Tasks-O-Matic

Re-Design: Database

Section 1: Database Technology

Technology: Relational Database

The Tasks-O-Matic will allow users to make groups of tasks items. These tasks items can be reused in multiple groups. To do this, the project MVP will use a Relational database (Postgres) with a Many-to-Many relation between <mark>four</mark> tables.

- <mark>Users</mark>
- Groups
- Task Groups
- Task Items

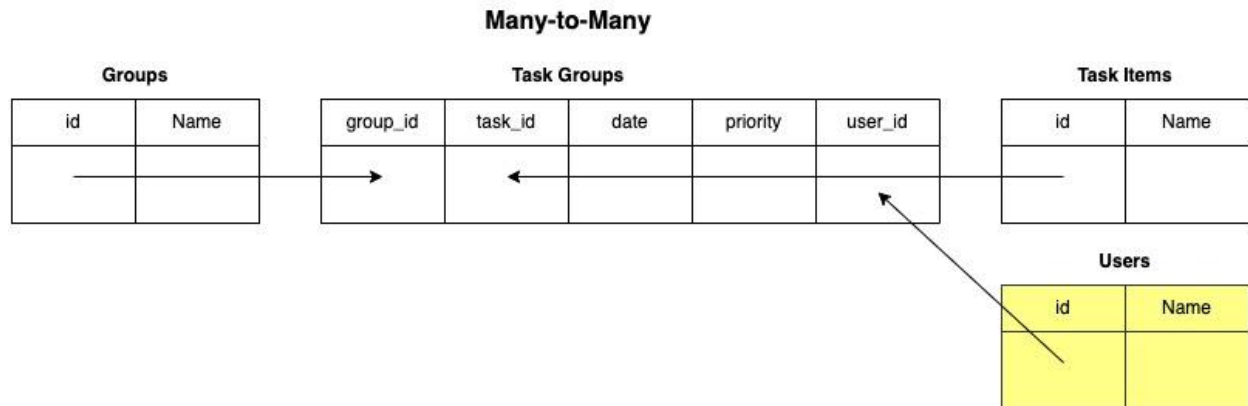This project will persist data in a relational data store because:

- <mark>A user can be assigned infinite Task Groups.</mark>
- Task items can be reused in infinite Task Groups.
- Task Groups table's task_id (FK) columns will change often.
- Data changes will cascade to the Task Groups table from other tables.

<mark>Each Task Group can have a User related to it. A user can be related to infinite Groups, so the relation will be One- to-Many.</mark>

Each Task Group will have a single-use Group related to it, so Group to Task Groups will be One-to-One. Task Groups will take group_id as a foreign key.
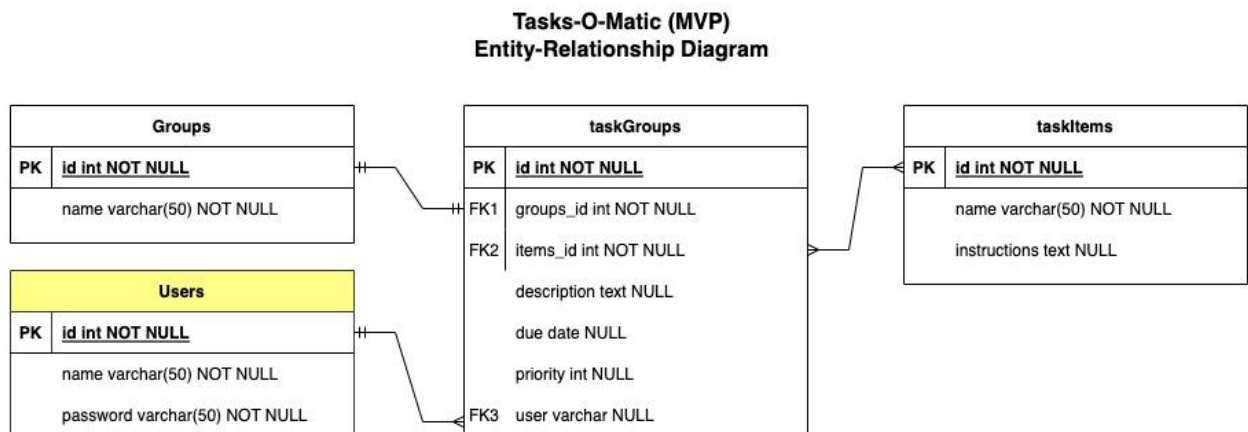
Each Task Group will have infinite related Task Items, so Task Items to Task Groups will be One-to-Many. Task Groups will take task_id as a foreign key.

Task Groups will serve as a Many-to-Many relational data bridge between Groups to Task Items. The Bridge table will store common data from both tables and will be updated as data from either table change.

Many-to-Many

Groups		Task Groups					Task Items	
id	Name	group_id	task_id	date	priority	user_id	id	Name

Users	
id	Name

Section 2: Data Structure

The Tasks-O-Matic (MVP) will use the following data structure:

Tasks-O-Matic (MVP)
Entity-Relationship Diagram

Groups	
PK	id int NOT NULL
	name varchar(50) NOT NULL

Users	
PK	id int NOT NULL
	name varchar(50) NOT NULL
	password varchar(50) NOT NULL

taskGroups	
PK	id int NOT NULL
FK1	groups_id int NOT NULL
FK2	items_id int NOT NULL
	description text NULL
	due date NULL
	priority int NULL
FK3	user varchar NULL

taskItems	
PK	id int NOT NULL
	name varchar(50) NOT NULL
	instructions text NULL

Section 3: Data Use

The Tasks-O-Matic will follow the Third Normal Form. The Users table will authenticate user login credentials. The taskGroup table will take foreign keys user_id (Users table), group_id (Groups table), and items_id (taskItems table). To avoid redundancy, taskGroups will reject redundant unique key (FK1, FK2) entries.

Persona Example Table Use:

Sarah Manager
      Users: Sarah will create a user account to login to the application.
      Groups: Sarah will create several groups for employees by area of responsibility.
      taskItems: Sarah will create a list of tasks to be completed.
      taskGroups: Sarah will assign the individual tasks to the appropriate Groups.
                    Sarah will add a user to the taskGroup.

Sarah will view a report based on taskGroups' data.