# 中医养生类小程序数据库设计文档（整理版）

# 作者：hjt

## 一、文档基础信息

|  |  |
| --- | --- |
| 项目 | 说明 |
| 文档名称 | 中医养生类小程序数据库设计文档 |
| 文档版本 | V1.0 |
| 编写日期 | 2025 年 8 月 |
| 适用范围 | 小程序后端开发、数据库运维 |
| 技术选型 | MySQL 8.0（主数据库）、Redis（缓存辅助） |

## 二、设计概述

本数据库核心用于存储中医养生类小程序的**核心业务数据**，支撑小程序基础功能与个性化服务，具体包括：

1. 存储用户基础信息与健康偏好（如体质标签、饮食禁忌）；
2. 存储养生内容数据（图文 / 视频形式，如药膳做法、养生知识）；
3. 记录用户与内容的互动行为（点赞、收藏、评论）；
4. 支撑个性化推荐功能（如基于用户偏好推荐药膳、中药奶茶）。

设计遵循**关系型数据库规范**，通过外键建立表间关联确保数据完整性；同时结合业务需求优化字段设计（如多标签用逗号分隔），兼顾存储效率与查询便捷性。

## 三、数据表详细设计

### 3.1 用户表（t\_user）

**作用**：存储小程序用户基础信息与健康偏好，以微信 OpenID 作为用户唯一标识，支持微信一键登录绑定。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 描述 | 约束条件 |
| user\_id | BIGINT | 用户 ID（自增） | 主键（PRIMARY KEY） |
| openid | VARCHAR(64) | 微信用户唯一标识（小程序端获取） | 唯一键（UNIQUE KEY）、非空 |
| phone | VARCHAR(20) | 用户手机号（可选绑定） | 唯一键（UNIQUE KEY）、可空 |
| nickname | VARCHAR(50) | 用户昵称（默认微信昵称） | 非空 |
| avatar\_url | VARCHAR(255) | 用户头像 URL（默认微信头像） | 可空 |
| physique\_tags | VARCHAR(255) | 体质标签（如 “平和质，痰湿质”） | 可空、多标签用逗号分隔 |
| dietary\_restrictions | VARCHAR(255) | 饮食禁忌（如 “无羊肉，低盐”） | 可空、多禁忌用逗号分隔 |
| taste\_preferences | VARCHAR(255) | 口味偏好（如 “辛辣，甜”） | 可空、多偏好用逗号分隔 |
| created\_time | DATETIME | 用户账号创建时间 | 非空、默认当前时间（CURRENT\_TIMESTAMP） |
| updated\_time | DATETIME | 用户信息最后更新时间 | 非空、更新时自动刷新（ON UPDATE CURRENT\_TIMESTAMP） |

### 3.2 内容表（t\_content）

**作用**：存储养生类图文 / 视频内容，关联内容作者（用户或管理员）。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 描述 | 约束条件 |
| content\_id | BIGINT | 内容 ID（自增） | 主键（PRIMARY KEY） |
| title | VARCHAR(100) | 内容标题（如 “四神汤的做法”） | 非空 |
| description | TEXT | 内容简介（列表展示用） | 可空 |
| content\_type | VARCHAR(20) | 内容类型 | 非空、枚举值："article"（图文）、"video"（视频） |
| image\_url | VARCHAR(255) | 图文封面图 URL（视频内容可空） | 可空 |
| video\_url | VARCHAR(255) | 视频内容 URL（图文内容可空） | 可空 |
| author\_id | BIGINT | 内容作者 ID（关联用户表） | 外键（FOREIGN KEY）、非空；关联t\_user.user\_id，支持级联查询 |
| created\_time | DATETIME | 内容发布时间 | 非空、默认当前时间 |
| updated\_time | DATETIME | 内容最后更新时间 | 非空、更新时自动刷新 |

### 3.3 内容点赞表（t\_content\_like）

**作用**：记录用户对内容的点赞行为，通过联合唯一键确保 “一人一内容仅一次点赞”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 描述 | 约束条件 |
| like\_id | BIGINT | 点赞记录 ID（自增） | 主键（PRIMARY KEY） |
| user\_id | BIGINT | 点赞用户 ID | 外键（FOREIGN KEY）、非空；关联t\_user.user\_id，用户删除时**级联删除**点赞记录 |
| content\_id | BIGINT | 被点赞内容 ID | 外键（FOREIGN KEY）、非空；关联t\_content.content\_id，内容删除时**级联删除**点赞记录 |
| created\_time | DATETIME | 点赞时间 | 非空、默认当前时间 |
| 联合约束 | - | 防止重复点赞 | UNIQUE KEY (user\_id, content\_id) |

### 3.4 内容收藏表（t\_content\_favorite）

**作用**：记录用户对内容的收藏行为，逻辑与点赞表一致，确保 “一人一内容仅一次收藏”。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 描述 | 约束条件 |
| fav\_id | BIGINT | 收藏记录 ID（自增） | 主键（PRIMARY KEY） |
| user\_id | BIGINT | 收藏用户 ID | 外键（FOREIGN KEY）、非空；关联t\_user.user\_id，用户删除时**级联删除**收藏记录 |
| content\_id | BIGINT | 被收藏内容 ID | 外键（FOREIGN KEY）、非空；关联t\_content.content\_id，内容删除时**级联删除**收藏记录 |
| created\_time | DATETIME | 收藏时间 | 非空、默认当前时间 |
| 联合约束 | - | 防止重复收藏 | UNIQUE KEY (user\_id, content\_id) |

### 3.5 内容评论表（t\_content\_comment）

**作用**：记录用户对内容的评论，支持分页查看单条内容的所有评论。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 描述 | 约束条件 |
| comment\_id | BIGINT | 评论 ID（自增） | 主键（PRIMARY KEY） |
| content\_id | BIGINT | 被评论内容 ID | 外键（FOREIGN KEY）、非空；关联t\_content.content\_id，内容删除时**级联删除**评论 |
| user\_id | BIGINT | 评论用户 ID | 外键（FOREIGN KEY）、非空；关联t\_user.user\_id，用户删除时**级联删除**评论 |
| comment\_text | TEXT | 评论内容 | 非空、长度限制：1-2000 字符 |
| created\_time | DATETIME | 评论发布时间 | 非空、默认当前时间 |

## 四、实体关系（ER 图文字描述）

### 4.1 核心实体

系统包含 5 个核心实体：用户（t\_user）、内容（t\_content）、点赞（t\_content\_like）、收藏（t\_content\_favorite）、评论（t\_content\_comment）。

### 4.2 实体间关系

|  |  |  |
| --- | --- | --- |
| 实体组合 | 关系类型 | 关联逻辑与字段 |
| 用户（t\_user）- 内容（t\_content） | 一对多 | 1 个用户可发布多条内容，1 条内容仅归属 1 个用户；通过t\_content.author\_id关联t\_user.user\_id |
| 用户（t\_user）- 点赞（t\_content\_like） | 一对多 | 1 个用户可点赞多条内容，1 条点赞记录仅归属 1 个用户；通过t\_content\_like.user\_id关联t\_user.user\_id |
| 内容（t\_content）- 点赞（t\_content\_like） | 一对多 | 1 条内容可被多个用户点赞，1 条点赞记录仅对应 1 条内容；通过t\_content\_like.content\_id关联t\_content.content\_id；**间接实现 “用户 - 内容” 多对多关系** |
| 用户（t\_user）- 收藏（t\_content\_favorite） | 一对多 | 逻辑与 “用户 - 点赞” 一致；通过t\_content\_favorite.user\_id关联t\_user.user\_id；**间接实现 “用户 - 内容” 多对多关系** |
| 内容（t\_content）- 收藏（t\_content\_favorite） | 一对多 | 逻辑与 “内容 - 点赞” 一致；通过t\_content\_favorite.content\_id关联t\_content.content\_id |
| 用户（t\_user）- 评论（t\_content\_comment） | 一对多 | 1 个用户可发布多条评论，1 条评论仅归属 1 个用户；通过t\_content\_comment.user\_id关联t\_user.user\_id |
| 内容（t\_content）- 评论（t\_content\_comment） | 一对多 | 1 条内容可被多个用户评论，1 条评论仅对应 1 条内容；通过t\_content\_comment.content\_id关联t\_content.content\_id |

## 五、设计说明

### 5.1 字段格式约定

1. 数据库字段采用**下划线命名法**（如physique\_tags），接口返回时转为**驼峰命名法**（如physiqueTags），适配前后端协作习惯；
2. 多标签 / 多偏好字段（如physique\_tags、dietary\_restrictions）用**逗号分隔**存储，兼顾查询便捷性（如通过FIND\_IN\_SET查询特定标签用户）与存储效率。

### 5.2 性能优化

1. 高频查询字段（如内容的点赞数、收藏数、评论数）建议通过**Redis 缓存**存储，而非实时查询中间表（t\_content\_like/t\_content\_favorite），减少 MySQL 数据库压力；
2. 数据更新规则：更新点赞 / 收藏 / 评论数时，同步更新 Redis 缓存与 MySQL 数据库，确保数据一致性。

### 5.3 数据安全

1. 敏感信息加密：用户openid、phone等敏感数据需通过**AES 加密**存储，防止数据泄露；
2. 接口脱敏返回：敏感信息返回前端时按需脱敏（如手机号显示为 “138\*\*\*\*5678”），避免明文暴露。