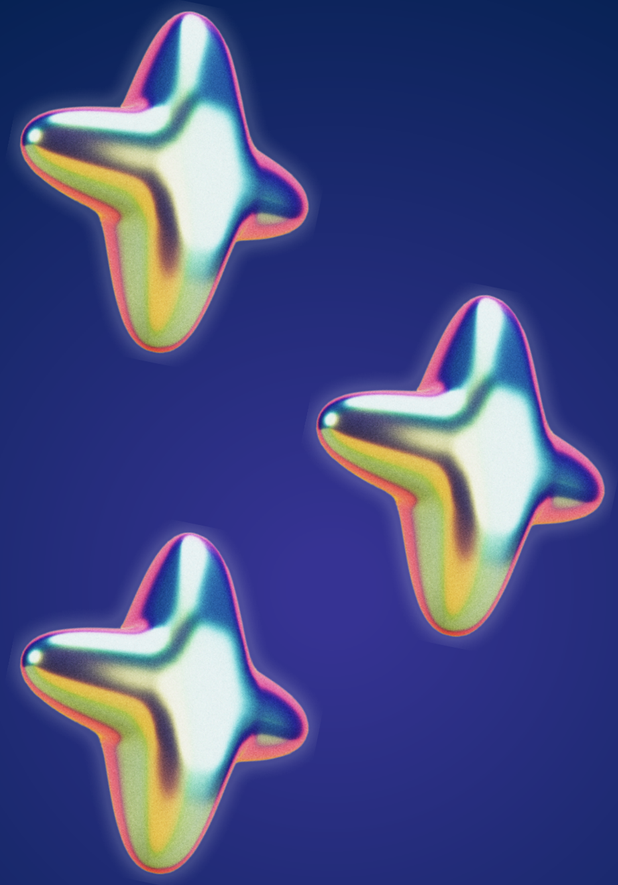
UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS
Acreditación Institucional de Alta Calidad

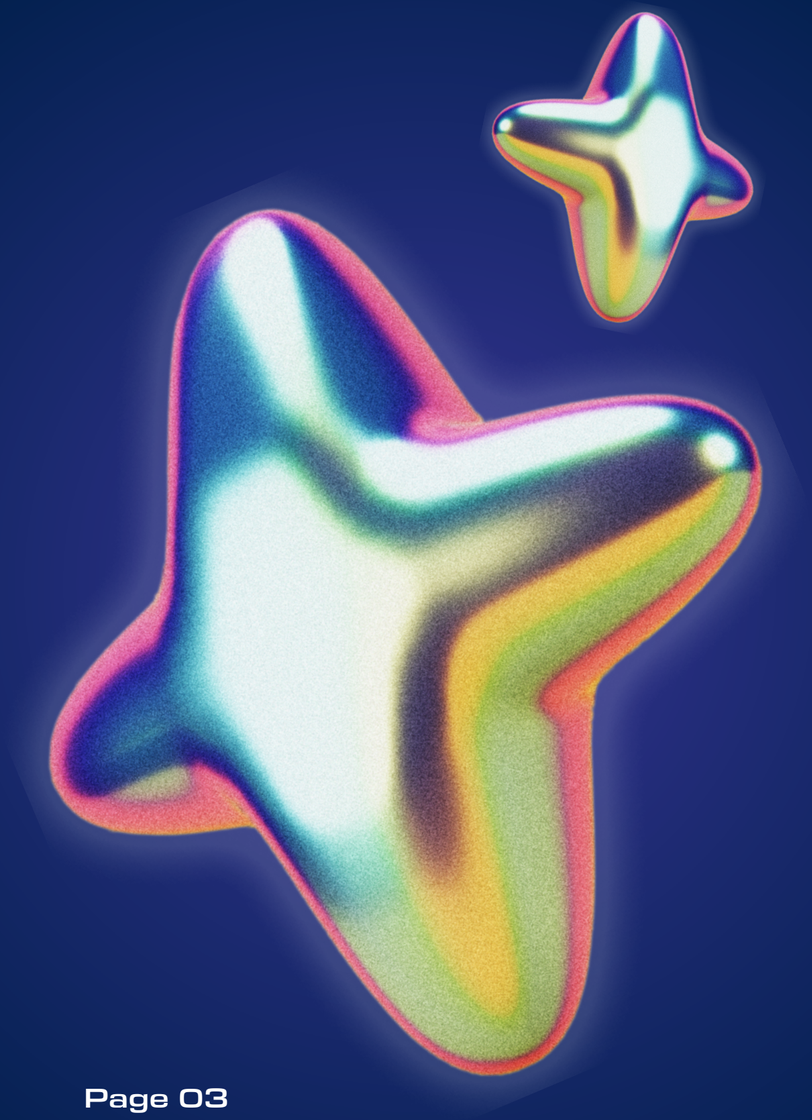
Google Drive Clone

Design of a Secure and Scalable Platform

Cristian David Casas Diaz — 20192020091
Dylan Alejandro Solarte Rincón — 20201020088

Table of Contents





Introduction

This project replicates core functionalities of large-scale cloud platforms in an educational setting.
It focuses on understanding the architectural and data management challenges behind modular, scalable systems.
Beyond storage, the system integrates analytics for governance, usage visibility, and decision support.
Developed under the Databases II course, the project emphasizes real-world modeling, ingestion, and query performance.

Problem Statement

Modern cloud storage platforms are efficient but often lack transparency and native analytical capabilities.

Our goal is to replicate essential features—storage, access control, and analytics—in a modular system tailored for learning and experimentation.

Data governance, user-level insight, and flexibility in modeling different data types (relational, document, graph) are key drivers.

Objectives

General Objective

Design and implement a scalable, modular cloud storage platform integrating hybrid persistence and analytics.

Specific Objectives:

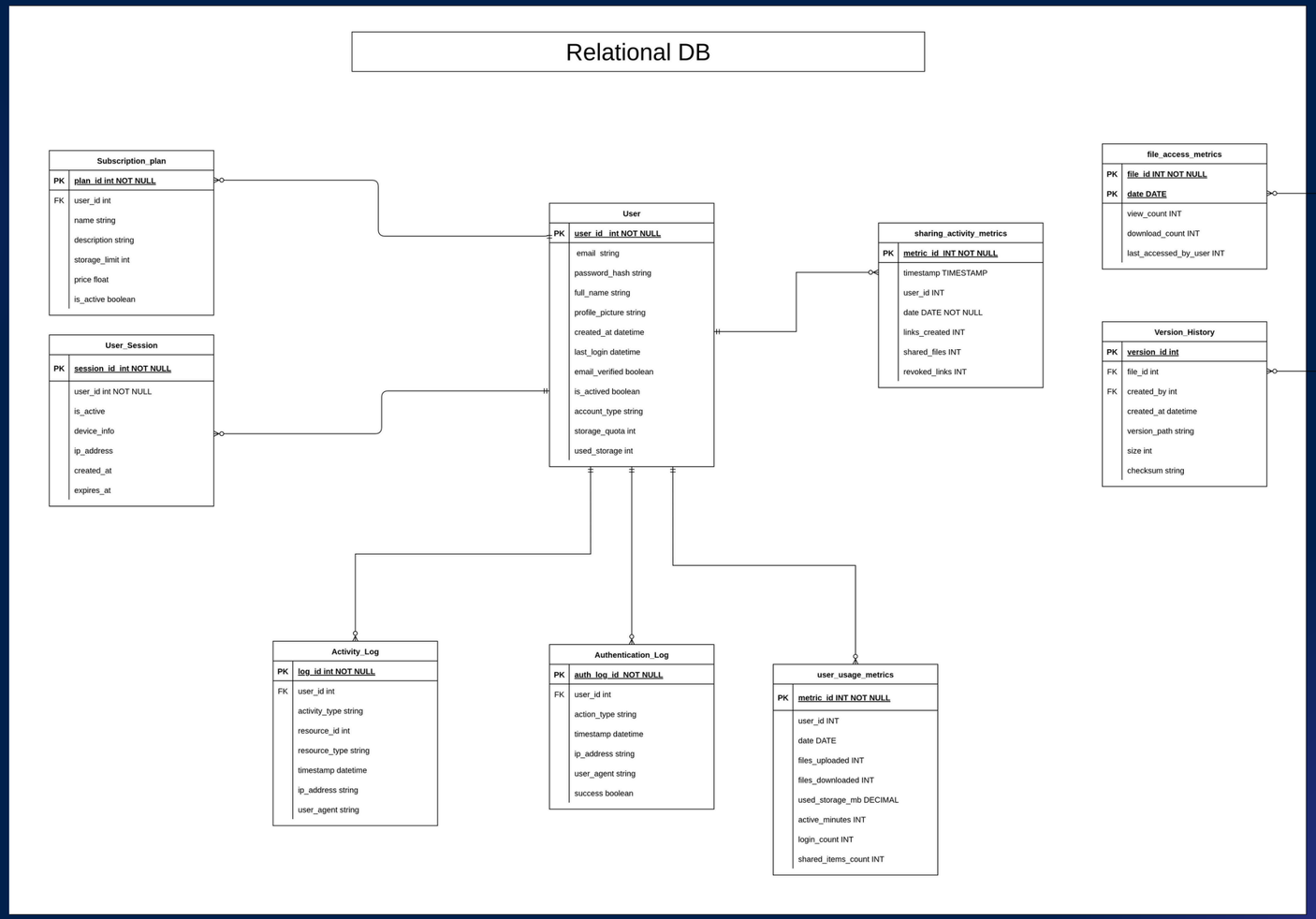
- Apply a hybrid data modeling approach based on data nature.
- Deploy and populate storage modules with synthetic but realistic data.
- Validate data ingestion and query performance across different persistence layers.
- Simulate administrative dashboards with aggregated analytics.

System Architecture

- We adopted a Hexagonal Architecture (Ports & Adapters) to promote modularity and loose coupling.
- Each domain—User, File, Sharing, Permissions, Logging, Analytics—acts as a service with clear boundaries.
- Storage adapters connect to SQL, NoSQL, and graph-based engines depending on the domain's data structure.

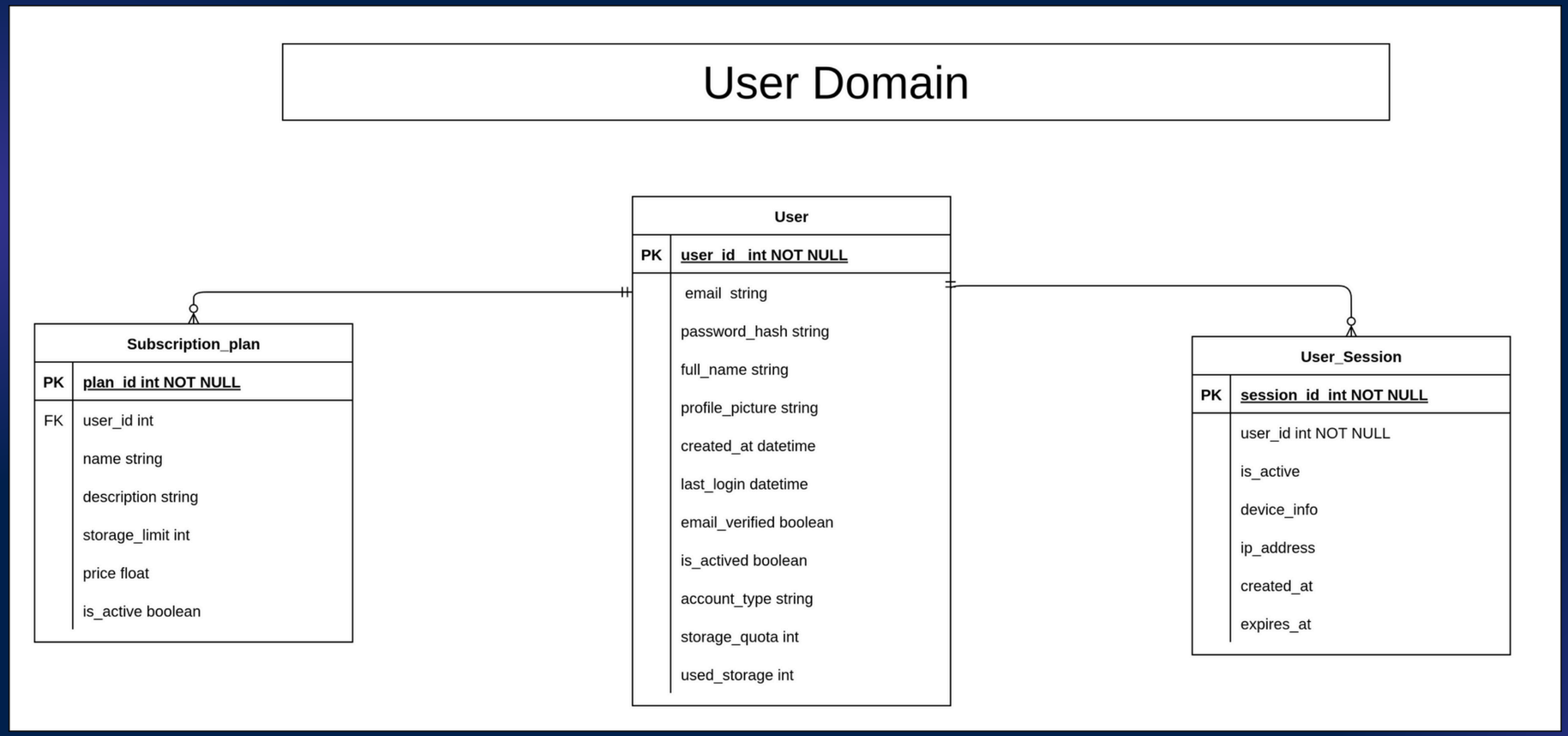
Database Design

- SQL: Used for structured data such as users, sessions, logs, and performance metrics.
- NoSQL: Applied to flexible data like files, folders, and tags.
- Graph-oriented storage (planned): Considered for complex relationships such as permission hierarchies.



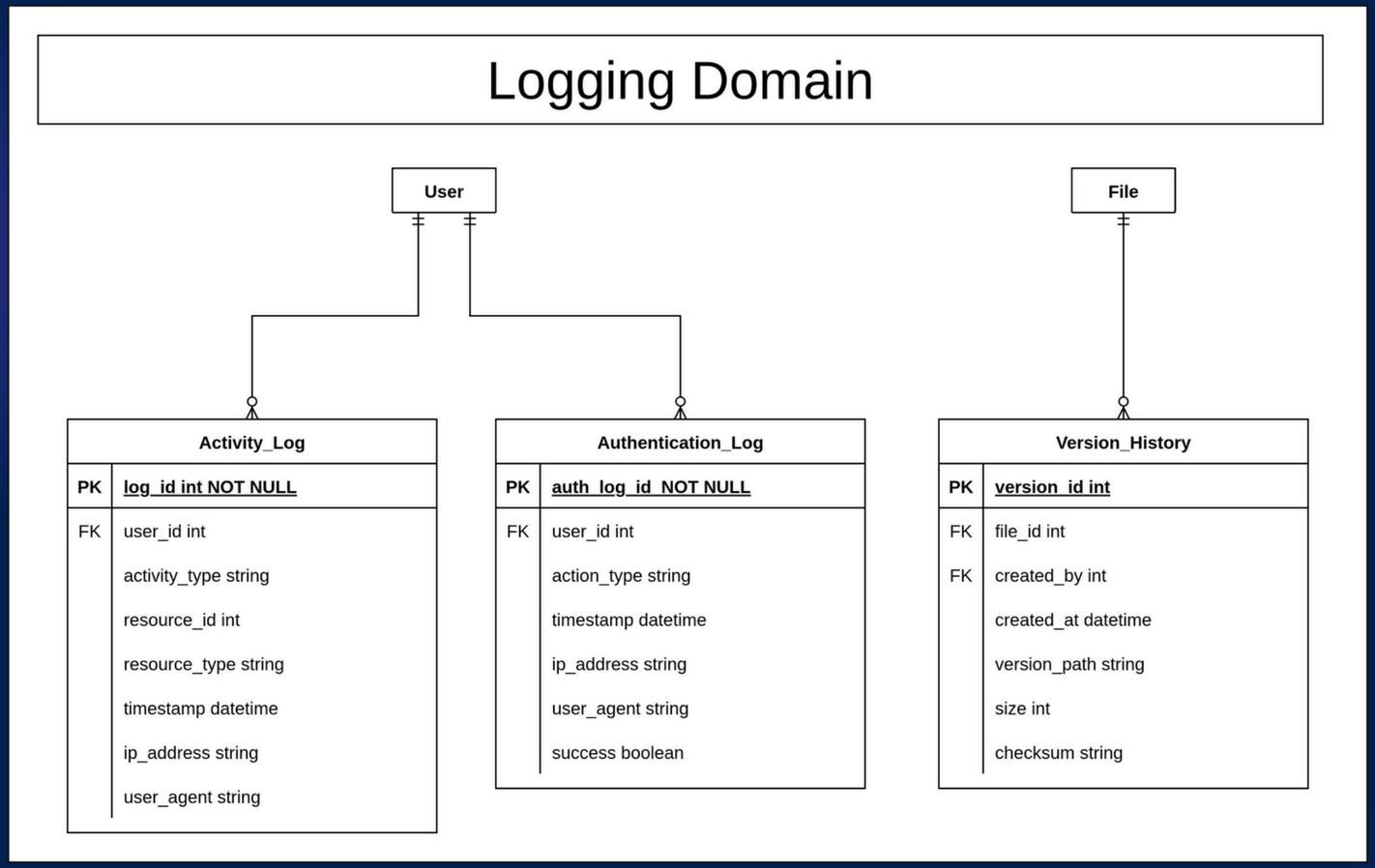
User Domain

- SQL



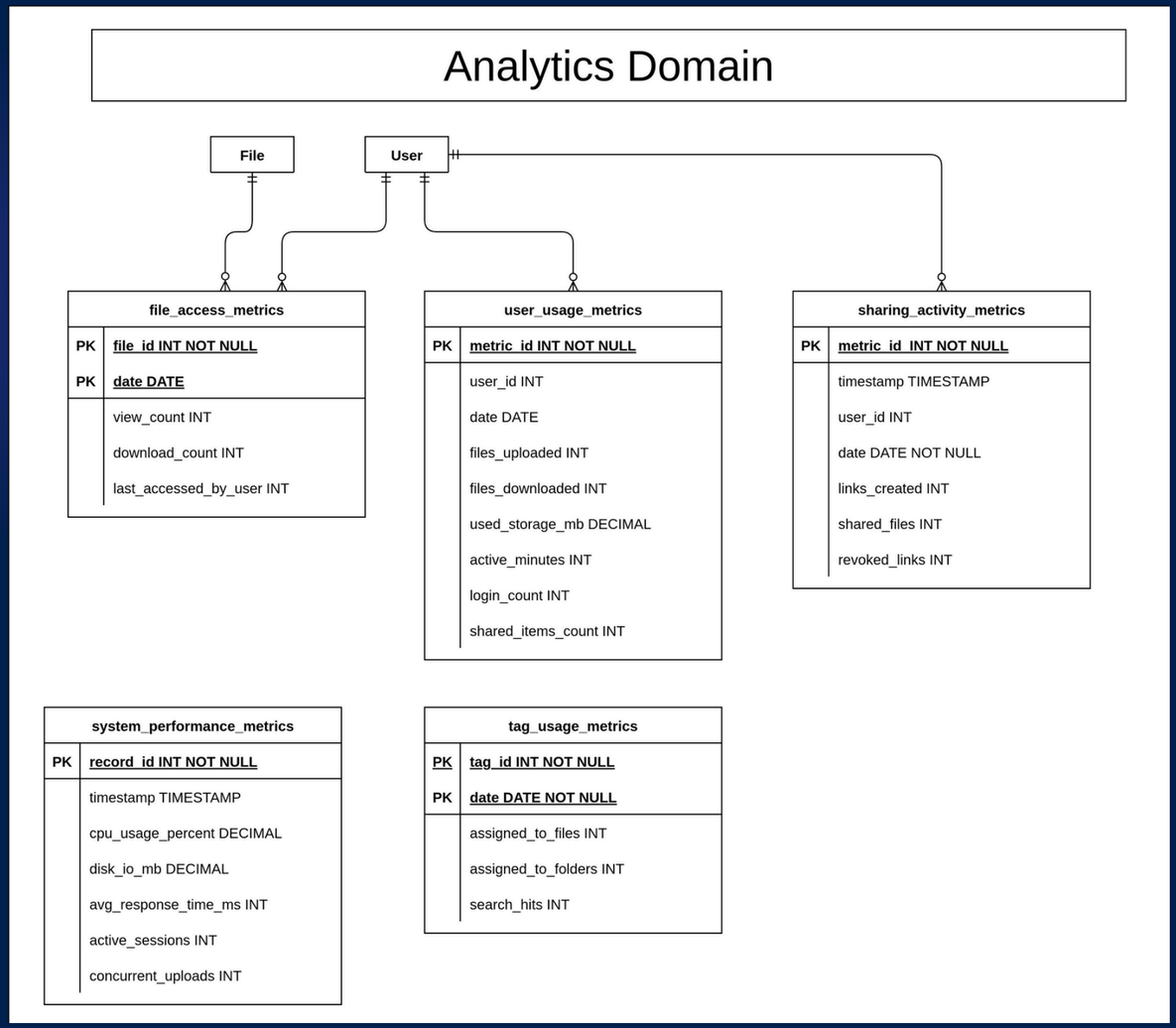
Logging Domain

- SQL



Analytics Domain

- SQL



File Management Domain

- No SQL

files

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
4.86 MB	20 K	560.00 B	1	258.05 kB

folders

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
507.90 kB	5 K	310.00 B	1	81.92 kB

tags

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
20.48 kB	101	86.00 B	1	36.86 kB

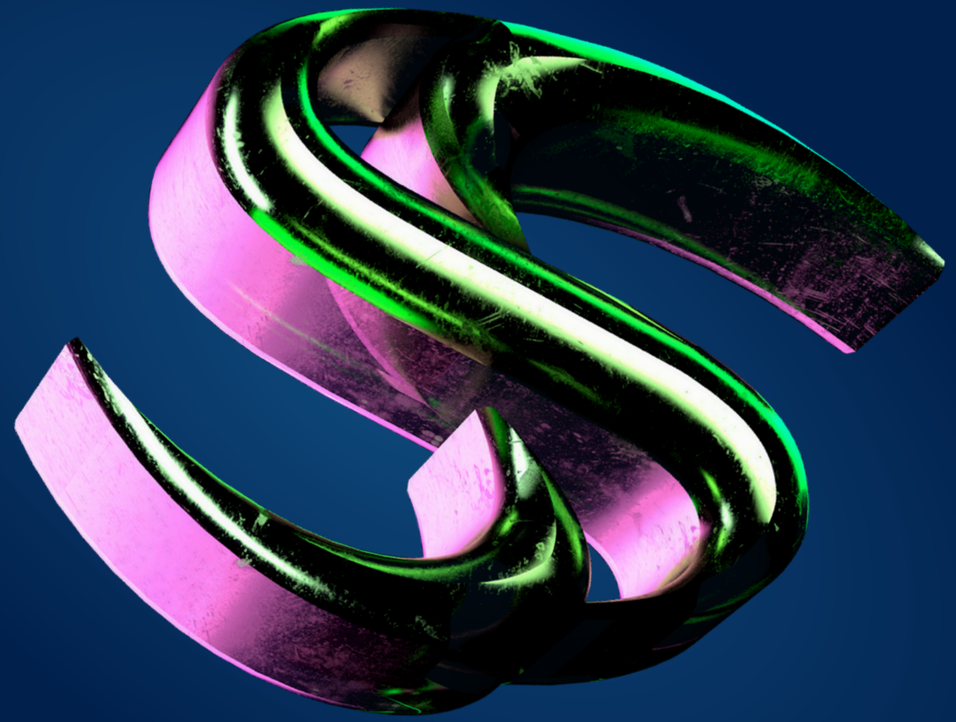
Data Ingestion & Testing

Synthetic data was generated using automated scripts.

Dataset totals:

- 5,000 users
- 20,000 files
- 8,000 file versions
- 30,000+ activity logs
- 80,000+ usage and system metrics

SQL and NoSQL engines were populated and tested under realistic workloads.



Query Performance Results

We evaluated over 20 queries spanning different domains:

- Users: Plan distribution, retention analysis
- Logs: Suspicious login patterns, active users
- Analytics: Storage usage, tag activity, system load
- Files: Tag popularity, folder density, risky file exposure

Query ID	Description	Time (ms)
Relational Queries (PostgreSQL)		
User Q1	Plan distribution & revenue	2.064
User Q2	Premium users >80% quota	2.175
User Q3	Top 10 active sessions	0.657
User Q4	Inactive paid users >90d	3.858
User Q5	Free users >80% quota	2.873
Log Q1	Activity by type/resource	4.781
Log Q2	Failed logins/day/user	2.925
Log Q3	Last successful login	9.218
Log Q4	Files with most versions	13.155
Log Q5	Daily activity summary	4.701
Analytics Q1	Top accessed files	7.662
Analytics Q2	Most active users	4.651
Analytics Q3	Weekly performance trend	0.278
Analytics Q4	Revoked vs created links	5.730
Analytics Q5	Top tags (search & assign)	2.099
Non-Relational Queries (MongoDB)		
NoSQL Q1	Top 10 most downloaded files (last 30 days)	20
NoSQL Q2	Storage usage per user (active files only)	39
NoSQL Q3	Folders with highest file count (top 15)	30
NoSQL Q4	Tag popularity across files and folders	39
NoSQL Q5	Public/confidential file risk detection	16

Conclusions

- The project successfully delivered a modular foundation for cloud-based storage and analytics.
- Synthetic data and custom ingestion enabled realistic scenarios and stress tests.
- Query performance was consistent across storage types, validating architectural choices.
- The system offers a clear path to future extensions like real dashboards and advanced access models.

Thank You
Thank You
Thank You

References

[1] Amazon Web Services. (n.d.). What is data architecture? Retrieved May 13, 2025, from https://aws.amazon.com/es/what-is/data-architecture/

[2] Business model canvas examples. (n.d.). Corporate Finance Institute. Retrieved March 17, 2025, from https://corporatefinanceinstitute.com/resources/management/business-model-canvas-examples/

[3] IBM. (n.d.). What is data architecture? Retrieved May 13, 2025, from https://www.ibm.com/es-es/topics/data-architecture

[4] UNIR. (n.d.). Entity-relationship model. UNIR Revista. Retrieved May 13, 2025, from https://www.unir.net/revista/ingenieria/modelo-entidad-relacion/

[5] Workspace, G. (n.d.). Google Drive: Share files online with secure cloud storage. Google Workspace. Retrieved March 17, 2025, from https://workspace.google.com/intl/es-419/products/drive/

Appendix

- Workshop 1
- Workshop 2
- Workshop 3
- Workshop 4
- GitHub Proyecto Bases II