◎ Recipe Management System – Refined Data Model

✇ Conventions

- Tables and columns in **snake_case**, in **singular**.
- Primary keys type **UUID**, generated with `gen_random_uuid()` (`CREATE EXTENSION pgcrypto;`)
- **Audit fields (7):**

    - `status`
    - `created_at`
    - `updated_at`
    - `deleted_at`

- Foreign keys use **ON UPDATE CASCADE** and appropriate **ON DELETE** rules for the domain
- Unique constraints on names and codes where applicable

⬡ Module: Recipe Management

- **recipe**
  `{id, title, description, preparation_time, difficulty, category_id, cooking_technique_id, source_id, status, created_at, updated_at, deleted_at}`

- **ingredient**
  `{id, recipe_id, name, quantity, unit, status, created_at, updated_at, deleted_at}`

- **step_instruction**
  `{id, recipe_id, order, description, status, created_at, updated_at, deleted_at}`

- **category**
  `{id, name, status, created_at, updated_at, deleted_at}`

- **cooking_technique**
  `{id, name, description, status, created_at, updated_at, deleted_at}`

- **source**
  `{id, name, type, description, status, created_at, updated_at, deleted_at}`

- **utensil**
  `{id, name, description, status, created_at, updated_at, deleted_at}`

- **recipe_utensil**
  `{id, recipe_id, utensil_id, status, created_at, updated_at, deleted_at}`

☗ Module: User Management

- **user**
  `{id, name, email, registration_date, status, created_at, updated_at, deleted_at}`

○ Module: Comment and Rating Management

- **comment**
  `{id, recipe_id, user_id, text, date, status, created_at, updated_at, deleted_at}`

- **rating**
  `{id, recipe_id, user_id, score, date, status, created_at, updated_at, deleted_at}`

▣ Key Relationships

- `recipe` 1 --- * `ingredient`
- `recipe` 1 --- * `step_instruction`
- `recipe` * --- * `utensil` *(through `recipe_utensil`)*
- `recipe` * --- 1 `category`
- `recipe` * --- 1 `cooking_technique`
- `recipe` * --- 1 `source`

- recipe 1 — * comment
- recipe 1 — * rating
- user 1 — * comment
- user 1 — * rating