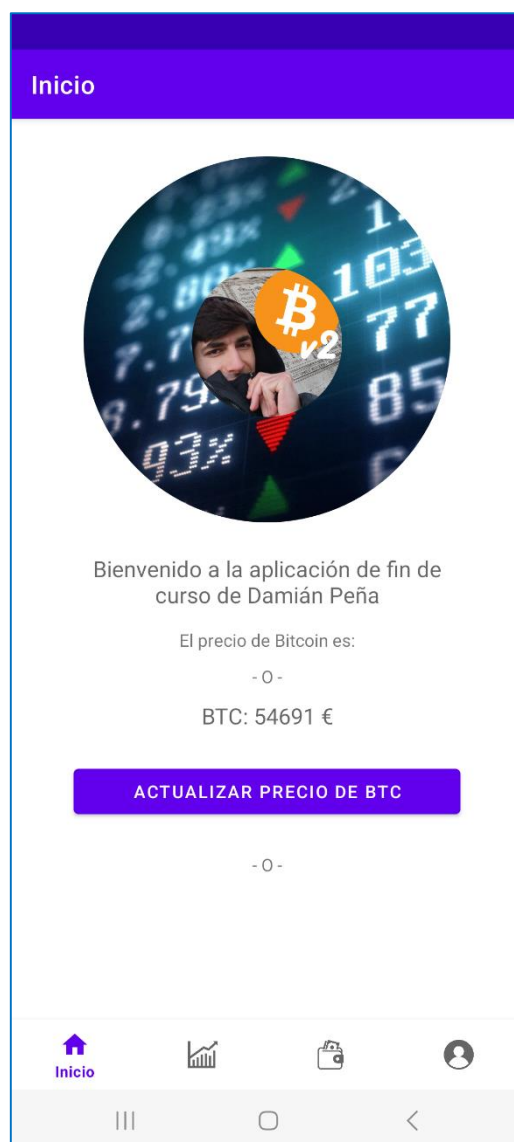


* Los bloques hexagonales indican que no se trata de pantallas completas sino cuadros de dialogo

Diagrama de flujo de las pantallas de la aplicación

2. Pantalla de Inicio

La pantalla de inicio de la aplicación muestra un diseño centrado en la criptomoneda Bitcoin.



En la parte superior, se encuentra el título “Inicio”, indicando que esta es la página principal de la aplicación.

En el centro de la pantalla, hay una representación gráfica animada para hacer la pantalla de inicio más agradable a la vista.



Debajo de los elementos gráficos, hay un texto que da la bienvenida al proyecto de fin de curso.

Más abajo se muestra el precio actual de Bitcoin en euros, actualizado a través de la API o el último valor conocido en caso de ejecutar la aplicación sin conexión a internet.

Debajo hay un botón dedicado a actualizar el precio de bitcoin manualmente, el cual mostrará mensajes tipo Snackbar cuando se presione para indicar el estado de la actualización

En la parte inferior de la pantalla, hay iconos de navegación típicos de una interfaz de aplicación. Cada de estos iconos estará resaltado, indicando cual es la vista actual. Este flujo de pantallas está diseñado para ofrecer una experiencia sencilla y rápida para el usuario, permitiéndole acceder fácilmente a todas las funcionalidades de la aplicación criptomonedas.

3. Pantalla de

4. As

5. As

6. Das

7. Da

8. S

9. ads

IX. A