



Universidad Don Bosco

Investigación aplicada 02

DSM – G01T

- Alfredo Steven Aguilera Salazar. AS240144
- William Adonay Bonilla Pérez. BP220765
- Christian Javier Rosales Palacios RP241102
- Ingeniero Juan Carlos Menjívar Ramírez
- Técnico en Ingeniería en Computación
- 30 de septiembre de 2025
- Enlace del repositorio:
https://github.com/Chris8881/Investigacion02_DSM.git

Introducción

En el ámbito del desarrollo de los conceptos más importantes y utilizados en la actualidad es el de las **API REST**. Una **API (Application Programming Interface o Interfaz de Programación de Aplicaciones)** es un conjunto de definiciones, protocolos y herramientas que permiten la comunicación entre diferentes aplicaciones o sistemas. Dicho de otra forma, una **API** actúa como un puente que facilita el intercambio de datos y funcionalidades sin que las aplicaciones involucradas necesiten conocer en detalle cómo están implementadas internamente.

Por su parte, **REST (Representational State Transfer)** es un estilo de arquitectura de software que define un conjunto de principios y restricciones para diseñar servicios web escalables y eficientes. Cuando hablamos de una **API REST**, nos referimos a una interfaz que sigue estos principios, permitiendo que los clientes (como aplicaciones móviles, páginas web o sistemas externos) puedan interactuar con los recursos de un servidor de manera sencilla y estandarizada, principalmente a través del protocolo **HTTP**.

Lo que caracteriza a una API REST es que los recursos se representan mediante **URL (Uniform Resource Locators)**, y sobre ellos se realizan operaciones utilizando los métodos estándar de HTTP:

- **GET** para consultar información.
- **POST** para crear un nuevo recurso.
- **PUT** o **PATCH** para actualizar recursos existentes.
- **DELETE** para eliminarlos.

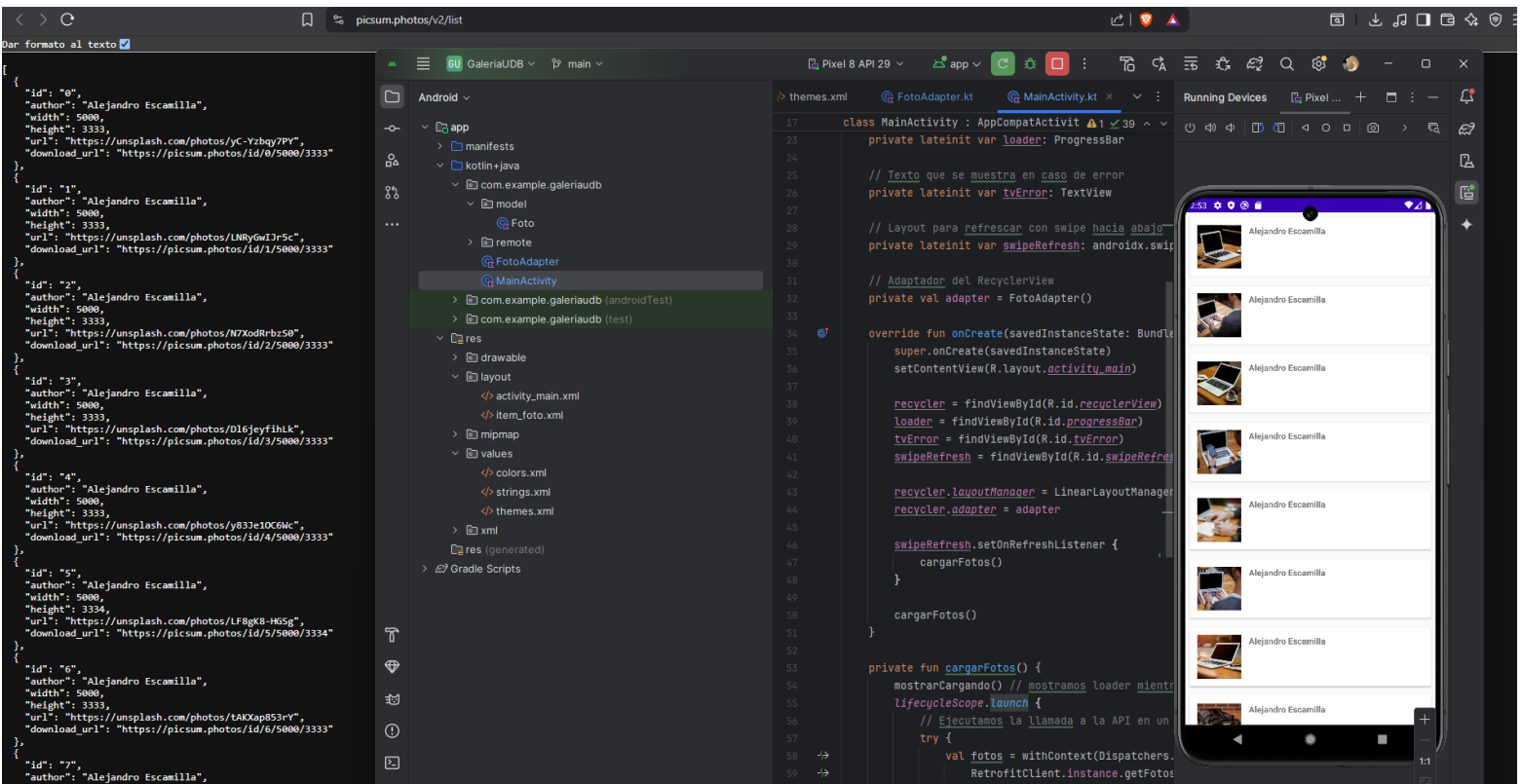
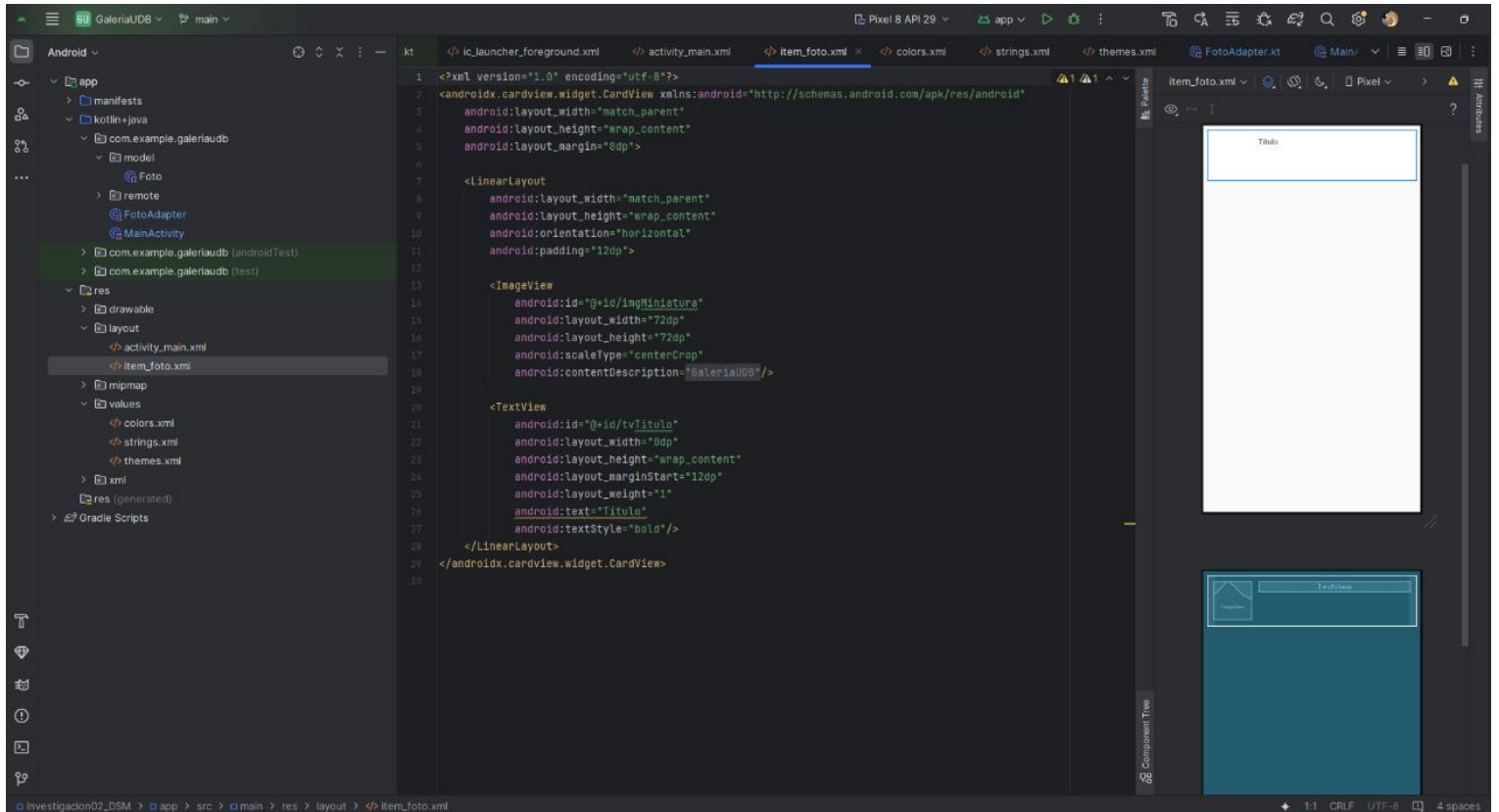
Cada respuesta del servidor se devuelve generalmente en un formato ligero y ampliamente utilizado como **JSON (JavaScript Object Notation)**, lo que facilita la integración con distintos lenguajes de programación y plataformas.

Breve explicación de la API utilizada

La API que usamos en el proyecto se llama **Picsum Photos**. Es un servicio gratuito en internet que nos proporciona fotos de prueba junto con información básica de cada una, como el nombre del autor, el tamaño de la imagen y un enlace para descargarla.

En nuestro caso, utilizamos el endpoint **<https://picsum.photos/v2/list>**, que devuelve una lista en formato **JSON** con varias fotos. De esa respuesta usamos principalmente dos datos: el autor y la **URL** de la imagen, la cual mostramos con la ayuda de la librería **Glide** para que aparezca correctamente en el **RecyclerView**.

Capturas de pantalla de la aplicación funcionando



Conclusión

La realización de este proyecto permitió comprender de manera práctica y aplicada el funcionamiento de las **API REST** y su integración dentro de aplicaciones **Android**. A través del uso de librerías especializadas como **Retrofit** y **Glide**, se logró consumir de forma eficiente un servicio externo, en este caso, la **API** pública **Picsum Photos** y transformar las respuestas en **objetos JSON** que posteriormente se mostraron dinámicamente en la interfaz mediante un **RecyclerView**.

Uno de los aspectos más relevantes fue aprender a manejar adecuadamente los diferentes estados de la aplicación, como la carga inicial de datos, la detección de errores de conexión y la presentación de mensajes claros al usuario en situaciones adversas. Esta gestión no solo incrementa la robustez de la aplicación, sino que también mejora la experiencia de uso.

Asimismo, la implementación de un diseño adaptable con listas dinámicas y la capacidad de refrescar la información reforzaron el entendimiento de conceptos clave en el desarrollo móvil, como el manejo de adaptadores, la vinculación de datos en tiempo real y el consumo eficiente de recursos remotos.

El proyecto permitió consolidar conocimientos teóricos sobre las **API REST** y llevarlos a un entorno práctico en **Android**, logrando cumplir los requisitos funcionales propuestos y fortaleciendo competencias fundamentales en el desarrollo de aplicaciones modernas, escalables y orientadas al consumo de servicios web.