

# 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, planMealId)

### - 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).