

Meal Planner DB Schema & Data Structures (v0.1)

This document outlines the core relational schema required to support the initial API contract and core user loop (Discover, Plan, Home, Shop, Insights). Auth/admin tables are intentionally omitted for now.

1. Users & Preferences

1.1 users

- id (PK)
- email (unique, not null)
- name
- created_at (timestamp)

1.2 user_preferences

- user_id (PK, FK -> users.id)
- diet_type (text)
- calories_target (int)
- exclude_ingredients (int[] of ingredient_ids)
- cuisine_likes (text[])
- cuisine_dislikes (text[])
- effort_level (text: 'low'|'medium'|'high')
- updated_at (timestamp)

Used by: /users/me/preferences, reco engine, plan generation, home summary.

2. Recipes & Ingredients

2.1 recipes

- id (PK)
- title (text, not null)
- description (text)
- image_url (text)
- cook_time_min (int)
- servings (int, default 2)
- created_at (timestamp)

2.2 ingredients

- id (PK)
- name (text, not null)
- category (text, not null) -- 'meat','veg','pantry',...
- canonical_unit (text, not null) -- 'g','ml','piece'
- aggregation_mode (text, default 'sum') -- 'sum' or 'presence'
- grams_per_unit (numeric) -- optional, for discrete units/portions
- kcal_per_gram (numeric)
- protein_per_gram (numeric)
- carbs_per_gram (numeric)
- fat_per_gram (numeric)

2.3 recipe_ingredients

- id (PK)
- recipe_id (FK -> recipes.id)
- ingredient_id (FK -> ingredients.id)
- amount_canonical (numeric, not null) -- in canonical_unit
- canonical_unit (text, not null)

- display_text (text, not null) -- '1 chicken breast (~150g)'
- quality (text, default 'auto') -- 'auto','suspect','manual'
- created_at (timestamp)

Used by: recipe detail, shopping list amalgamator, nutrition and cost estimation.

3. Meal Plans & Plan Meals

3.1 meal_plans

- id (PK)
- user_id (FK -> users.id)
- start_date (date, not null)
- end_date (date, not null)
- created_at (timestamp)
- UNIQUE (user_id, start_date) -- one plan per user per week

3.2 plan_meals

- id (PK)
- meal_plan_id (FK -> meal_plans.id)
- date (date, not null)
- meal_type (text, not null) -- 'dinner' (for now)
- recipe_id (FK -> recipes.id)
- position (int) -- optional ordering within the day
- created_at (timestamp)

Used by: /meal-plans*, /home (upcoming meal), /shop, /insights/weekly.

4. Interactions, Likes, Reviews, Suggestions

4.1 user_interactions

- id (PK)
- user_id (FK -> users.id)
- recipe_id (FK -> recipes.id)
- interaction_type (text: 'like','dislike','view','cooked',...)
- source (text: 'discover','plan','home','insights',...)
- created_at (timestamp)

4.2 user_likes

- user_id (FK -> users.id)
- recipe_id (FK -> recipes.id)
- created_at (timestamp)
- PRIMARY KEY (user_id, recipe_id)

4.3 user_reviews

- id (PK)
- user_id (FK -> users.id)
- plan_id (FK -> meal_plans.id, nullable)
- recipe_id (FK -> recipes.id, nullable)
- rating (int, e.g. 1-5)
- comment (text)
- created_at (timestamp)

4.4 user_suggestions

- id (PK)
- user_id (FK -> users.id)

- context (text) -- e.g. 'insights','general'
- message (text, not null)
- created_at (timestamp)

Used by: Discover feedback, likes, Insights reviews and suggestions, training the reco/statistics modules.

5. Facts & Optional Summaries

5.1 facts

- id (PK)
- text (text, not null)
- placement (text, not null) -- 'home','shop','insights'
- active (boolean, default true)
- created_at (timestamp)

Used by: /home, /shop, /insights/weekly for 'Scranly' style facts.

5.2 (Optional) weekly_summaries

- user_id (FK -> users.id)
- week_start (date)
- total_cook_time_min (int)
- est_spend (numeric)
- top_cuisines (text[])
- created_at (timestamp)
- PRIMARY KEY (user_id, week_start)

This table can be used later to cache computed insights if querying raw data becomes expensive.

6. Logical Modules (Code-Level, Not Tables)

These are logical service modules in the codebase that operate on the tables above:

- Reco module:
 - * getDailyBites(userId)
 - * getDiscoverCarousel(userId)
 - * generatePlan(userId, weekStart)
 - * replaceSingleMeal(userId, planId, mealId)
- Amalgamator module:
 - * buildShoppingListForPlan(userId, planId)

Uses: meal_plans, plan_meals, recipe_ingredients, ingredients.
- Statistics module:
 - * getHomeStats(userId)
 - * getWeeklyInsights(userId, weekStart)

Uses: meal_plans, plan_meals, recipes, recipe_ingredients, ingredients, user_interactions, user_reviews (and weekly_summaries if present).