

Workshop No. 1 Data Base II

Universidad Distrital Francisco José de Caldas
Computer Engineering
Cuellar B. Paola A.
Ebratt S. Johan D.

Business Canvas Model

The Canva Model is a simple methodology that defines the business strategy and the target audience. Its objective is to provide a clear understanding of the type of business to be created and introduced in the market, who it is aimed at, how it will be sold and how income will be obtained.

Having this clear, we want to develop a social network, UDshare, which allows to connect users with their followers in a safe way, the followers will be other users, public figures and companies. In UDshare it will be possible to make publications with multi-media content and users will be able to interact with them, with comments and reactions in addition users will be able to follow their publications.

KEY PARTNERS	KEY ACTIVITIES	KEY RESOURCE	VALUE PROPOSITIONS	CHANNELS	CUSTOMER SEGMENTS
Advertising Companies Companies that want to advertise their resources, representing a source of income and attracting users interested in their offerings.	User Identity Verification Implementation of secure identity verification using official documents based on the user's country, ensuring full compliance with data privacy regulations.	Human Talent Skilled developers and DevOps professionals proficient in modern technologies, responsible for continuous development, deployment, and maintenance of the platform.	We offer a personalized social media platform that connects users with shared interests through intelligent recommendations and secure identity verification. Our propose fosters meaningful interactions between users and companies and continuously improves the experience using data analytics and business intelligence.	Core Platform The web applications serve as the main interface for users to interact with the platform. Through these channels, users can publish content, engage with others.	General Audience Our platform is designed for a broad audience, targeting anyone over the age of 15 who seeks to connect with others through shared interests, trends, and personalized content.
Other Social Media Platforms Interconnectivity with other platforms helps us attract new users and increase reach.	Interest-Based Interaction Channels Development of official interest spaces where users and companies can engage around shared topics and opportunities.	Security Frameworks Resources and technologies to guarantee secure user authentication, identity verification, and privacy compliance.		Distribution and Growth	Users
Data Analysis Companies Partners with whom we exchange information to improve recommendations and provide valuable insights for their analysis.	Recommendation Engine Management Continuous refinement of algorithms that personalize user feeds based on behavior and preferences.	Data Storage and Processing Systems Data centers and warehouses that ensure efficient handling of big data, supporting ingestion and retrieval at scale.	CUSTOMER RELATIONSHIPS Direct relationships with users to foster trust and long-term engagement.		They are the primary customers who engage daily with the platform, posting content, following others, and consuming media. Their needs revolve around usability, content relevance, privacy, and connection.
Moderation and NLP API Providers External services that help detect hate speech, spam, and inappropriate content automatically.	Data Ingestion & Analytics Constant ingestion of user interactions for real-time analytics and improved recommendations.	Business Intelligence & Analytics Tools Systems and expertise for analyzing user behavior and market trends to refine recommendations and target content.	Immediate and effective response to account-related issues to ensure a safe and smooth user experience.	Support and Communication To ensure a responsive and user-centric experience, we offer immediate support through online communication channels such as in-app chat and email. We also provide topic-based channels where users and companies can interact around shared interests.	Companies Businesses and service providers use the platform to advertise, engage with targeted audiences, and participate in interest-based channels. They seek visibility, analytics, and effective communication with users.
Cloud Infrastructure Providers Enable scalable, high-availability hosting and distributed data storage.		Cloud Infrastructure Scalable and reliable cloud services supporting multi-location access, fast queries, and high availability.	24/7 support availability through chat and email for quick resolution of concerns.		Personas These are socially influential users, public figures, thought leaders, or content creator, who use the platform to amplify their reach and shape public opinion. Their needs include audience analytics, content promotion, and identity verification.
			Feedback and interaction channels that allow users to report issues, suggest improvements, and feel heard.		Members Paying users who subscribe to premium features such as enhanced visibility, analytics, post boosts, or exclusive content. They expect added value and personalization for their investment.
COST STRUCTURE			REVENUE STREAMS		
Data analysis and storage Major investments go toward processing and storing vast amounts of user data to deliver personalized experiences and accurate recommendations			Membership subscriptions Users pay for access to premium features such as exclusive content, performance analytics, and improved visibility on the platform.		
Technology and network infrastructure Costs include cloud services, high-speed connectivity, and computing power to ensure scalability, fast response times, and user interaction.			Advertising Companies and public figures invest in ad placements to reach specific audiences, increasing brand exposure and engagement.		
Server maintenance Ongoing expenses involve maintaining server uptime, applying security patches, system monitoring, and ensuring platform availability 24/7.			Data insights services Businesses pay for anonymized, aggregated analytics that help them understand trends and improve their strategies, respecting privacy standards.		

Figure 1: Business Canvas Model Source: Authors

User Stories

The following user stories describe the functional and non-functional requirements of the system from the end user's perspective. Each story is written in a format that captures a specific need and the value it brings to the user within the social network being developed.

The purpose of documenting these stories is to facilitate a shared understanding among the development team and stakeholders about what is expected from the system. These stories form the foundation of the product backlog and guide the iterative and incremental development of the solution.

Each user story has been assigned a priority (High, Medium, or Low), which reflects its impact on system functionality and the value it delivers. Additionally, an effort estimation is included for each story, expressed in story points. These points do not represent a direct measure of time (e.g., hours or days), but rather a relative measure of complexity, uncertainty, and the amount of work required to implement the story.

The estimation technique used was Scrum Poker or Planning Poker, a collaborative practice in agile methodologies where each development team member independently assigns effort points to a story. The differences in estimations are then discussed until a

group consensus is reached, promoting a shared understanding of the scope and encouraging active participation from all team members.

Below is the set of prioritized and estimated user stories for this phase of the project.

ID	UH-1: User Registration
Priority	High
Estimate	8
User Story	As a user, I want to register on the platform by providing my basic information so that I can access all the features of the social network.
Acceptance Criteria	<ul style="list-style-type: none"> • Given a visitor wants to create an account on the platform • When they submit the registration form with their name, email, password, and date of birth • Then the system must validate: <ul style="list-style-type: none"> – The email has a valid format and is unique – The password meets security requirements (min. 8 characters, including uppercase, numbers, special characters) – All required fields are completed – If all validations pass, create the user account and grant access

ID	UH-2: User Login
Priority	High
Estimate	5
User Story	As a user, I want to log in using my email and password or a third-party account so that I can securely access the social network.
Acceptance Criteria	<ul style="list-style-type: none"> • Successful Login: Given a registered user has valid credentials or linked third-party account. When they log in, then authenticate and generate a JWT token with expiration. • Invalid Credentials: Given incorrect credentials. When they attempt to log in, then show generic error (no email existence hint). • Account Lockout: If failed attempts exceed threshold, temporarily lock account.

ID	UH-3: Secure Authentication Implementation
Priority	High
Estimate	8
User Story	As a user, I want to log in securely using a token-based authentication method so that I can access the platform safely.
Acceptance Criteria	<ul style="list-style-type: none"> • Token includes digital signature and expiration • Third-party login issues secure tokens under same standards • Tokens follow platform-wide security policies

ID	UH-4: View Profile
Priority	High
Estimate	8
User Story	As a user, I want to be able to view my profile so that I can see how my account appears to others.
Acceptance Criteria	<ul style="list-style-type: none"> • Profile shows name, biography, profile picture, and cover photo • Biography shown in full if near character limit • Sensitive data like email shown securely to owner only

ID	UH-5: Profile Editing
Priority	Medium
Estimate	5
User Story	As a user, I want to be able to edit my profile to personalize my account on the social network.
Acceptance Criteria	<ul style="list-style-type: none"> • Update name, biography, picture, and cover photo • Notify user on successful update • Prevent save if biography exceeds character limit • Store history of changes to sensitive data (email/username)

ID	UH-6: Posting
Priority	High
Estimate	8
User Story	As a user, I want to write and post up to 500 characters to share ideas and updates.
Acceptance Criteria	<ul style="list-style-type: none"> • Allow text input up to 500 characters • Prevent further input or show error if limit exceeded • Save and display post in timeline upon publishing

ID	UH-7: Add Multimedia to Post
Priority	Medium
Estimate	8
User Story	As a user, I want to create posts with text and media to share updates with my followers.
Acceptance Criteria	<ul style="list-style-type: none"> • Attach media to posts optionally • Publish post with media to timeline

ID	UH-8: Follow Other Users
Priority	High
Estimate	5
User Story	As a user, I want to follow others to see their posts and stay updated.
Acceptance Criteria	<ul style="list-style-type: none"> • Add users to “Following” list • Show their posts in user’s timeline • Display followers/following count

ID	UH-9: Unfollow Users
Priority	Medium
Estimate	5
User Story	As a user, I want to follow and unfollow users to personalize my content feed.
Acceptance Criteria	<ul style="list-style-type: none"> • Toggle follow/unfollow on user profile • Feed updates accordingly

ID	UH-10: Interact with Posts
Priority	Medium
Estimate	5
User Story	As a user, I want to interact with posts and find them easily later.
Acceptance Criteria	<ul style="list-style-type: none"> • Allow like/favorite interaction • Highlight interaction visually • List interacted posts in user profile

ID	UH-11: Comment on Posts
Priority	High
Estimate	8
User Story	As a user, I want to comment on posts so that I can engage in conversations.
Acceptance Criteria	<ul style="list-style-type: none"> • Add comment below the post • Notify original post author

ID	UH-12: Share Post
Priority	Low
Estimate	3
User Story	As a user, I want to share posts so that they can be seen by my followers.
Acceptance Criteria	<ul style="list-style-type: none"> • Re-publish post to sharer's profile timeline • Visibility according to sharer's privacy setting

ID	UH-13: Edit a Post
Priority	Medium
Estimate	5
User Story	As a user, I want to edit a post after publishing it to correct errors or add information.
Acceptance Criteria	<ul style="list-style-type: none"> • Edit post content in editable text field • Save changes • Show "edited" label on updated posts

ID	UH-14: Configure Post Privacy
Priority	Medium
Estimate	8
User Story	As a user, I want to configure post privacy to control who sees my content.
Acceptance Criteria	<ul style="list-style-type: none"> • Set privacy level (Public, Followers, Private) when posting • Enforce visibility based on selected privacy

ID	UH-15: Reported Post
Priority	Medium
Estimate	5
User Story	As an admin, I want to review reported content to moderate and enforce standards.
Acceptance Criteria	<ul style="list-style-type: none"> • View report details from moderation dashboard • Allow actions like delete, ignore, or ban

ID	UH-16: Reported Users
Priority	Medium
Estimate	5
User Story	As an admin, I want to suspend users who violate rules to ensure platform safety.
Acceptance Criteria	<ul style="list-style-type: none"> • Review user violation history • Suspend account and notify the user

ID	UH-17: Verification Process
Priority	High
Estimate	8
User Story	As an admin, I want to manage verification requests to ensure credibility for public accounts.
Acceptance Criteria	<ul style="list-style-type: none"> • Approve or reject verification request • Reflect decision on user profile

ID	UH-18: Boost Post
Priority	High
Estimate	5
User Story	As a member, I want to boost my posts so they reach a larger audience.
Acceptance Criteria	<ul style="list-style-type: none"> • Post is prioritized in timelines • Post is labeled as “promoted” • Requires active premium subscription

ID	UH-19: Advanced Analytics to Members
Priority	Low
Estimate	5
User Story	As a member, I want access to advanced analytics to evaluate my content's performance.
Acceptance Criteria	<ul style="list-style-type: none"> • Display metrics such as impressions, engagement rate, and demographics • Accessible from profile for premium users

System-Level Analysis and Requirement Justification

The proposed platform is a social networking application intended to support user-generated content, profile management, and personalized feeds, similar to existing platforms such as X (formerly Twitter). Users will be able to interact through posts, comments, likes, and shares. Key technical constraints include the need for scalability to support high user volumes, rapid data delivery for a seamless experience, and robust moderation to ensure community safety.

This system is expected to handle thousands of concurrent users and deliver content within strict performance limits. For example, loading a timeline or executing a search query must occur within 100 milliseconds under typical conditions. This benchmark is informed by user expectations based on industry standards from leading platforms, where high responsiveness directly affects user retention. Additionally, user interactions like posting or following must update system state with minimal latency and ensure consistent delivery across geographically distributed data centers.

Security and privacy are essential due to the collection and storage of sensitive user information. Thus, the system must implement token-based authentication, enforce encryption, and comply with international regulations such as GDPR.

These factors inform the following functional and non-functional requirements.

Functional Requirements Specification

Requirement ID	FR-001: User Registration
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Authentication service
Description	The system must allow a new user to register using an email or an external authentication provider (OAuth).
Specific Data	<ul style="list-style-type: none"> • Username • Email address • Password or external auth token
Priority	High
Status	In development
Comments	The system must validate that the email is not already in use.

Requirement ID	FR-002: Create Post
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Media service
Description	The system must allow users to create and publish messages with text, images, or videos.
Specific Data	<ul style="list-style-type: none"> • Post content (text) • Media files (optional) • Timestamp of publication
Priority	High
Status	In development
Comments	Content filtering and quality checks should be applied before publishing.

Requirement ID	FR-003: Follow/Unfollow Users
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Notification service
Description	The system must allow users to follow or unfollow other users, modifying their personalized feed accordingly.
Specific Data	<ul style="list-style-type: none"> • Follower user ID • Followed user ID • Timestamp of the action
Priority	High
Status	In development
Comments	A notification should be sent if the followed user has this setting enabled.

Requirement ID	FR-004: Like and Repost Content
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Notification system
Description	The system must allow users to like or repost content created by others.
Specific Data	<ul style="list-style-type: none"> • User ID • Post ID • Type of interaction (like or repost) • Timestamp
Priority	High
Status	In development
Comments	Notifications should be generated for the content creator.

Requirement ID	FR-005: Comment on Posts
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Moderation system
Description	The system must allow users to write and publish comments on existing posts.
Specific Data	<ul style="list-style-type: none"> • Comment text • Post ID • Commenting user ID • Timestamp
Priority	High
Status	In development
Comments	Comments must pass a content validation process before being displayed.

Requirement ID	FR-006: Report Content
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Moderation engine
Description	The system must allow users to report posts they consider inappropriate, offensive, or harmful.
Specific Data	<ul style="list-style-type: none"> • Report reason • Post ID • Reporting user ID • Timestamp
Priority	High
Status	In development

Comments	Reports should trigger a review process by moderators.
-----------------	--

Non-Functional Requirements Specification

Requirement ID	NFR-001: Performance for Search
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Database • Caching engine
Description	The system must ensure that search queries and timeline loading are executed in under 100 milliseconds.
Specific Data	<ul style="list-style-type: none"> • Search engine response time • Timeline fetch latency
Priority	High
Status	Planned
Comments	Response time must be validated under peak load conditions.

Requirement ID	NFR-002: Scalability
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Cloud infrastructure • Load balancer
Description	The system must support elastic scaling to handle traffic surges during peak activity periods.
Specific Data	<ul style="list-style-type: none"> • Auto-scaling policies • Concurrent user support
Priority	High
Status	Planned
Comments	Stress testing should be performed before deployment.

Requirement ID	NFR-003: High Availability
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Infrastructure as Code • Group of geographically distributed servers
Description	The system must guarantee 99.9% uptime with regional redundancy to minimize service disruption.
Specific Data	<ul style="list-style-type: none"> • Regional server replicas • Downtime tolerance
Priority	High
Status	Planned
Comments	Use of failover systems and availability zones is required.

Requirement ID	NFR-004: Security Standards
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • OAuth provider • Encryption layer
Description	The platform must enforce encrypted data storage and secure authentication via OAuth, with API rate limiting.
Specific Data	<ul style="list-style-type: none"> • Encryption algorithm • Auth flow compliance • API rate thresholds
Priority	High
Status	Planned
Comments	Must comply with GDPR and platform privacy standards.

Requirement ID	NFR-005: Real-Time Data Ingestion
Version	1
Date	28/April/2025

Dependency	<ul style="list-style-type: none"> • OAuth provider • Encryption layer
Description	The platform must enforce encrypted data storage and secure authentication via OAuth, with API rate limiting.
Specific Data	<ul style="list-style-type: none"> • Encryption algorithm • Auth flow compliance • API rate thresholds
Priority	High
Status	Planned
Comments	Must comply with GDPR and platform privacy standards.

Requirement ID	NFR-006: Multi-Region Access
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Global CDN • Data replication system
Description	User data must be accessible from different regions with low latency, using synchronized replicas.
Specific Data	<ul style="list-style-type: none"> • Regional replication delay • Access latency
Priority	High
Status	Planned
Comments	Ensure consistency and data integrity across replicas.

Requirement ID	NFR-007: Real-Time Data Ingestion
Version	1
Date	28/April/2025

Dependency	<ul style="list-style-type: none"> • ML engine • Data pipeline
Description	The system must implement content-based and collaborative filtering algorithms for user feed personalization.
Specific Data	<ul style="list-style-type: none"> • User interaction data • Embedding vectors
Priority	Medium
Status	Planned
Comments	Must comply with GDPR and platform privacy standards.

Requirement ID	NFR-008: Business Intelligence Module
Version	1
Date	28/April/2025
Dependency	<ul style="list-style-type: none"> • Data warehouse • BI tool (Superset/Power BI)
Description	The system must integrate a BI module for generating insights on user engagement, content performance, and growth.
Specific Data	<ul style="list-style-type: none"> • User metrics • Post interactions • Regional trends
Priority	Medium
Status	Planned
Comments	Dashboards must be accessible to managers and admins.

Initial Database Architecture

Entity-Relationship Model

Initially the Entity-Relationship Model was designed based on the user stories and the requirements, the following is a step-by-step description of the first design of the Entity-Relationship Model.

1. Define components:

Two components were defined, one for users and the other for publications, in users we will take into account all the information, registration and login and the respective security modules according to the requirement, on the other hand, for the publications component we will take into account the interactions and comments, the images that are attached and the history of publications.

2. Define Entities:

	USER
e1	user
e2	follower
e3	report_user
e4	user_add_data
e5	user_account_conected
e6	cat_type_user
	POST
e7	post
e8	comment
e9	cat_reaction
e10	attached_multimedia
e11	historical_post
e12	reaction

Figure: Entities

3. Define Attributes per Entity:

	USER	
e1	user	id, date_create, date_modificate, date_delete, status, nickname, name, date_birth, password, email, profile_image
e2	follower	id, date_begin_follow
e3	report_user	id, date_report, reason, add_comment
e4	user_add_data	id, profile_description, phone, social_media
e5	user_account_connected	facebook_key, google_key
e6	cat_type_user	id, type, label, description
	POST	
e7	post	id, date_post, text_post, date_create, date_modificate, date_delete, status, visibility_type
e8	comment	id, text_comment, date_create, date_modificate, date_delete, status
e9	cat_reaction	id, reaction, label, description, date_reaction
e10	attached_multimedia	id, name, format, size, characteristics
e11	historical_post	date_modificate, text_version
e12	reaction	id, date_reaction

Figure: Attributes of each entity

4. Define Relationships:

	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10	e11	e12
e1	\	✓	✓	✓	✓	✓	✓	✓				✓
e2	✓	\										
e3	✓		\									
e4	✓			\								
e5	✓				\							
e6	✓					\						
e7	✓						\	✓		✓	✓	✓
e8	✓						✓	\				
e9									\			✓
e10							✓			\		
e11							✓				\	
e12	✓						✓		✓			\

Figure: Entity relationships

5. Define Relationships Types:

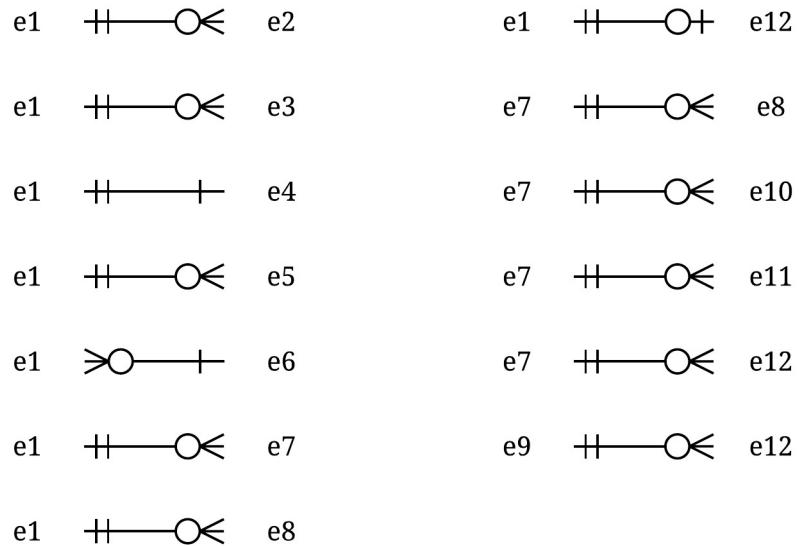


Figure: Relationship Types

6. First Entity-Relationship Model Draw:

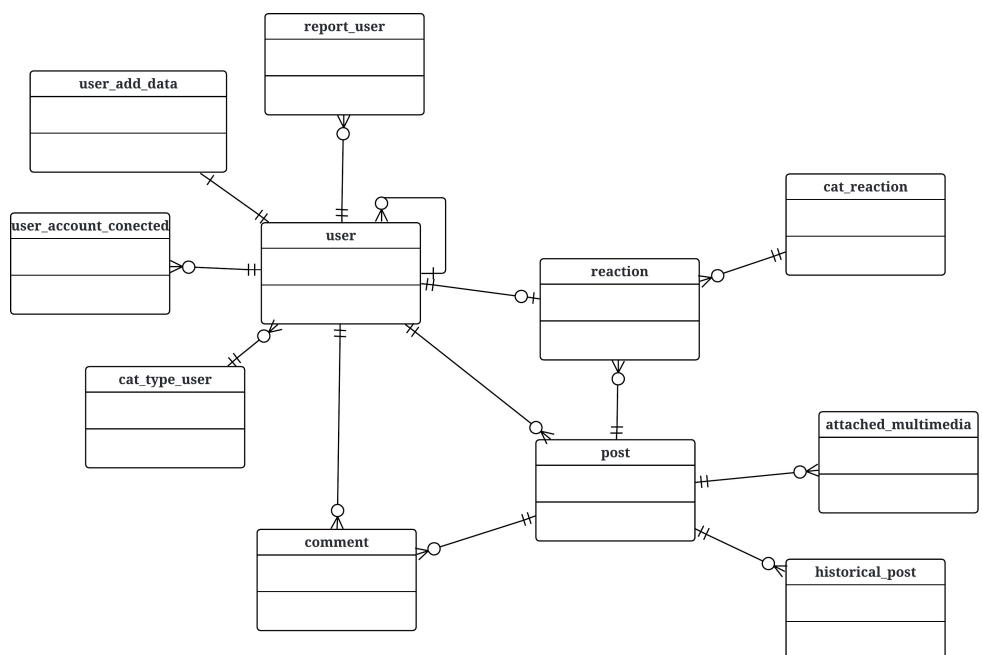


Figure: First Entity-Relationship Model

7. **Split Many-to-Many Relationships:** At this point we thought about the user being an entity with a reflective one-to-many relationship but we saw that it was a better option to separate it to have a follower entity.

8. Second Entity-Relationship Model Draw:

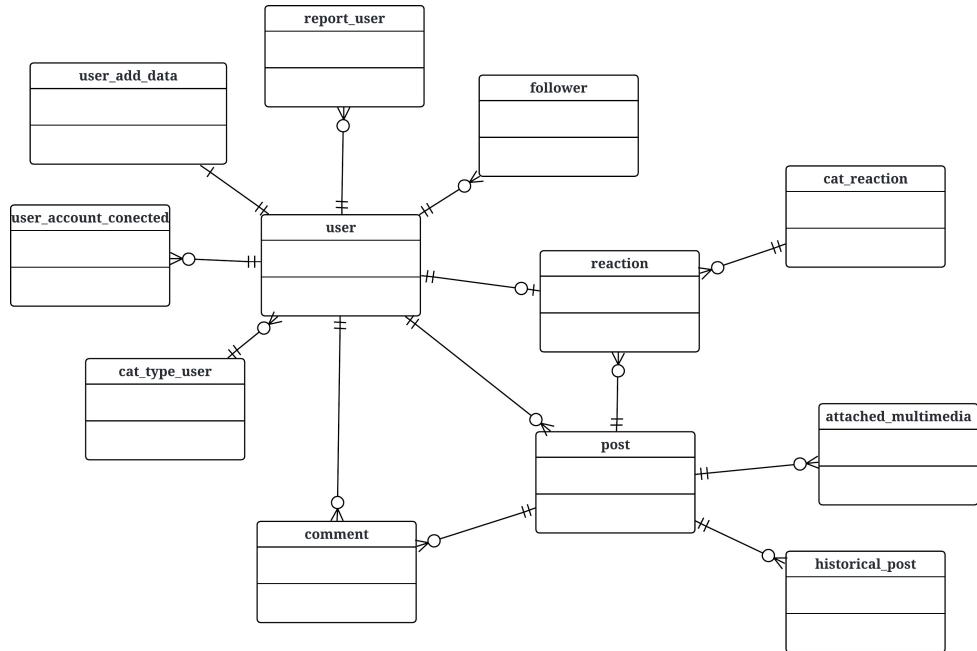


Figure: Second Entity-Relationship Model

9. Get Data-Structure Entity-Relationship Model:

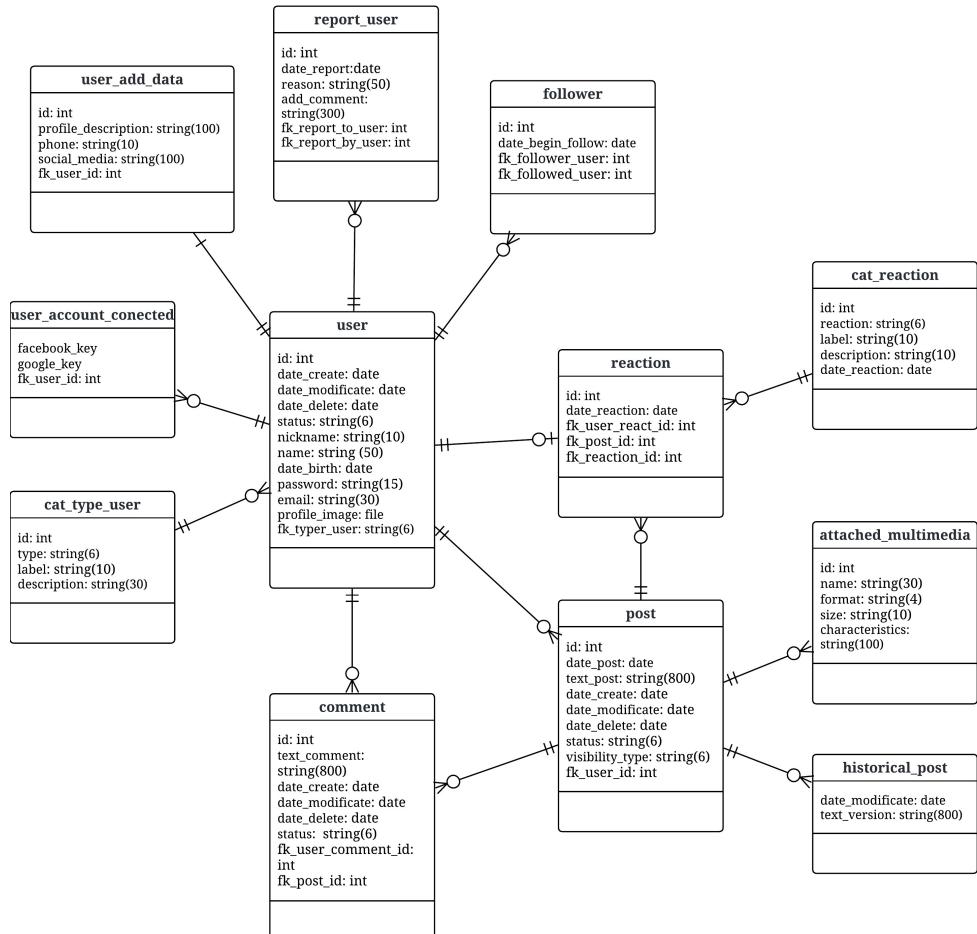


Figure: Entity-Relationship Model

10. Define Constraints and Properties of Data:

user	
id	PK, INT, AUTO_INCREMENT, NOT NULL
date_create	DATE, DEFAULT CURRENT_DATE, NOT NULL
date_modificate,	DATE, NULL
date_delete	
status	STRING(6), CHECK (status IN ('active', 'inactive', 'banned')), NOT NULL
nickname	STRING(10), UNIQUE, NOT NULL
email	STRING(30), UNIQUE, NOT NULL
password	STRING(60), NOT NULL (hashed)
profile_image	STRING(255), ruta o URL del archivo
fk_type_user	FK → cat_type_user(id), NOT NULL
cat_type_user	
id	PK
type	STRING(6), UNIQUE, NOT NULL
label	STRING(10), NOT NULL
description	STRING(30)
user_account_connected	
facebook_key,	STRING(100)
google_key	
fk_user_id	FK → user(id), NOT NULL, UNIQUE
user_add_data	
id	PK
profile_description	STRING(100)
phone	STRING(10), CHECK (phone LIKE '\d{10}')
social_media	STRING(100)
fk_user_id	FK → user(id), UNIQUE, NOT NULL
report_user	
id	PK
date_report	DATE, NOT NULL
reason	STRING(50), NOT NULL
add_comment	STRING(300)
fk_report_to_user,	FK → user(id), NOT NULL
fk_report_by_user	CHECK (fk_report_by_user ≠ fk_report_to_user)
follower	
id	PK
date_begin_follow	DATE, NOT NULL
fk_follower_user,	FK → user(id), NOT NULL
fk_followed_user	UNIQUE(fk_follower_user, fk_followed_user) CHECK (fk_follower_user ≠ fk_followed_user)

post	
id	PK
date_post	DATE, NOT NULL
text_post	STRING(800), NOT NULL
date_create,	DATE
date_modificate,	
date_delete	
status	STRING(6), CHECK (status IN ('active', 'deleted')), NOT NULL
visibility_type	STRING(6), CHECK (visibility_type IN ('public', 'private', 'friends'))
fk_user_id	FK → user(id), NOT NULL
comment	
id	PK
text_comment	STRING(800), NOT NULL
date_create,	DATE
date_modificate,	
date_delete	
status	STRING(6), CHECK (status IN ('visible', 'hidden')), NOT NULL
fk_user_comment_id	FK → user(id), NOT NULL
fk_post_id	FK → post(id), NOT NULL
reaction	
id	PK
date_reaction	DATE, DEFAULT CURRENT_DATE, NOT NULL
fk_user_react_id	FK → user(id), NOT NULL
fk_post_id	FK → post(id), NOT NULL
fk_reaction_id	FK → cat_reaction(id), NOT NULL
	UNIQUE(fk_user_react_id, fk_post_id)
cat_reaction	
id	PK
reaction	STRING(6), UNIQUE, NOT NULL
label, description	STRING(10)
date_reaction	DATE
attached_multimedia	
id	PK
name	STRING(30), NOT NULL
format	STRING(4), CHECK (format IN ('jpg', 'png', 'mp4', 'pdf'))
size	STRING(10)
characteristics	STRING(100)
historical_post	
date_modificate	DATE, NOT NULL
text_version	STRING(800), NOT NULL

After obtaining a first version of the entity-relationship model, we decided to make some corrections to separate the information and thus improve data processing. A second version of the entity-relationship model is presented below:

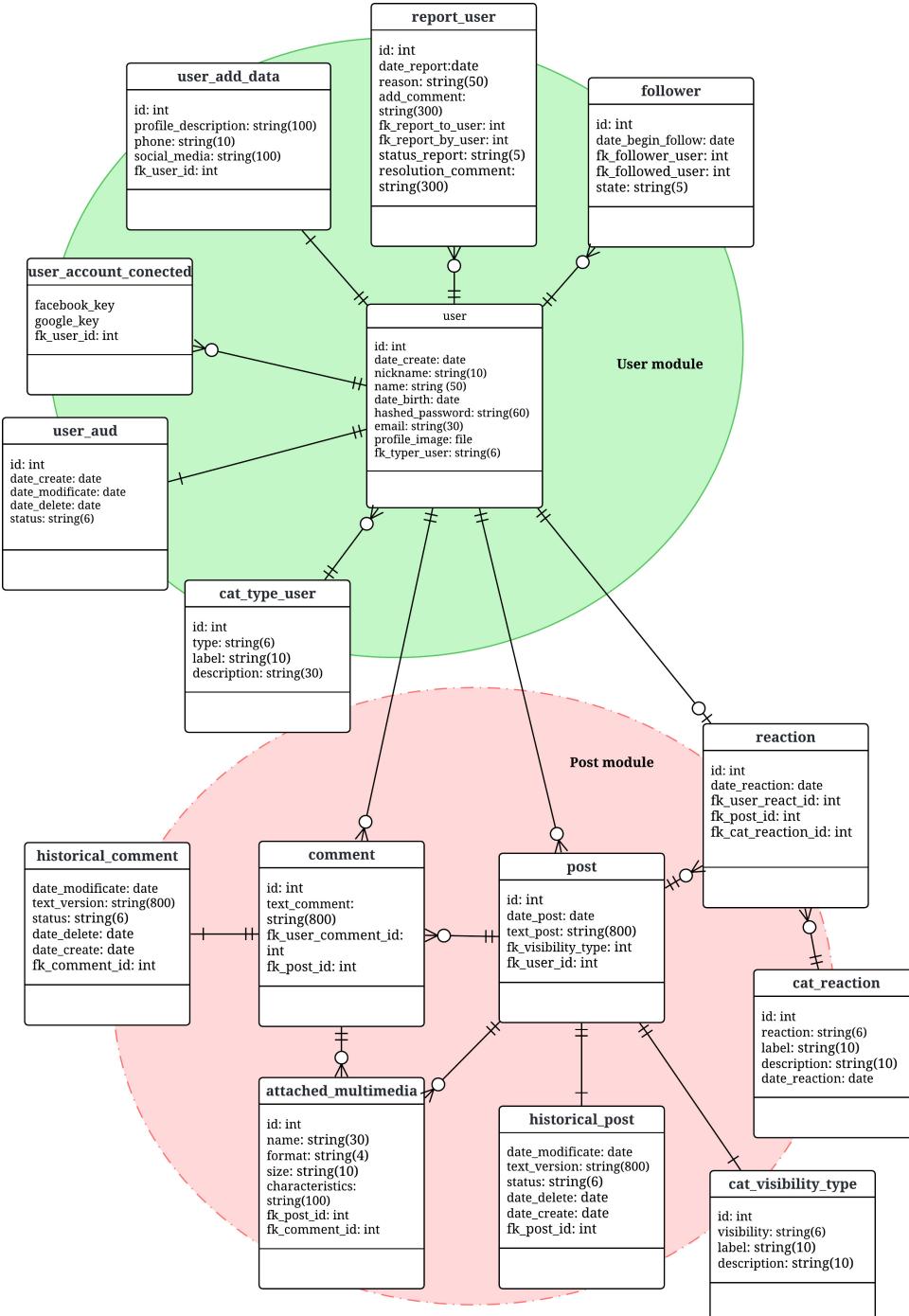


Figure: Second version of the Entity-Relationship Model

In the following figure it is represented according to the colors:

1. **White**: SQL entities because they are highly structured.

2. Gray: Non-SQL entities, which contain a large volume of information that has a flexible structure.

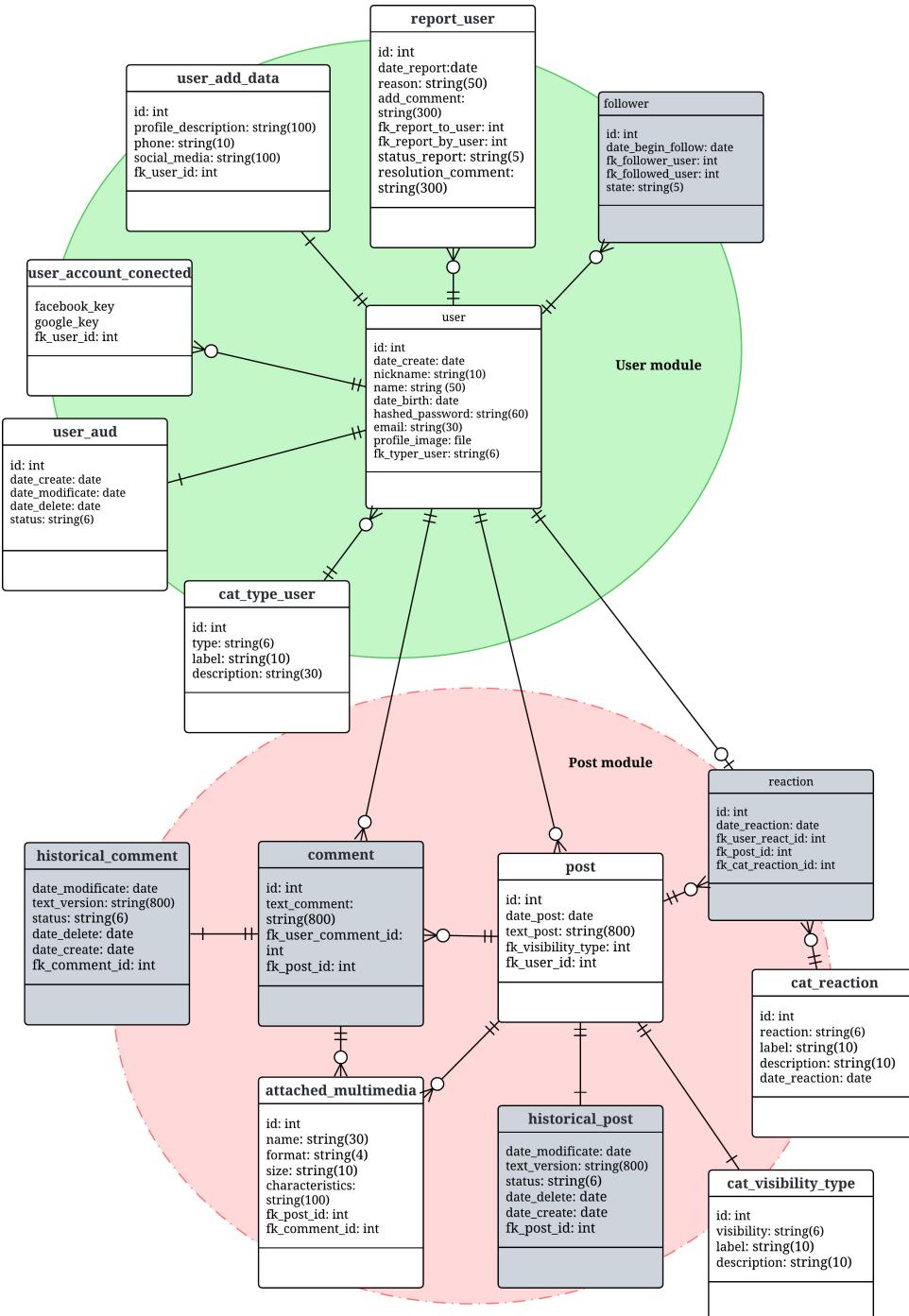


Figure: Explanation of the diagram