Refined Final Project Definition and Business Model - Google Drive Clone

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

DATABASES II

Students:
Cristian David Casas Diaz - 20192020091
Dylan Alejandro Solarte Rincón - 20201020088
Professor: Carlos Andrés Sierra Virguez

User Stories

Title: Secure File Upload	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want to securely and quickly upload files to the cloud to ensure accessibility from any device.		
Acceptance Criteria: Given I have an internet connection, when I upload a file, then the file is securely stored in the cloud and accessible from my devices.		

Title: Folder Organization	**Priority:** High	**Estimate:** 3 days
User Story: As a user, I want to create and organize folders to manage my files efficiently.		
Acceptance Criteria: Given I am logged in, when I create a new folder, then I can move files into it and manage my files efficiently.		

Title: File Sharing	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want to share files and folders with others to enable seamless collaboration.		
Acceptance Criteria: Given I have selected a file or folder, when I choose to share it, then I can provide access to others for collaboration.		

Title: Access Permissions	**Priority:** High	**Estimate:** 4 days
User Story: As a user, I want to manage detailed access permissions (view, edit, download) to maintain control over my shared files.		
Acceptance Criteria: Given I am sharing a file or folder, when I set the permissions, then I can specify who can view, edit, or download the file.		

Title: File Synchronization	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want to synchronize my files across devices to always have the latest version available.		
Acceptance Criteria: Given I have multiple devices, when I update a file on one device, then the latest version is available on all other devices.		

Title: Quick Search	**Priority:** High	**Estimate:** 3 days
User Story: As a user, I want to perform quick searches to find files and folders easily.		
Acceptance Criteria: Given I have many files and folders, when I use the search function, then I can quickly find the files or folders I need.		

Title: File Preview	**Priority:** Medium	**Estimate:** 3 days
User Story: As a user, I want to preview documents, images, and videos without downloading them to save time.		
Acceptance Criteria: Given I have a file, when I select to preview it, then I can view the content without downloading it.		

Title: Mobile Access	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want to access my files from mobile devices for greater flexibility.		
Acceptance Criteria: Given I have a mobile device, when I log in, then I can access and manage my files.		

Title: Drag-and-Drop Upload	**Priority:** Medium	**Estimate:** 2 days

User Story: As a user, I want a drag-and-drop upload feature for a more intuitive file management experience.
Acceptance Criteria: Given I want to upload files, when I drag and drop them, then the files are uploaded to the system.

Title: Automatic Backup	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want the system to automatically back up my data to prevent data loss.		
Acceptance Criteria: Given my files are in the system, when the system performs a backup, then my data is backed up to prevent loss.		

Title: Recover Deleted Files	**Priority:** Medium	**Estimate:** 3 days
User Story: As a user, I want to recover deleted files within a certain period to avoid accidental loss.		
Acceptance Criteria: Given I have deleted a file, when I go to the recycle bin, then I can recover the file within the specified period.		

Title:	**Priority:**	**Estimate:**

Activity Log	Low	5 days
User Story: As a user, I want an activity log panel showing who accessed my files and when.		
Acceptance Criteria: Given I want to monitor access, when I view the activity log, then I can see who accessed my files and when.		

Title: User-Friendly Interface	**Priority:** High	**Estimate:** 5 days
User Story: As a user, I want a user-friendly interface that simplifies navigation and usability.		
Acceptance Criteria: Given I am using the application, when I navigate, then the interface is intuitive and easy to use.		

Title: Storage Monitoring	**Priority:** Medium	**Estimate:** 3 days
User Story: As a user, I want to monitor my storage space and receive alerts when approaching my limit.		
Acceptance Criteria: Given I have a storage quota, when I use the system, then I can see my storage usage and receive alerts when close to the limit.		

Title: Access Request Notifications	**Priority:** Low	**Estimate:** 3 days

User Story: As a user, I want to receive notifications for access requests to manage them efficiently.
Acceptance Criteria: Given someone requests access to my files, when a request is made, then I receive a notification.

Title: Shareable Links	**Priority:** High	**Estimate:** 4 days
User Story: As a user, I want to generate shareable links for quick file sharing without individual invitations.		
Acceptance Criteria: Given I have a file, when I generate a shareable link, then others can access the file using the link.		

Title: Favorite Files and Folders	**Priority:** Medium	**Estimate:** 3 days
User Story: As a user, I want to mark files or folders as favorites for quick access.		
Acceptance Criteria: Given I have files and folders, when I mark them as favorites, then I can quickly access them.		

Title: Multiple Display Options	**Priority:** Low	**Estimate:** 3 days

User Story: As a user, I want multiple display options (list/grid view) to organize my files better.
Acceptance Criteria: Given I am viewing my files, when I select a display option, then I can view my files in either a list or grid view.

Title: Rename and Move Files	**Priority:** High	**Estimate:** 3 days
User Story: As a user, I want to rename and move files or folders to keep my workspace tidy.		
Acceptance Criteria: Given I have files and folders, when I rename or move them, then the changes are saved.		

Title: Monitor Storage Usage	**Priority:** High	**Estimate:** 5 days
User Story: As a system administrator, I want to monitor storage usage and performance metrics to optimize system capacity.		
Acceptance Criteria: Given I am a system administrator, when I view the dashboard, then I can monitor storage usage and performance metrics.		

Title: Two-Factor Authentication	**Priority:** High	**Estimate:** 5 days

User Story: As a user, I want to enable two-factor authentication to enhance account security.
Acceptance Criteria: Given I want to enhance security, when I enable two-factor authentication, then my account has an extra layer of security.

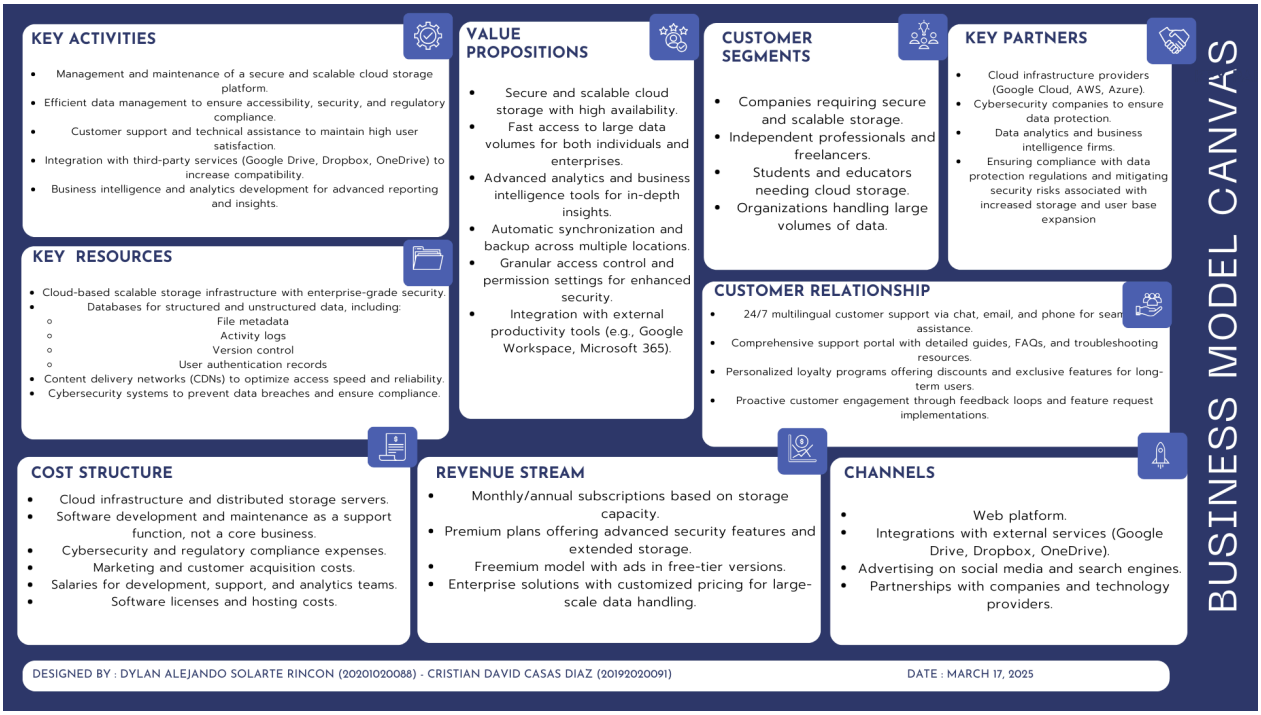
Title: Large File Support	**Priority:** Medium	**Estimate:** 5 days
User Story: As a premium user, I want to upload and manage large files without interruptions.		
Acceptance Criteria: Given I am a premium user, when I upload large files, then the upload is not interrupted.		

Title: Analytics Dashboard	**Priority:** High	**Estimate:** 5 days
User Story: As an administrator user, I want to be able to visualize a dashboard with analytics and statistics.		
Acceptance Criteria: Given I am an administrator user, when I view the dashboard, then I can see analytics and statistics.		

Business Model

In the next link you can observe the diagram with a better resolution:

[Link Business Model Canva]



Key Partners

- **New Addition:** "Ensuring compliance with data protection regulations and mitigating security risks associated with increased storage and user base expansion."

Value Propositions

- **Modification:** "Secure and scalable storage with high availability" → "Secure and scalable cloud storage with high availability."
- **New Additions:**

  "Fast access to large data volumes for both individuals and enterprises."

  "Advanced analytics and business intelligence tools for in-depth insights."

  "Granular access control and permission settings for enhanced security."

"Integration with external productivity tools (e.g., Google Workspace, Microsoft 365)."

Cost Structure

- **Modification:** "Software development and maintenance." → "Software development and maintenance as a support function, not a core business."
- **New Addition:** "Salaries for development, support, and analytics teams."
- **Clarification:** "Software licenses and hosting." → "Software licenses and hosting costs."

Revenue Streams

- **Modification:** "Premium plans with advanced features." → "Premium plans offering advanced security features and extended storage."
- **New Addition:** "Enterprise solutions with customized pricing for large-scale data handling."

Key Activities

- **New Additions:**

  "Management and maintenance of a secure and scalable cloud storage platform."

  "Efficient data management to ensure accessibility, security, and regulatory compliance."

  "Integration with third-party services (Google Drive, Dropbox, OneDrive) to increase compatibility."

  "Business intelligence and analytics development for advanced reporting and insights."

Customer Relationship

- **Modification:** "24/7 customer support via chat, email, and calls." → "24/7 multilingual customer support via chat, email, and phone for seamless assistance."
- **New Additions:**

  "Comprehensive support portal with detailed guides, FAQs, and troubleshooting resources."

  "Personalized loyalty programs offering discounts and exclusive features for long-term users."

"Proactive customer engagement through feedback loops and feature request implementations."

Key Resources

New Additions:

"Cloud-based scalable storage infrastructure with enterprise-grade security."

- "Databases for structured and unstructured data, including:"
- "File metadata"
- "Activity logs"
- "Version control"
- "User authentication records"
- "Cybersecurity systems to prevent data breaches and ensure compliance."

Data Sources and Structures

Key Data Sources

To support the business model and platform functionalities, the following data sources are required:

- **User Data:** Account information, authentication logs, subscription status, and activity history.
- **File Data:** Uploaded documents, metadata (size, type, owner, timestamps), and version history.
- **Transaction Logs:** File sharing records, access permissions, modifications, and user activity logs.
- **Analytics Data:** User behavior insights, storage consumption, most accessed files, and system performance metrics.
- **Security & Compliance Data:** Login attempts, failed access attempts, audit trails, and encrypted backups.

Main Data Structures

To efficiently store and manage data, the following structures inspired by Google Drive's architecture can be used. These are not rigid requirements but rather suggested models for scalability and efficiency:

- **Blob Storage System:** Utilized for scalable file storage, where files are stored as immutable objects across distributed cloud storage nodes.

- **Metadata Indexing System:** Keeps track of file metadata such as file name, size, type, owner, timestamps, and sharing permissions, using a NoSQL database for rapid querying.
- **Access Control Lists (ACLs):** Define user permissions and roles for files and folders, ensuring proper security enforcement through a hierarchical access model.
- **Distributed Logging System:** Captures all user actions such as file uploads, downloads, sharing events, and modifications, leveraging event-driven architecture.
- **Caching:** Speeds up file retrieval and reduces latency by caching frequently accessed files.

Expected Data Volume and Transactions

Estimated Data Growth

Based on the user stories and business model, the estimated data volume and transactions are:

1. User Base and Growth

- **Number of Users:** 1,000 concurrent users (basic), scalable to 5,000 concurrent users
- **User Growth Rate:** Capability to add 1,000 concurrent users per server node

2. Transaction Volumes

- **File Uploads:**
    - Upload Speed: Minimum 2 MB/s on 50 Mbps download / 10 Mbps upload connections
    - Concurrent Uploads: Support for 100 concurrent uploads per server at minimum 1 MB/s per user
- **File Operations:**
    - Response Time: < 500 ms for navigation operations
    - Processing Time: < 2 seconds for complex operations (folders with 1000+ files)

Potential Problems and Considerations

Data Handling & Storage Challenges

- **Storage Optimization:** Efficient data compression and deduplication techniques to reduce space consumption.
- **Data Retrieval Efficiency:** Fast indexing and caching mechanisms for high-speed access.
- **Redundancy & Backup:** Implementing distributed storage solutions to prevent data loss.

Performance, Security, and Scalability Issues

- **Performance:**
    - Load balancing strategies to distribute user requests efficiently.
- **Security:**
    - End-to-end encryption for all stored and transmitted files.
    - Role-based access control to prevent unauthorized data access.
- **Scalability:**
    - Implementing auto-scaling cloud infrastructure to handle traffic spikes.
    - Utilizing distributed databases to manage high transaction volumes.

Functional and non-functional requirements.

Functional Requirements

1. User Management

- **RF1.1:** The system must allow user registration via email and password.
- **RF1.2:** The system must implement an email verification process to activate new accounts.
- **RF1.3:** The system must allow registered users to log in using validated credentials.
- **RF1.4:** The system must allow password recovery via email.
- **RF1.5:** The system must allow users to update their personal information.
- **RF1.6:** The system must implement different access levels (administrator, standard user (Individual and Corporate), premium user).
- **RF1.7:** The system must allow users to log out from any page of the application.

2. File Management

- **RF2.1:** The system must allow users to upload files in various formats (documents, images, videos, audio, etc.).
- **RF2.2:** The system must allow downloading individual files and multiple files (in compressed format).
- **RF2.3:** The system must allow users to preview common files (images, PDFs) without downloading them.
- **RF2.4:** The system must allow creating, renaming, moving, and deleting files.
- **RF2.5:** The system must implement a recycle bin where deleted files remain temporarily before permanent deletion.
- **RF2.6:** The system must allow restoring files from the recycle bin.
- **RF2.7:** The system must allow emptying the recycle bin manually and automatically (after a defined period).

3. Organization and Structure

- **RF3.1:** The system must allow creating, renaming, moving, and deleting folders.
- **RF3.2:** The system must allow organizing files in hierarchical folder structures.
- **RF3.3:** The system must display used and available storage space.
- **RF3.4:** The system must implement a tagging system to categorize files.
- **RF3.5:** The system must allow marking files and folders as favorites for quick access.
- **RF3.6:** The system must display a recent activity view with the latest accessed or modified files.

4. Content Search

- **RF4.1:** The system must allow searching for files and folders by name, type, date, size.
- **RF4.2:** The system must provide advanced filters to refine searches.
- **RF4.3:** The system must allow sorting search results by different criteria (name, date, size, etc.).

5. Sharing and Collaboration

- **RF5.1:** The system must allow sharing files and folders via links.
- **RF5.2:** The system must allow configuring permissions for shared links (read-only, download).
- **RF5.3:** The system must allow setting expiration dates for shared links.
- **RF5.4:** The system must allow sharing files and folders with other registered users.
- **RF5.5:** The system must allow defining different permission levels for users with whom content is shared (owner, reader).
- **RF5.6:** The system must notify users when a resource is shared with them.
- **RF5.7:** The system must allow revoking access to shared files.
- **RF5.8:** The system must allow adding passwords to shared links for increased security.

6. Web Access and Availability

- **RF6.1:** The system must allow access to files from any compatible web browser.
- **RF6.2:** The system must maintain a record of upload dates and last download dates for each file.
- **RF6.3:** The system must automatically update the interface without needing to reload the page when changes are made.

7. Plans and Quotas

- **RF7.1:** The system must implement different subscription plans with specific storage limits.
- **RF7.2:** The system must allow upgrading between plans through a payment system.
- **RF7.3:** The system must notify users when they approach their storage quota limit.
- **RF7.4:** The system must limit certain functionalities when the assigned quota is exceeded.
- **RF7.5:** The system must provide storage usage statistics.

8. File Security

- **RF8.1:** The system must encrypt files during transmission and storage.
- **RF8.2:** The system must scan uploaded files for viruses and malware.

- **RF8.3:** The system must log all operations performed on files (auditing).
- **RF8.4:** The system must allow implementing two-factor authentication for increased security.
- **RF8.5:** The system must implement protection against unauthorized mass downloads.

9. Integration with External Services

- **RF9.1:** The system must allow viewing files through integrated web viewers (PDF, images).
- **RF9.2:** The system must allow integration with social networks to share links.
- **RF9.3:** The system must offer APIs for third parties to develop web integrations.
- **RF9.4:** The system must allow importing files from other storage platforms via web upload.

10. Mobile Web Experience

- **RF10.1:** The system must offer a web interface optimized for mobile devices (responsive web design).
- **RF10.2:** The system must allow viewing compatible files from mobile browsers.
- **RF10.3:** The system must allow uploading files from the device's gallery or camera through the browser.
- **RF10.4:** The system must allow downloading files for local storage from the mobile browser.
- **RF10.5:** The system must implement functionalities adapted to touch screens to improve the experience in mobile browsers.

11. Data Analysis and Big Data

- **RF11.1:** The system must provide a dashboard with statistics and visualizations about platform usage.
- **RF11.2:** The dashboard must include metrics for storage, user activity, usage patterns, and temporal trends.
- **RF11.3:** The system must implement distributed processing technologies (such as Hadoop or Spark) to analyze file metadata and activity logs without affecting performance.
- **RF11.4:** The system must allow batch processing during low-activity hours for complex analyses.
- **RF11.5:** The system must allow generating customized reports on storage usage, sharing patterns, and user activity.
- **RF11.6:** The system must allow exporting reports in standard formats (CSV, PDF, Excel) for external analysis.
- **RF11.7:** The system must implement predictive models to anticipate future storage needs based on historical trends.
- **RF11.8:** The system must anticipate usage peaks to proactively optimize infrastructure resources.
- **RF11.9:** The system must implement indexing and analysis capabilities for common file types.

Non-Functional Requirements

1. Performance

- **RNF1.1:** The system must upload files at a minimum speed of 2 MB/s on connections with 50 Mbps download and 10 Mbps upload bandwidth, and latency less than 150 ms.
- **RNF1.2:** The system must respond to user actions in less than 500 ms for navigation operations and less than 2 seconds for processing operations (listing folders with more than 1000 files, complex searches).
- **RNF1.3:** The system must allow file storage of up to 10 GB in the basic plan and up to 100 GB in premium plans.
- **RNF1.4:** Initial loading time of the user interface (HTML, CSS, essential JavaScript) must not exceed 3 seconds on connections with 25 Mbps download bandwidth and 50 ms latency.
- **RNF1.5:** Loading time for file previews must not exceed 5 seconds for files of 20 MB or less (high-resolution images, PDFs of 100 pages or less).
- **RNF1.6:** The system must support at least 100 concurrent uploads per server, maintaining an upload performance of at least 1 MB/s per user.

2. Scalability

- **RNF2.1:** The system must support up to 1,000 concurrent users, with the capacity to scale to 5,000 users through additional resource provisioning.
- **RNF2.2:** The architecture must be horizontally scalable, allowing additional servers to be added to handle increments of 1,000 concurrent users per server node.
- **RNF2.3:** The system must implement load balancing with a maximum of 80% CPU usage per server before activating additional instances.
- **RNF2.4:** The architecture must allow increasing storage capacity in increments of 10 TB without downtime, with a maximum latency of 30 minutes to integrate new storage.
- **RNF2.5:** The system must implement distributed cache for frequently accessed files.
- **RNF2.6:** The system must use tiered storage (hot/warm/cold) with automatic migration based on access frequency, moving files not accessed for more than 30 days to cold storage.

3. Availability

- **RNF3.1:** The system must guarantee 99.9% availability (uptime), allowing a maximum of 8.76 hours of downtime per year.
- **RNF3.2:** The system must implement N+1 redundancy for all critical components (servers, databases, load balancers).
- **RNF3.3:** The system must perform incremental backups every 6 hours and full backups every 24 hours.
- **RNF3.4:** The maximum disaster recovery time (RTO) must be less than 4 hours, with a recovery point objective (RPO) of less than 30 minutes.
- **RNF3.5:** Scheduled updates and maintenance must be implemented through rolling updates with maximum downtime of 15 minutes per component.

4. Security

- **RNF4.1:** The system must use AES-128 encryption for file storage and TLS for data transfer.
- **RNF4.2:** All communications must use HTTPS with TLS/SSL certificates of 1024 bits or higher.
- **RNF4.3:** Passwords must be stored using the bcrypt algorithm with a minimum work factor of 12, and unique salt per user of at least 16 bytes.
- **RNF4.4:** The system must implement protection against common attacks, including:
    - XSS: Implement Content Security Policy (CSP) and escape input/output data
    - SQL Injection: Use parameterized queries and ORM
    - DDoS: Implement rate limiting (maximum 100 requests per minute per IP)
    - Brute Force: Account lockout after 5 failed attempts for 30 minutes
- **RNF4.5:** Login attempts must be limited to 5 failed attempts per account in a 30-minute period, with a progressive blocking period.
- **RNF4.6:** The system must scan files with an anti-malware engine that detects at least 90% of known threats.

5. Usability

- **RNF5.1:** The user interface must allow 90% of new users to complete basic tasks (upload, download, share) without external help on their first use.
- **RNF5.2:** The system must provide notifications and feedback messages for each operation, with a maximum appearance time of 500 ms and adjustable duration (2-10 seconds).
- **RNF5.3:** Interactive tutorials must be available for all main functions, completable in less than 3 minutes each, with the option to skip them.
- **RNF5.4:** Progress indicators for upload/download must update in real-time (every 500 ms), show percentage completed, current speed, and estimated completion time.

6. Compatibility

- **RNF6.1:** The system must function correctly on the latest versions of major browsers and their two previous versions: Chrome (versions 120+), Firefox (versions 115+), Safari (versions 16+), Edge (versions 110+).
- **RNF6.2:** The web interface must function correctly on mobile browsers iOS Safari (version 15+) for devices with iOS 13+, and Chrome/Firefox/Samsung Internet for Android 8.0+ devices.
- **RNF6.3:** The web interface must be fully responsive for screens from 320px to 2560px in width, and optimally adapt its design for at least 5 breakpoints (small mobile, large mobile, tablet, desktop, large screen).
- **RNF6.4:** The system must support direct browser preview for the following file formats, including: PDF, TXT, image formats (JPG, PNG, GIF, SVG, etc.), audio formats (MP3, WAV, FLAC, etc.), video formats (MP4, MKV, AVI, etc.), compressed file formats (ZIP, RAR, 7Z, etc.).

7. Maintainability

- **RNF7.1:** The system must implement detailed logging with at least 4 levels (DEBUG, INFO, WARNING, ERROR), log rotation every 7 days or 1GB, and minimum retention of 90 days.

- **RNF7.2:** The web architecture must follow a modular pattern (e.g., microservices or hexagonal architecture) with clearly defined and documented interfaces, allowing replacement of individual components without affecting more than 20% of the system.
- **RNF7.3:** Test coverage must be greater than 60% for unit tests, 50% for integration tests, and must include end-to-end tests for all critical user flows.
- **RNF7.4:** Continuous integration must run the complete set of automated tests in less than 30 minutes, and continuous deployment must allow production updates in less than 2 hours from code approval.

8. Localization and Internationalization

- **RNF8.1:** The interface must initially support Spanish and English (100% of texts translated), with the capacity to add additional languages through external resource files without modifying the code base.
- **RNF8.2:** The system must use UTF-8 encoding throughout the platform to correctly handle international character sets, including extended Latin alphabets, Cyrillic, Chinese, Japanese, Korean, and Arabic.
- **RNF8.3:** The system must automatically detect the user's time zone based on their browser configuration, with the option of manual selection among the 24 main time zones.
- **RNF8.4:** The system must display dates, times, numbers, and size units according to local conventions for each region (DD/MM/YYYY vs MM/DD/YYYY, decimal separators, etc.), based on at least 10 main regions.

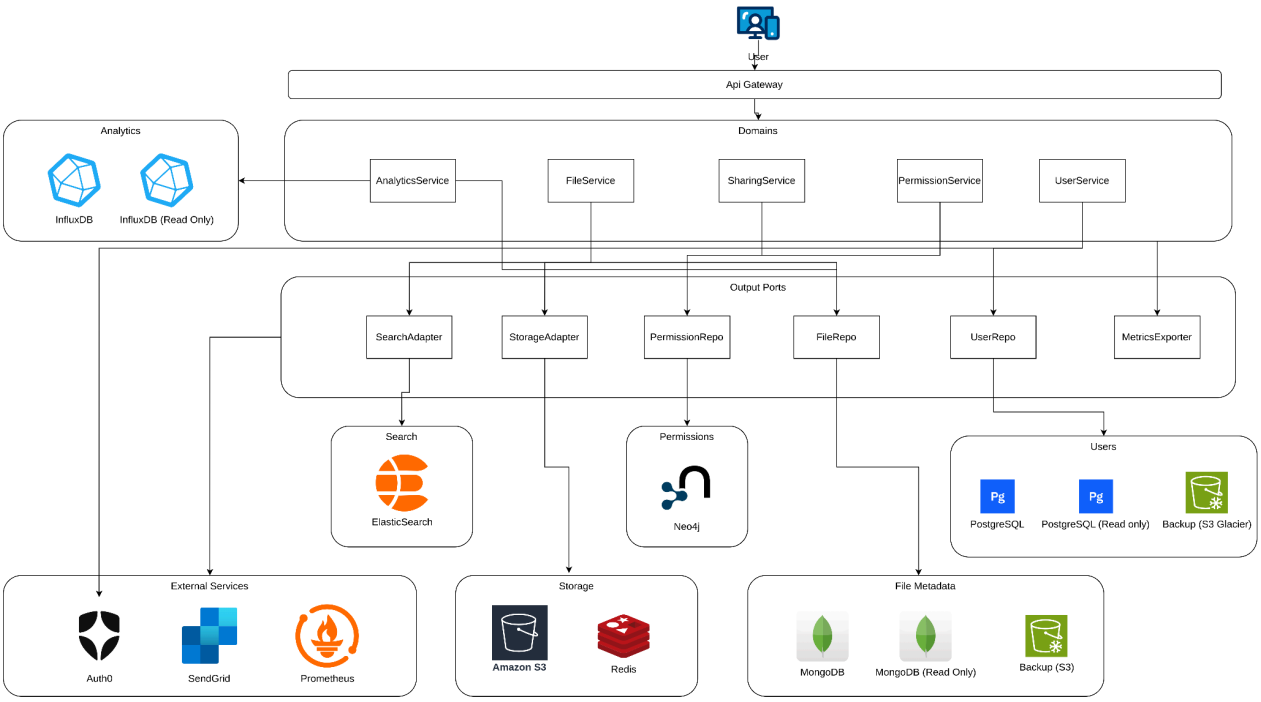
9. Efficiency

- **RNF9.1:** The system must implement gzip or brotli compression for all textual resources (HTML, CSS, JavaScript, JSON), reducing transfer size by at least 70%.
- **RNF9.2:** Lazy loading must be applied to all non-critical resources, including images and interface components that are outside the initial view, reducing initial loading time by at least 40%.
- **RNF9.3:** The system must automatically optimize uploaded images, reducing their size by at least 50% for JPG and PNG without perceptible loss of quality, and convert inefficient formats to WebP when the browser supports it.

Initial Database Architecture

This system architecture represents a scalable and secure cloud-based file management platform inspired by Google Drive. The platform is designed using a hexagonal (ports and adapters) architecture, ensuring modularity, ease of testing, and maintainability.

Link Repository Initial Architecture



1. Entry Point – API Gateway

All incoming requests from the client applications are routed through the **API Gateway**, which acts as a single entry point to the backend services. It handles request routing, authentication, throttling, and load balancing.

2. Domain Layer

This layer contains the core services that encapsulate business logic:

- **AnalyticsService** – Gathers and manages platform usage statistics.
- **FileService** – Manages file uploads, downloads, previews, and metadata.
- **SharingService** – Handles file/folder sharing functionalities.
- **PermissionService** – Manages access control with fine-grained permissions.
- **UserService** – Manages user accounts and profiles.

3. Output Ports (Adapters Layer)

These adapters connect domain logic to external systems or infrastructure:

- **SearchAdapter** → Connects to **Elasticsearch** for full-text and filtered search capabilities.
- **StorageAdapter** → Interfaces with **Amazon S3** and **Redis** for file storage and caching.
- **PermissionRepo** → Communicates with **Neo4j** to manage hierarchical permission models.
- **FileRepo** → Stores metadata in **MongoDB**, including read-only replicas for analytics and recovery.
- **UserRepo** → Manages user information stored in **PostgreSQL**, including read-only replicas and cold storage backups in **S3 Glacier**.
- **MetricsExporter** → Exposes system performance and health metrics for monitoring (e.g., **Prometheus**).

4. Analytics Layer

Uses **InfluxDB** for time-series data and analytics, including a read-only instance for visualization dashboards and historical data review.

5. External Services

- **Auth0** – Provides authentication and two-factor security mechanisms.
- **SendGrid** – Used for transactional emails, such as password recovery and file-sharing notifications.
- **Prometheus** – Collects performance metrics, supporting real-time monitoring and alerting.

6. Storage Systems

- **Amazon S3** – Primary object storage for files.
- **Redis** – In-memory cache for faster access to frequently used files and metadata.

7. File Metadata Management

- **MongoDB** – Stores file metadata and access logs.
- **S3 Backup** – Ensures backup of metadata and logs with long-term durability.

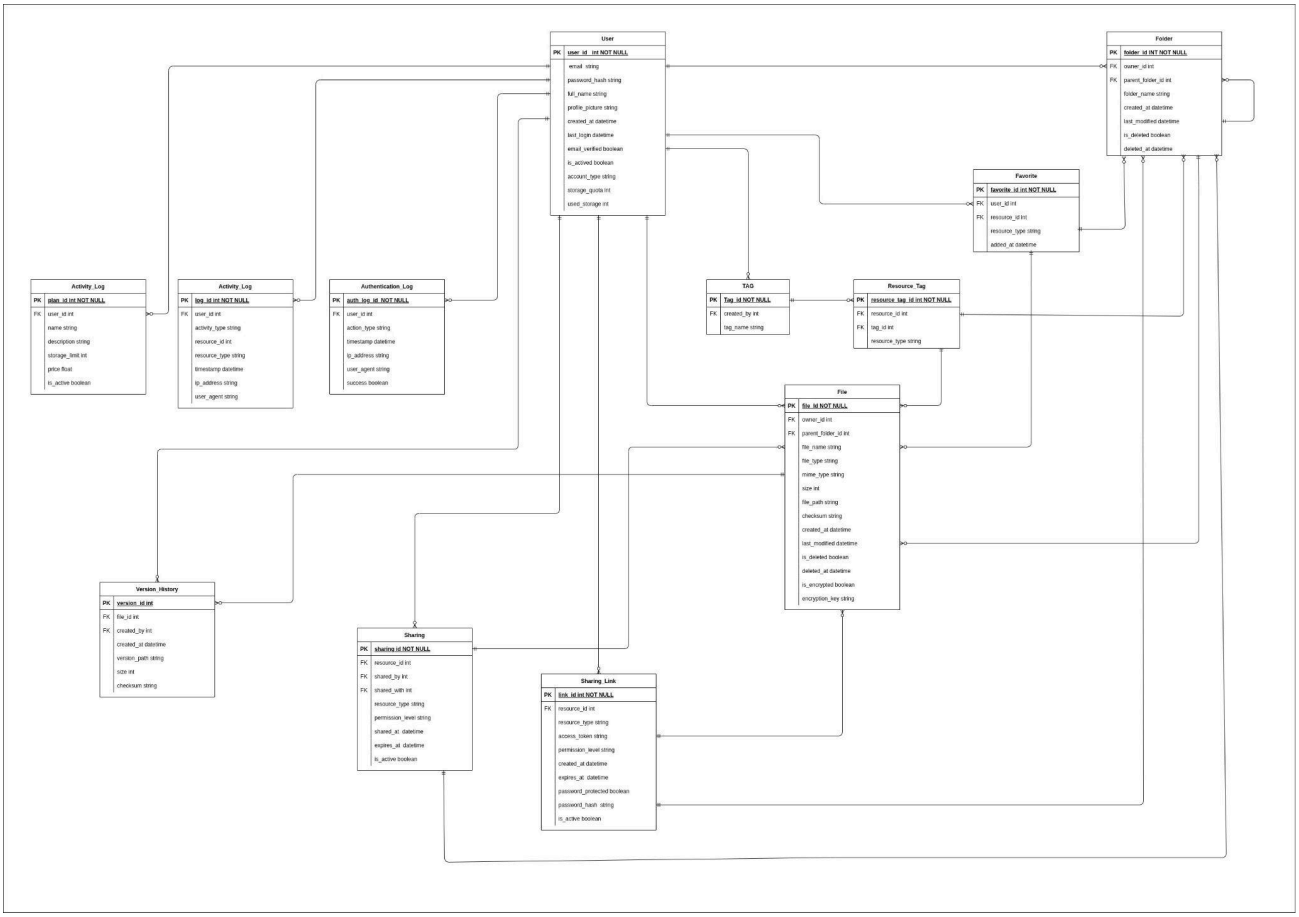
8. User Data Storage

- **PostgreSQL** – Stores user credentials, profiles, and activity logs.
- **S3 Glacier** – Ensures long-term, low-cost backup for user data.

First Version ER Diagram

Below is a detailed explanation of the Entity-Relationship Diagram, including its key entities, relationships, and data flow, designed for our Google Drive clone platform.

[ER Diagram Repository Link]



Main Entities:

- **USER**: Stores user information, including credentials, personal details, storage quotas, and account type (standard, premium, administrator).
- **SUBSCRIPTION_PLAN**: Defines the different available plans along with their storage limits and pricing.

- **FILE**: Represents files uploaded by users, containing metadata such as name, type, size, location, and security settings.
- **FOLDER**: Organizational structure that holds files and other folders, enabling hierarchical organization.
- **SHARING**: Manages access permissions between users for shared files and folders.
- **SHARED_LINK**: Enables access to resources through shareable links, with expiration and password protection options.
- **TAG**: Tagging system used to categorize files and folders.
- **ACTIVITY_LOG**: Records all user actions on the platform, useful for auditing and analytics.
- **FAVORITE**: Marks resources as favorites for quick access.
- **VERSION_HISTORY**: Maintains historical versions of files for recovery and change tracking.

Key Relationships:

- A user can have multiple files, folders, shares, and activity records.
- Folders can contain multiple files and subfolders (recursive relationship).
- Files and folders can be shared with multiple users, each with different permission levels.
- Files can have multiple historical versions.
- Resources (files/folders) can be tagged with multiple labels and marked as favorites.

Data Flow and Storage Considerations

Storage Structure:

- **Layered Storage:**
    - **Hot Storage**: For frequently accessed files.

    - **Cold Storage**: For rarely accessed files, optimizing storage costs.

- **Separation of Metadata and Content:**
    - Relational databases are used to store metadata (table-based information).
    - Object storage systems manage the actual file content.
- **Large File Handling:**
    - Implementation of chunked upload for large files.
    - Efficient storage through deduplication techniques.

Data Flow:

- **Processing and Scalability:**
    - Distributed architecture to support high volumes of data.
    - Caching system for frequently accessed files to improve performance.
- **Data Security:**
    - End-to-end encryption during data transmission.
    - Encryption at rest for stored data.
    - Role-based access control with granular permission settings.
- **Analytics and Monitoring:**
    - Batch processing of logs for usage analytics.
    - Monitoring systems to track platform performance and storage usage.

References

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

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

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

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

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/