Table name: core_project

The core project table has the project names, the start date and end date of each project.

Table name: allowance_employee_role

The allowance_employee_role table tracks the association between employee roles and allowances within an organizational structure. It contains the following columns: created_at and updated_at record the timestamps for when the record was created and last updated, respectively; id serves as the primary key for unique identification of each entry. The allowance_id column links to the id field in the allowance table, signifying the specific allowance associated with the role. The created_by_id column references the id field in the auth_user table, identifying the user who created the entry. The resource_code_uuid stores a unique identifier for the resource, while resource_id links to the id in the cost_sheet_resources table, representing the associated cost resource. Finally, tenant_id references the id in the tenant table, specifying the tenant or organizational context of the record. These relationships enable the table to connect employee roles, allowances, users, and organizational resources comprehensively.

Table name: categories

The columns of this table is as follows:

1. **created_at**: This column stores the timestamp when the category record was created. It helps track the creation time of each entry.
2. **updated_at**: Similar to created_at, this column holds the timestamp of the last update made to the category record. It ensures that any modifications are recorded for auditing purposes.
3. **name**: The name column holds the name of the category. This is a key attribute used to identify the category in the system.
4. **slug**: The slug is a URL-friendly version of the category's name. Typically, it is used for creating SEO-friendly URLs or paths for categories (e.g., /categories/{slug}).
5. **description**: This column stores a textual description of the category, which provides more detailed information about its purpose or contents.
6. **default**: The default column likely stores a boolean value indicating whether this category is the default or main category within the system.
7. **id**: This is the primary key of the table and uniquely identifies each category record.
8. **created_by_id**: This column references the id of the user who created the category. It is a foreign key linked to the auth_user table, helping track who was responsible for creating a particular category.
9. **parent_id**: The parent_id is used to establish hierarchical relationships between categories. If the category is a subcategory, this field will reference the id of its parent category. It allows for nested or tree-like category structures.
10. **tenant_id**: The tenant_id is a foreign key that links the category to a specific tenant or organization within a multi-tenant system. It references the id in the tenant table, ensuring that categories are scoped to a particular tenant and not shared across multiple tenants.

Relationships:

1. **created_by_id references auth_user(id)**: This foreign key constraint ensures that every category is associated with a user who created it. The auth_user table typically stores information about application users, so this relationship allows you to track which user is responsible for creating a given category.
2. **parent_id references categories(id)**: This self-referencing foreign key relationship enables the creation of hierarchical categories. If a category is a subcategory, its parent_id will point to the id of its parent category, forming a parent-child relationship between categories.
3. **tenant_id references tenant(id)**: In a multi-tenant environment, this foreign key ensures that categories belong to a specific tenant or organization. It helps separate data for different tenants, ensuring that categories are scoped properly and are only visible to the relevant tenant.

Usage:

The categories table is crucial for organizing data into hierarchical structures, allowing categories to have a parent-child relationship. The relationships to other tables such as auth_user and tenant provide further context, allowing the system to track who created each category and which tenant it belongs to. The slug and name columns ensure that the categories are easily identifiable and can be accessed via user-friendly URLs, while the default column helps identify the main or default category. The structure of this table is essential for maintaining a well-organized categorization system, especially in applications with large amounts of data or multiple tenants.