VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

DATABASE SYSTEMS (CO2013)

Assignment 1

Database Design Report

Topic: Task Management for Individuals or Teams

Instructor: Nguyễn Minh Tâm

Student(s): Võ Việt Huy - 2352417
Vũ Hoàng Long - 2352707
Lê Văn Hùng - 2152606
Bùi Quốc Thái - 2353086

Ho Chi Minh City, October 2025

Contents

Member list & Workload

No.	Fullname	Student ID	Missions	% done
1	Võ Việt Huy	2352417	- Describe the proposed system - EERD Design	100%
2	Vũ Hoàng Long	2352707	- Mapping to Database Schema - Data Dictionary	100%
3	Lê Văn Hùng	2152606	- Describe semantic constraints - Summarize and Write report	100%
4	Bùi Quốc Thái	2353086	- Research related applications/systems (reference) - Describe the proposed system	100%

1 Introduction

In the modern digital workspace, efficient project management and team collaboration have become essential components of organizational success. The growing complexity of distributed teams, remote work, and agile methodologies demands systems capable of supporting dynamic workflows, real-time communication, and structured task tracking. To address these needs, our team proposes the design and implementation of a **Task Management System** modeled after popular project management platforms such as *Trello*, *Asana*, *Monday.com*, *Jira*, and *Notion*.

This database-driven platform aims to enable users to create and manage workspaces, organize tasks into boards, lists, and cards, assign responsibilities, and monitor project progress effectively. The system integrates collaboration features such as commenting, activity tracking, and notifications while supporting extensibility through custom fields and automation rules. By designing this database from the ground up, we seek to model real-world data interactions and relationships that underpin professional project management systems.

The goal of this report is to analyze data requirements, construct an Enhanced Entity–Relationship Diagram (EERD), and map it into a fully normalized relational schema. This work serves as the foundation for further database implementation and demonstrates our understanding of conceptual and logical database design principles.

2 Data Requirements Analysis

2.1 Research Related Systems

The Project Management System is a collaborative platform designed to facilitate team-based project organization and task tracking through a workspace structure, drawing inspiration from established systems such as Trello, Asana, Monday.com, Jira, and Notion. Similar to how Trello pioneered Kanban-based task management, Asana excels in list and timeline views for complex project dependencies, Monday.com emphasizes visual workflow customization with color-coded boards, Jira focuses on agile software development with sprint tracking, and Notion combines project management with knowledge base capabilities, this system integrates the strengths of these platforms into a unified database architecture supporting multiple project management methodologies. The system accommodates two distinct types of users through a total disjoint specialization: Members, who are authenticated users with full system privileges including the ability to create, edit, and delete content after logging in with their credentials (similar to workspace administrators in Monday.com or board owners in Trello), and Monitors, who are view-only users granted token-based access on a per-workspace basis with configurable authentication but no modification capabilities (comparable to observer roles in Asana or guest access in Jira).

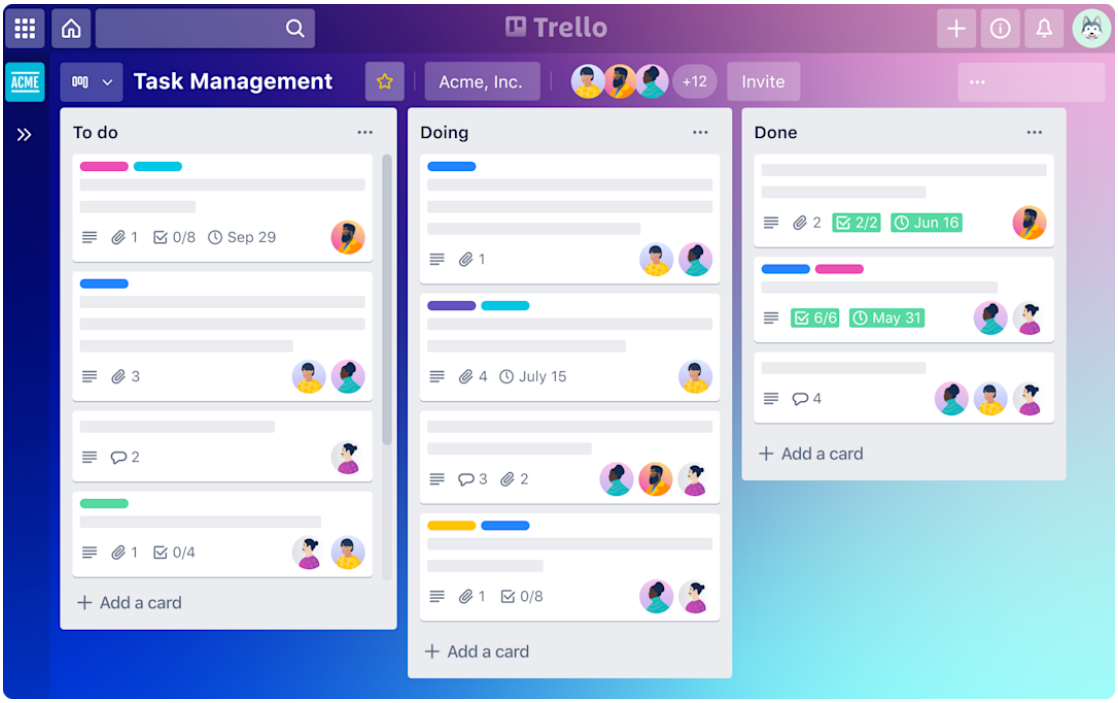


Image 1: Trello Task Management System[1]

The system provides comprehensive functionality for project management through workspace management with configurable billing plans and domain configurations (inspired by Notion's workspace structure and Monday.com's account hierarchies), board creation with customizable backgrounds and visibility settings (following Trello's flexible board paradigm), and hierarchical list and card management for detailed task tracking with priorities, dates, and member assignments (incorporating Asana's task dependency features and Jira's issue tracking capabilities). Additional collaborative features include checklist functionality for task decomposition (similar to Trello's checklist power-ups and ClickUp's subtask systems), commenting systems for team communication (mirroring discussion threads in Asana and activity streams in Monday.com), label categorization for visual organization (adopting Trello's label system and extending it with custom fields like those in Jira), file attachment capabilities for asset management (comparable to document storage in Notion and file sharing in Monday.com), and team formation to group members under designated managers (resembling team structures in Asana and workspace permissions in Jira).

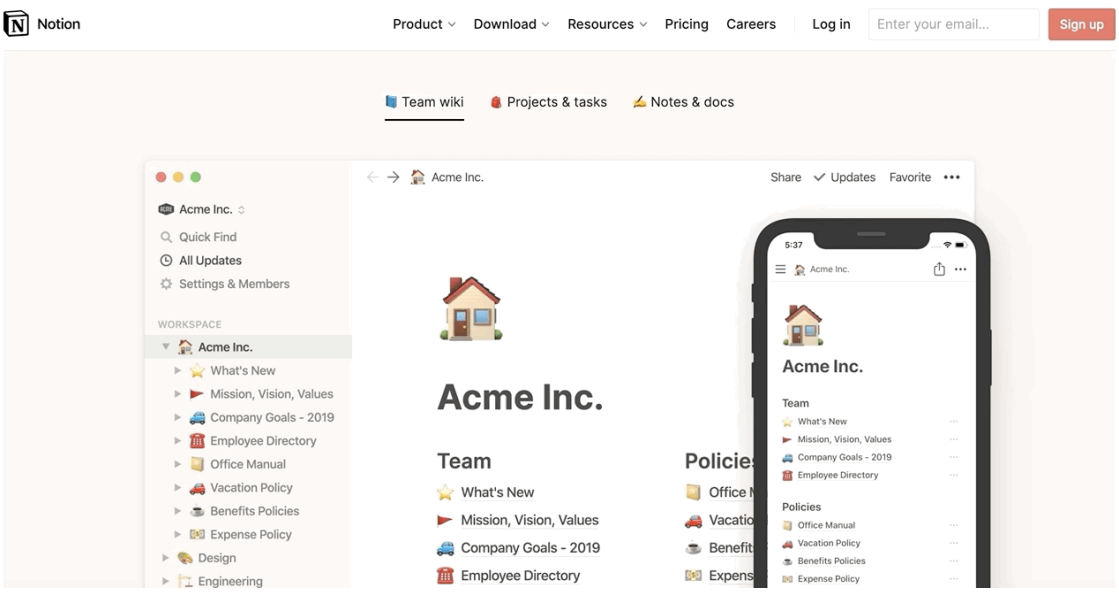


Image 2: Notion Workspace[2]

The platform also supports monitor invitation for external collaborators with limited permissions (inspired by external stakeholder access in Monday.com and guest users in Atlassian products), workflow automation through configurable rules with triggers and actions (building upon Butler automation in Trello, workflow automation in Asana, and custom automation recipes in Monday.com), and custom field definitions supporting multiple data types including Checkbox, Date, Dropdown, Number, and Text (drawing from Trello's custom fields power-up, Jira's extensive custom field system, and Monday.com's column type flexibility), allowing organizations to tailor the system to their specific project management needs across various methodologies including Kanban, Scrum, Waterfall, and hybrid approaches.

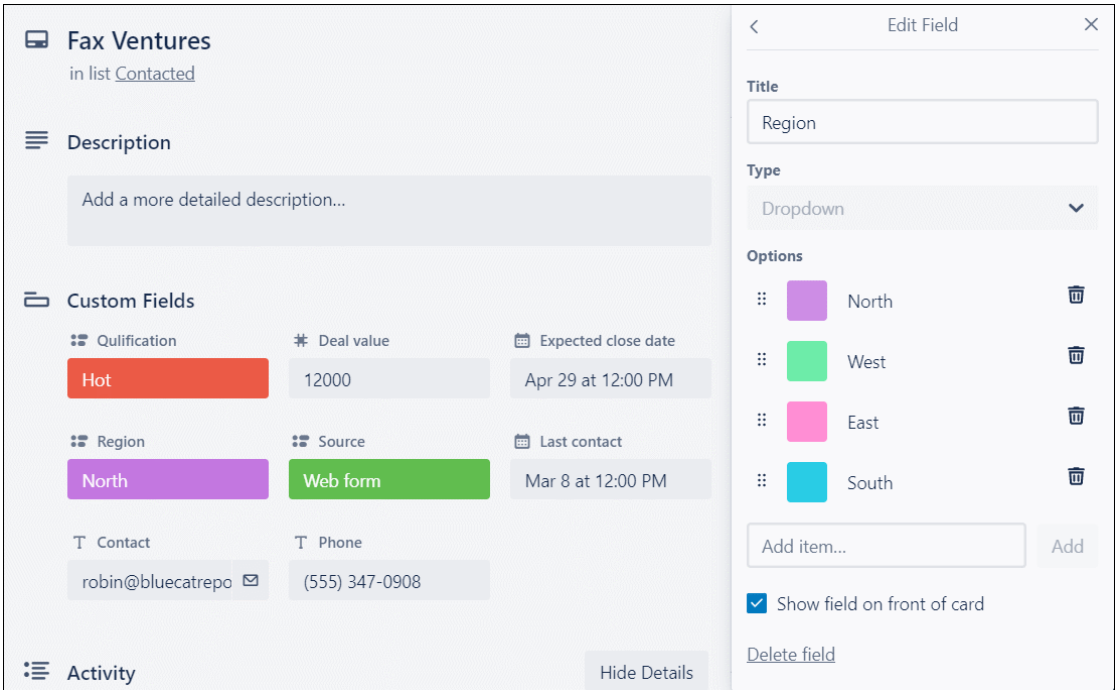


Image 3: Trello Custom Fields[3]

2.2 Proposed System Description

2.2.1 System Description

The proposed system is a collaborative project management platform that enables teams to organize projects, track tasks, and collaborate effectively in a workspace-based environment. The system implements a two-tier user structure: **Members** are authenticated users with full privileges to create, edit, and delete content, while **Monitors** are view-only stakeholders granted token-based access for observation without modification capabilities.

2.2.2 Main System Functions

The system provides hierarchical organization through workspaces, boards, lists, and cards. **Workspaces** serve as organizational containers with configuration and domain restrictions. **Boards** organize projects with customizable appearances and permissions, while **Lists** represent workflow stages and **Cards** function as individual tasks with detailed information including dates, priorities, descriptions, and overdue tracking. Enhanced organization features include color-coded labels for categorization, checklists for task decomposition, custom fields supporting multiple data types (Checkbox, Date, Dropdown, Number, Text), and file attachments for documentation.

Collaboration and automation capabilities enable effective teamwork through member assignments, role-based permissions, threaded commenting, and activity tracking that creates complete audit trails. Teams can be formed with designated managers, and external collaborators can be invited with controlled token-based access. The platform includes workflow automation

with configurable triggers and actions, a real-time notification system for updates and alerts, and comprehensive security features including role-based access control at workspace, board, and team levels.

2.3 Entity Types, Attributes and Relationship Description

The User entity serves as a [SUPERCLASS] that captures common attributes shared by all individuals in the system, including UUID as the primary identifier, a composite name attribute consisting of username, first_name and last_name components, a composite contact containing contact_email and multivalued phones which include area_code and phone_number, birth_date from which the derived age attribute is calculated, avatar_url for profile customization, and time_created timestamp tracking creation. The Member entity extends User as a [SUBCLASS], adding authentication-specific attributes such as login_email for login identification, password for secure credential storage, status controlling system access, and last_login timestamp tracking recent activity. The Monitor entity also extends User as a [SUBCLASS], incorporating attributes specific to external stakeholders including role representing their function per workspace.

The Workspace entity represents the highest organizational container with attributes including workspace_id as the primary identifier, name for identification, multivalued domains attribute storing authorized email domains, composite settings attribute containing various configuration options, time_created and time_updated timestamps for tracking changes, and billing_plan indicating the subscription level. Each workspace participates in relationships with Members through the is_member_of associative entity that includes role assignment and time_joined timestamp, maintains connections to boards through the include relationship, links to teams through the organize relationship, contains custom field definitions through the used_in relationship, and generates activity logs for audit purposes.

The Board entity organizes work within workspaces, containing attributes such as board_id for unique identification, name for identification within the workspace, description for context, composite settings attribute with visibility, permission, background_color, background_img, and position as sub-attributes, time_created and time_updated timestamps for tracking changes. Boards maintain relationships with lists through the include relationship, connect to Members through the is_member_of associative relationship with specific roles, and relate to labels through the defined relationships.

The List entity represents workflow stages within boards, featuring list_id as primary key, name for identification, and position determining display order. Lists contain cards through the include relationship and must maintain unique positions within their parent board. The Card entity serves as the fundamental work unit with extensive attributes including card_id for identification, name and description containing work details, position for ordering within lists, start_date and due_date defining timeframes, a derived is_overdue flag calculated from current date and due_date, time_completed timestamp recording finish time, and priority_level enumeration. Cards participate in numerous relationships including assignments to Members through the assigned_to relationship, categorization via the apply relationship with Labels that includes time_applied timestamp, custom field values through the belong_to relationship, comments for collaboration, attachments for documentation, and checklists for subtask tracking.

The Label entity provides categorical tagging with label_id as identifier, name for identification, color for visual distinction, and description for clarification. The Checklist entity enables subtask

management with checklist_id primary key, name for identification, and position for ordering, while maintaining an identifying relationship with ChecklistItem as a [WEAK] entity that cannot exist without its parent checklist. ChecklistItem includes checklist_item_id for identification within the checklist, description describing the task, position for ordering, time_completed timestamp, and connects to Members through the complete relationship.

The Comment entity facilitates discussion with comment_id as primary key, text containing the message, time_created timestamp, and is_edited flag if modified. Comments connect to cards through the include relationship and to Members through the post relationship. The Attachment entity manages file associations through attachment_id identification, name and url for access, size in bytes, type for handling, and time_uploaded timestamp. Attachments connect to cards through the include relationship and to Members through the upload relationship. The Activity entity manages activities in its card through activity_id identification, time_created and action. Activity stays in Cards and is referred to Members.

The CustomFieldDef entity defines workspace-level field templates with cfd_id as identifier, field_name and field_type specifying the data structure, default_value for initialization, time_created and time_updated timestamps. CustomFieldValue stores actual field data per card with value_id as identifier and value storing the actual data. The system connects custom field definitions to cards through CustomFieldValue entities.

The AutomationRule entity enables workflow automation with rule_id identifier, name and description for documentation, time_created and time_updated timestamps for tracking changes. The Log entity provides immutable audit records with log_id primary key, actor_type categorization, composite action attribute containing action_type and entity_affected sub-attributes, and time_created timestamp for sequencing. Log connect to users through the refer_to relationship and to workspaces through the generated relationship.

The Notification entity manages user alerts with noti_id identifier, type categorization, title and message content, time_delivered, time_read for tracking, and multivalued noti_options attribute. Users receive notifications through the notify relationship, and notifications may be triggered by activity logs. The Invitation entity controls access grants with invitation_id primary key, channel for communication method, and token for validation. Invitations connect workspaces to potential members through various relationship chains.

Team entities organize Members within workspaces through team_id identification, name and description for context, time_created and time_updated timestamps. Teams connect to workspaces via the organize relationship, to Members through the is_member_of relationship with role assignments, and to boards through the manage relationship for permission control. The system implements various monitoring relationships where Members and Monitors connect to workspaces and cards for tracking oversight activities.

2.4 Semantic Constraints

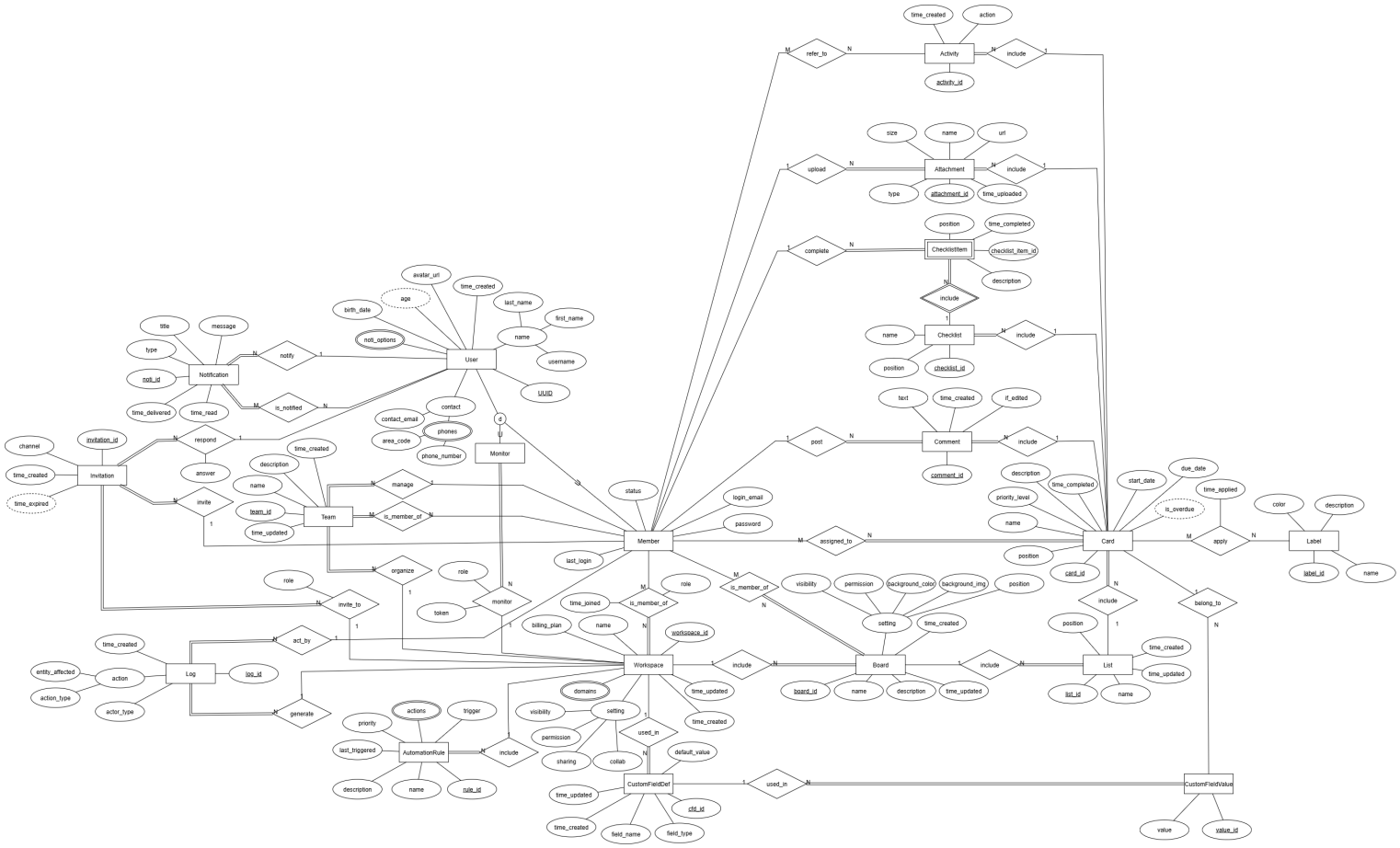
1. Phone numbers must be real and in use. All email addresses must conform to an email address standard. All URLs (avatar_url, attachment.url) must be valid HTTPS URLs.

2. Usernames and passwords must follow defined pattern rules (length, character set, profanity filtering). Passwords must meet complexity requirements and be stored as secure hashes only.

3. Position values must be unique within their scope, and must not be negative:
    - List positions unique within the board.
    - Card positions are unique within the list.
    - Checklist positions unique within the card.
    - ChecklistItem positions are unique within the checklist.
    - Moving items between positions must maintain uniqueness without gaps or conflicts. Position changes must efficiently maintain sort order indexes.
4. All timestamps must respect creation order (time_created $\leq$ time_updated $\leq$ now()). And the time_created of the child object must be $\leq$ time_created of its parent.
5. start_date $\leq$ due_date when both are present. due_date must be constrained as (due_date $<$ now() AND time_completed is NULL).
6. Only specific status changes are allowed (e.g., invited $\rightarrow$ active, active $\rightarrow$ suspended).
7. time_read must be NULL or $\geq$ time_delivered (NULL is not read)
8. CustomFieldValue must reference CustomFieldDef from the same workspace as the card's board.
9. Members assigned to cards must have membership in the card's board or workspace. Only members with appropriate scope can be assigned to cards.
10. Monitors cannot be assigned to cards or perform data modifications.
11. Custom Field type must be one of the following:
    - Checkbox Fields: Values must be boolean (true/false)
    - Date Fields: Values must be valid ISO 8601 datetime format
    - Number Fields: Values must be numeric and within defined range constraints
    - Text Fields: Values must respect maximum length
    - Dropdown Fields: Values must reference existing predefined options, no free text allowed. Must contain at least one option with all options unique within the dropdown.
12. Field type cannot be changed if any CustomFieldValue records exist unless explicit migration is performed.
13. Hierarchical Access:
    - Workspace membership grants board access unless visibility settings override.
    - Board membership grants access only to that board and its cards.
14. Activity and Log records cannot be updated or deleted once created. You must reset the database to be able to force that.
15. Log entries must maintain chronological sequence.
16. Each logged action must identify the responsible actor (Member, System, or AutomationRule).

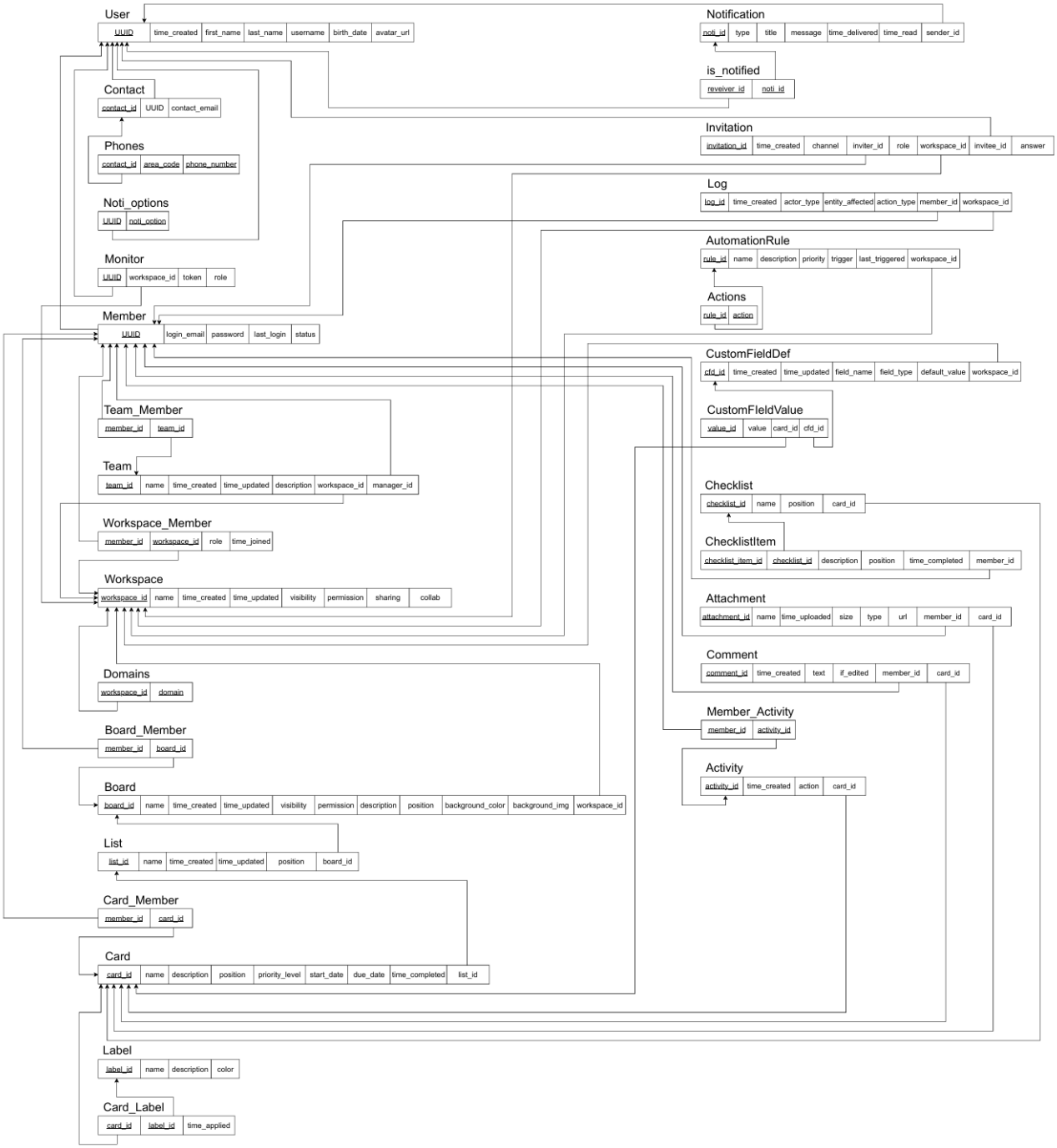
17. Automation rules must not create infinite loops through chain reactions. Rules cannot trigger themselves directly or indirectly within the same execution context.
18. Rules with identical priority (can be position) must execute in deterministic order. Rule actions must be idempotent to prevent duplicate effects during retries.
19. Workspace limits (boards, members, storage) must be enforced based on billing_plan.
20. File size and type restrictions must be enforced based on workspace policies. If the file is too large, suggest dividing the file to smaller files. Duplicate file detection in each card.
21. Notifications and invitations can be deleted / declined automatically after a long time period (default is 90 days of user's inactivity).
22. Email and notification sending must respect rate limits and anti-spam policies.

3 Enhanced Entity-Relationship Diagram (EERD)



4 Database Schema Mapping

4.1 Relational Schema Diagram



4.2 Data Dictionary

RELATION NAME	ATTRIBUTE NAME	PK	FK	FK with REFERENCE to	NULLABLE	DATA TYPE	DOMAIN	DESCRIPTION
User	UUID	PK			NOT NULL	UUID		Unique user identifier
	time_created				NOT NULL	TIMESTAMP		Account creation timestamp
	first_name				NOT NULL	VARCHAR(100)		User first name
	last_name				NOT NULL	VARCHAR(100)		User last name
	username				NOT NULL	VARCHAR(50)		Unique username for identification
	birth_date					DATE		Date of birth
	avatar_url					VARCHAR(255)		HTTPS URL to profile picture
Contact	contact_id	PK			NOT NULL	UUID		Contact record identifier
	UUID	PK	FK	User.UUID	NOT NULL	UUID		Reference to user
	contact_email				NOT NULL	VARCHAR(255)		User email address
Phones	contact_id	PK	FK	Contact.contact_id	NOT NULL	UUID		Reference to contact
	area_code	PK			NOT NULL	VARCHAR(10)		Phone area code
	phone_number	PK			NOT NULL	VARCHAR(20)		Phone number
Noti_options	UUID	PK	FK	User.UUID	NOT NULL	UUID		Reference to user
	noti_option	PK			NOT NULL	VARCHAR(50)		Notification preference option
Monitor	UUID	PK	FK	User.UUID	NOT NULL	UUID		Monitor user identifier
	workspace_id		FK	Workspace.workspace_id		UUID		Monitored workspace
	token					VARCHAR(255)		Access token for monitoring
	role				NOT NULL	VARCHAR(50)		Monitor role in workspace
Member	UUID	PK	FK	User.UUID	NOT NULL	UUID		Member user identifier
	login_email				NOT NULL	VARCHAR(255)		Unique email for login
	password				NOT NULL	VARCHAR(255)		Hashed password
	last_login				NOT NULL	TIMESTAMP		Last login timestamp
	status				NOT NULL	ENUM	invited, active, suspended	Account status
Team_Member	member_id	PK	FK	Member.UUID	NOT NULL	UUID		Team member identifier
	team_id	PK	FK	Team.team_id	NOT NULL	UUID		Team identifier
Team	team_id	PK			NOT NULL	UUID		Unique team identifier
	name				NOT NULL	VARCHAR(100)		Team name
	time_created				NOT NULL	TIMESTAMP		Team creation timestamp
	time_updated				NOT NULL	TIMESTAMP		Last update timestamp
	description					TEXT		Team description
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Parent workspace
	manager_id		FK	Member.UUID		UUID		Team manager
Workspace_Member	member_id	PK	FK	Member.UUID	NOT NULL	UUID		Workspace member identifier
	workspace_id	PK	FK	Workspace.workspace_id	NOT NULL	UUID		Workspace identifier
	role				NOT NULL	VARCHAR(50)		Member role in workspace
	time_joined				NOT NULL	TIMESTAMP		Membership start timestamp
Workspace	workspace_id	PK			NOT NULL	UUID		Unique workspace identifier
	name				NOT NULL	VARCHAR(100)		Workspace name
	time_created				NOT NULL	TIMESTAMP		Workspace creation timestamp
	time_updated				NOT NULL	TIMESTAMP		Last update timestamp
	visibility				NOT NULL	ENUM	public, private	Workspace visibility
	permission				NOT NULL	VARCHAR(50)		Default permission level
	sharing					VARCHAR(50)		Sharing settings
	collab					VARCHAR(50)		Collaboration settings
Domains	workspace_id	PK	FK	Workspace.workspace_id	NOT NULL	UUID		Workspace identifier
	domain	PK			NOT NULL	VARCHAR(255)		Authorized email domain
Board_Member	member_id	PK	FK	Member.UUID	NOT NULL	UUID		Board member identifier
	board_id	PK	FK	Board.board_id	NOT NULL	UUID		Board identifier
Board	board_id	PK			NOT NULL	UUID		Unique board identifier
	name				NOT NULL	VARCHAR(100)		Board name
	time_created				NOT NULL	TIMESTAMP		Board creation timestamp
	time_updated				NOT NULL	TIMESTAMP		Last update timestamp
	visibility				NOT NULL	ENUM	workspace, private, public	Board visibility level
	permission				NOT NULL	VARCHAR(50)		Board permission settings
	description					TEXT		Board description
	position				NOT NULL	INT		Display order in workspace
	background_color					VARCHAR(20)		Background color hex
	background_img					VARCHAR(255)		Background image URL
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Parent workspace
List	list_id	PK			NOT NULL	UUID		Unique list identifier
	name				NOT NULL	VARCHAR(100)		List name
	time_created				NOT NULL	TIMESTAMP		List creation timestamp
	time_updated				NOT NULL	TIMESTAMP		Last update timestamp
	position				NOT NULL	INT		Display order in board
	board_id		FK	Board.board_id	NOT NULL	UUID		Parent board
Card_Member	member_id	PK	FK	Member.UUID	NOT NULL	UUID		Assigned member identifier
	card_id	PK	FK	Card.card_id)	NOT NULL	UUID		Card identifier
	card_id	PK			NOT NULL	UUID		Unique card identifier

Card	name				NOT NULL	VARCHAR(255)		Card title
	description					TEXT		Card detailed description
	position				NOT NULL	INT		Display order in list
	priority_level				NOT NULL	ENUM	low, medium, DEFAULT, high, critical	Task priority
	start_date					TIMESTAMP		Task start date
	due_date					TIMESTAMP		Task due date
	time_completed					TIMESTAMP		Completion timestamp
	list_id		FK	List.list_id	NOT NULL	UUID		Parent list
Label	label_id	PK			NOT NULL	UUID		Unique label identifier
	name				NOT NULL	VARCHAR(50)		Label name
	description					TEXT		Label description
	color				NOT NULL	VARCHAR(20)		Label color hex
Card_Label	card_id	PK	FK	Card.card_id	NOT NULL	UUID		Card identifier
	label_id	PK	FK	Label.label_id	NOT NULL	UUID		Label identifier
	time_applied				NOT NULL	TIMESTAMP		Label application timestamp
Notification	noti_id	PK			NOT NULL	UUID		Unique notification identifier
	type				NOT NULL	VARCHAR(50)		Notification type
	title				NOT NULL	VARCHAR(255)		Notification title
	message				NOT NULL	TEXT		Notification message content
	time_delivered				NOT NULL	TIMESTAMP		Delivery timestamp
	time_read					TIMESTAMP		Read timestamp
	sender_id		FK	User.UUID		UUID		Notification sender
is_notified	reveiver_id	PK	FK	User.UUID	NOT NULL	UUID		Notification receiver
	noti_id	PK	FK	Notification.noti_id	NOT NULL	UUID		Notification identifier
Invitation	invitation_id	PK			NOT NULL	UUID		Unique invitation identifier
	time_created				NOT NULL	TIMESTAMP		Invitation creation timestamp
	channel				NOT NULL	VARCHAR(50)		Invitation delivery channel
	inviter_id		FK	Member.UUID	NOT NULL	UUID		Inviting member
	role				NOT NULL	VARCHAR(50)		Invited role
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Target workspace
	invitee_id		FK	User.UUID		UUID		Invited user
	answer				NOT NULL	ENUM	pending, accepted, declined	Invitation response status
Log	log_id	PK			NOT NULL	UUID		Unique log entry identifier
	time_created				NOT NULL	TIMESTAMP		Log entry timestamp
	actor_type				NOT NULL	ENUM	member, monitor, system, automation	Type of actor
	entity_affected				NOT NULL	VARCHAR(100)		Affected entity type
	action_type				NOT NULL	VARCHAR(50)		Action performed
	member_id		FK	Member.UUID		UUID		Actor member
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Workspace context
AutomationRule	rule_id	PK			NOT NULL	UUID		Unique rule identifier
	name				NOT NULL	VARCHAR(100)		Rule name
	description					TEXT		Rule description
	priority				NOT NULL	INT		Execution priority
	trigger				NOT NULL	TEXT		Trigger condition
	last_triggered					TIMESTAMP		Last execution timestamp
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Parent workspace
Actions	rule_id	PK	FK	AutomationRule.rule_id	NOT NULL	UUID		Rule identifier
	action	PK			NOT NULL	TEXT		Action to perform
CustomFieldDef	cfd_id	PK			NOT NULL	UUID		Unique field definition identifier
	time_created				NOT NULL	TIMESTAMP		Definition creation timestamp
	time_updated				NOT NULL	TIMESTAMP		Last update timestamp
	field_name				NOT NULL	VARCHAR(100)		Field name
	field_type				NOT NULL	ENUM	Checkbox, Date, Number, Text, Dropdown	Field data type
	default_value					TEXT		Default field value
	workspace_id		FK	Workspace.workspace_id	NOT NULL	UUID		Parent workspace
CustomFieldValue	value_id	PK			NOT NULL	UUID		Unique value identifier
	value					TEXT		Field value
	card_id		FK	Card.card_id	NOT NULL	UUID		Associated card
	cfd_id		FK	CustomFieldDef.cfd_id)	NOT NULL	UUID		Field definition
Checklist	checklist_id	PK			NOT NULL	UUID		Unique checklist identifier
	name				NOT NULL	VARCHAR(100)		Checklist name
	position				NOT NULL	INT		Display order in card
	card_id		FK	Card.card_id	NOT NULL	UUID		Parent card
ChecklistItem	checklist_item_id	PK			NOT NULL	UUID		Item identifier within checklist
	checklist_id	PK	FK	Checklist.checklist_id)	NOT NULL	UUID		Parent checklist
	description				NOT NULL	TEXT		Item description
	position				NOT NULL	INT		Display order in checklist
	time_completed					TIMESTAMP		Completion timestamp
	member_id		FK	Member.UUID		UUID		Completing member
	attachment_id	PK			NOT NULL	UUID		Unique attachment identifier

Attachment	name				NOT NULL	VARCHAR(255)		File name
	time_uploaded				NOT NULL	TIMESTAMP		Upload timestamp
	size				NOT NULL	BIGINT		File size in bytes
	type				NOT NULL	VARCHAR(50)		File MIME type
	url				NOT NULL	VARCHAR(255)		HTTPS file URL
	member_id		FK	Member.UUID	NOT NULL	UUID		Uploader member
	card_id		FK	Card.card_id	NOT NULL	UUID		Associated card
Comment	comment_id	PK			NOT NULL	UUID		Unique comment identifier
	time_created				NOT NULL	TIMESTAMP		Comment creation timestamp
	text				NOT NULL	TEXT		Comment text content
	if_edited				NOT NULL	BOOLEAN		Edit flag
	member_id		FK	Member.UUID	NOT NULL	UUID		Comment author
	card_id		FK	Card.card_id	NOT NULL	UUID		Associated card
Member_Activity	member_id	PK	FK	Member.UUID	NOT NULL	UUID		Activity member
	activity_id	PK	FK	Activity.activity_id	NOT NULL	UUID		Activity identifier
Activity	activity_id	PK			NOT NULL	UUID		Unique activity identifier
	time_created				NOT NULL	TIMESTAMP		Activity timestamp
	action				NOT NULL	VARCHAR(50)		Action type
	card_id		FK	Card.card_id	NOT NULL	UUID		Associated card

5 Conclusion

This report presents a complete database design for a Task Management System that addresses the essential requirements of modern project management and team collaboration. Through careful analysis of data requirements and real-world interactions found in established platforms like Trello, Asana, Monday.com, Jira, and Notion, we have developed a robust database architecture that supports dynamic workflows, real-time communication, and structured task tracking across distributed teams.

This database design provides a solid foundation for implementation and serves as a practical demonstration of how theoretical database concepts translate into functional systems that address real-world organizational needs. The comprehensive structure supports scalability from small team coordination to enterprise-level project management, positioning the system as a versatile solution for diverse organizational contexts and use cases.

References

[1] https://trello.com/en/use-cases/task-management

[2] https://niftypm.com/blog/notion-vs-evernote/

[3] https://www.bluecatreports.com/blog/trello-custom-fields-how-to-and-examples/