PROYECTO:
CLIMAXPRESS

Lucas Iulian Colban Tiganescu
Programación de Servicios Y Procesos
CFGS Desarrollo de Aplicaciones Multiplataforma
2ºDAMD 2024-2025
CPIFP Los Enlaces

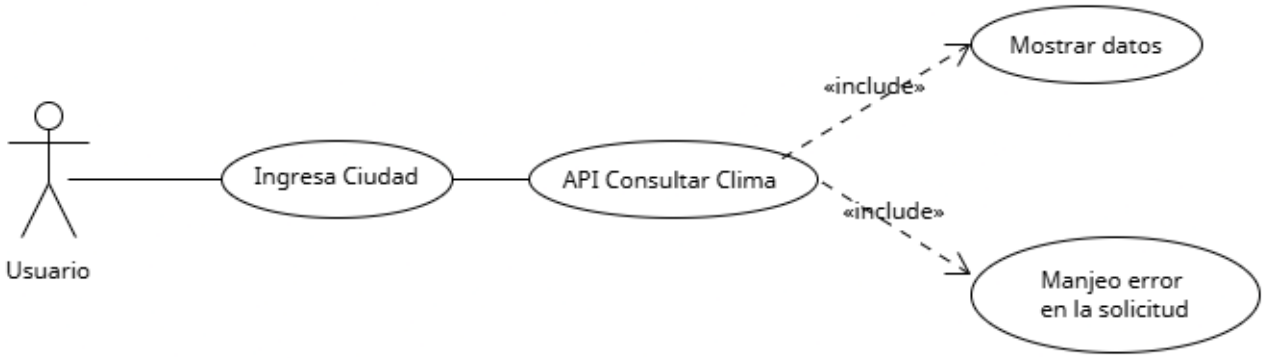
ÍNDICE DE CONTENIDOS

RESUMEN EJECUTIVO DE CLIMAXPRESS

Sección	Descripción
Descripción App	Aplicación móvil desarrollada para la plataforma Android que permite realizar la consulta del clima de cualquier ciudad en tiempo real utilizando la API de OpenWeather.
Objetivo	Desarrollar una herramienta de consulta meteorológica intuitiva de usar mediante un proceso de búsqueda de dicha información eficiente y directo
Tecnologías	➢ **Lenguaje:** Java ➢ **IDE:** Android Studio ➢ **Librerías Relevantes:** Retrofit, Gson ➢ **API:** OpenWeather ➢ **Arquitectura:** Cliente-Servidor (REST API).
Funcionalidades	➢ Búsqueda de clima por ciudad especificada por el usuario. ➢ Manejo de solicitudes empleando manejo de hilos en segundo plano con la librería Retrofit. ➢ Empleo de la librería Gson para la conversión de datos JSON a objetos Java ➢ Manejo básico de errores e información al usuario.
Flujo de la Aplicación	1. El usuario ingresa una ciudad. 2. Se envía una request a OpenWeather con Retrofit. 3. OpenWeather responde con datos JSON, que son procesados por la librería Gson. 4. Se muestra los datos solicitados por pantalla al usuario.
Beneficios	➢ Rápida integración con API REST. ➢ Experiencia fluida gracias al manejo de hilos asíncronos. ➢ Código modular y escalable para futuras mejoras.
Section	**Description**
App Description	Mobile application developed for the Android platform that allows real-time weather queries for any city using the OpenWeather API.
Objectives	Develop an intuitive weather query tool that provides an efficient and direct search process for this information.
Technologies	➢ **Language:** Java ➢ **IDE:** Android Studio ➢ **Relevant Libraries:** Retrofit, Gson ➢ **API:** OpenWeather ➢ **Architecture:** Client-Server (REST API).
Functionalities	➢ Search for weather by city specified by the user. ➢ Handle requests using background thread management with the Retrofit library. ➢ Use of the Gson library for converting JSON data to Java objects. ➢ Basic error handling and providing information to the user.
Application Flux	1. The user enters a city. 2. A request is sent to OpenWeather using Retrofit. 3. OpenWeather responds with JSON data, which is processed by the Gson library. 4. The requested data is displayed on the screen to the user.
Benefits	➢ Quick integration with REST API. ➢ Smooth experience thanks to asynchronous thread management. ➢ Modular and scalable code for future improvements.

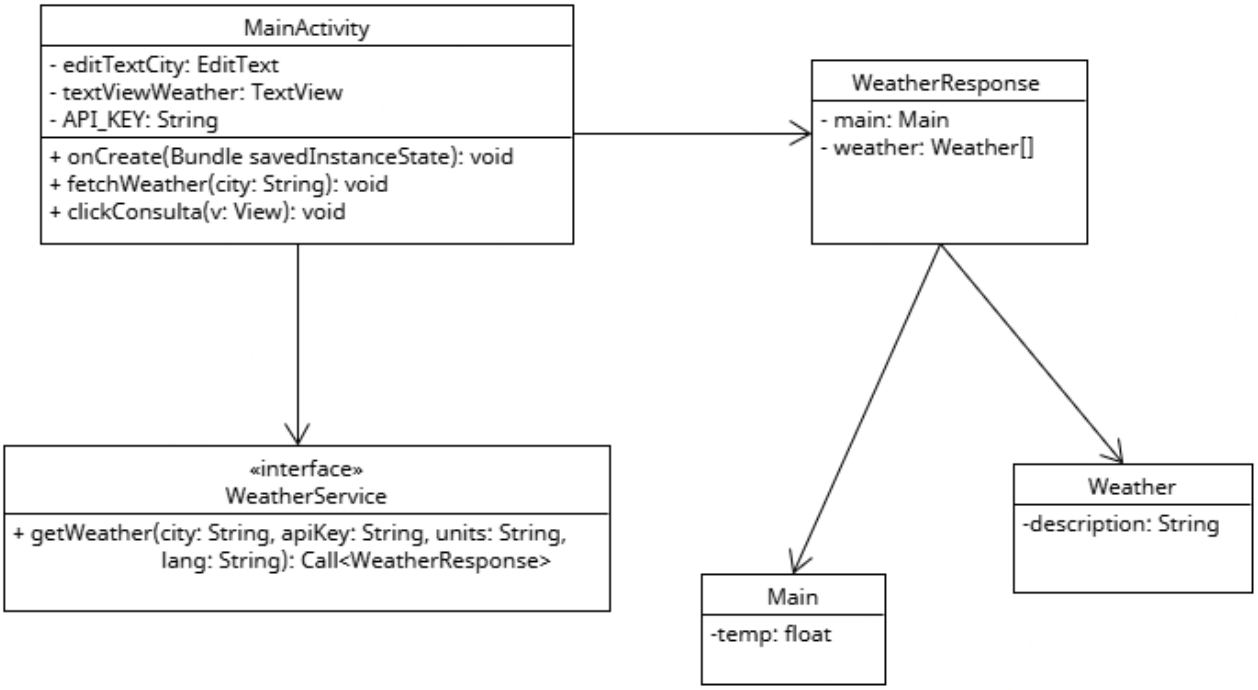
ANALISIS Y DISEÑO

Diagrama de casos de uso



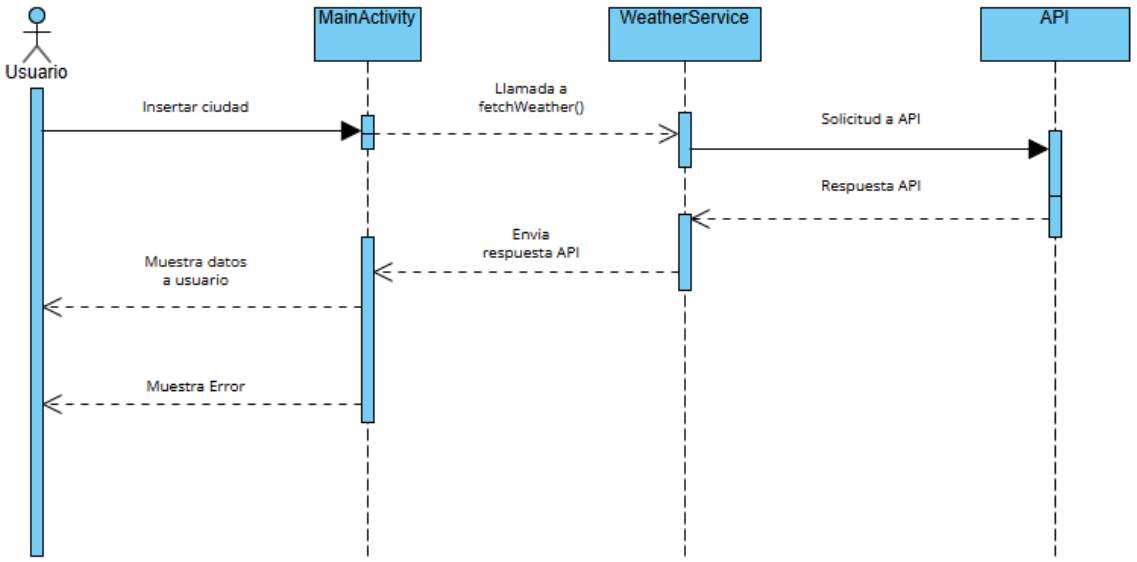
El diagrama se compone de un único actor, que es el usuario que emplea la aplicación, ingresando la ciudad que desea consultar. La API responderá al actor con los datos climatológicos de dicha ciudad o con un mensaje de error si ha habido algún error.

Diagrama de clases



El diagrama de clases compone de una clase principal, **MainActivity**, que gestiona la interfaz del usuario y donde se solicita la request a la API, que junto con la clase interfaz **WeatherService** y la clase **WeatherResponse** con subclases **Main** y **Weather** para gestionar la información recogida, se realiza la llamada la API y se procesa la información recibida para mostrársela al usuario.

Diagrama de flujo



El diagrama de flujo refleja el proceso llamando a la API y el procesado de la respuesta, todo ello realizado de forma asíncrona en segundo plano para evitar bloqueos en la interfaz de la aplicación.

Mockup

CONCLUSIONES

En lo que respecta al proyecto, me ha parecido muy interesante aprender como obtener datos a través de una API, y procesarlos, al igual que descubrir librerías cuya existencia desconocía, como son Retrofit y Gson. Considero que el proceso de aprendizaje ha sido el adecuado y que la implementación de la aplicación es correcta.

En lo que respecta a mi persona, sí que siento cierta frustración, debido a una mala organización personal, no he podido realizar un proyecto tal vez más elaborado, con funcionalidades adicionales y una experiencia de usuario más completa.

Igualmente, debido al interés que me ha generado este proyecto, no descarto la posibilidad de investigar en un futuro un desarrollo más completo de la aplicación.

BIBLIOGRAFIA/WEBGRAFIA

SQUARE. *Retrofit*. Disponible en: https://square.github.io/retrofit/

OPENWEATHERMAP. *OpenWeatherMap API*. Disponible en: https://openweathermap.org/api

CREATELY. *Tutorial del diagrama de secuencia*. Disponible en: https://creately.com/blog/es/diagramas/tutorial-del-diagrama-de-secuencia/

STACK OVERFLOW. *Questions*. Disponible en: https://stackoverflow.com/questions