

数据库设计说明书（GB8567——88）

引言

编写目的

说明本说明书用于描述 AwesomeCS 系统所用数据库（awesome_cs）的结构、约定、使用与维护要求，供开发、测试、运维与维护人员参考。预期读者包括后端开发人员、DBA、测试人员和运维人员。

背景

1. 待开发数据库名称：awesome_cs。
2. 使用该数据库的软件系统名称：AwesomeCS（包含博客、点赞、评论、付费咨询、模拟面试、学习路径等模块）。
3. 项目提出者、用户与部署环境：项目需求方、系统最终用户及将安装该软件和数据库的计算站（开发/测试/生产环境），具体部署由运维团队负责。

定义

- JSON 字段：数据库中以 JSON 类型或文本形式存储的复合数据结构（如 tags、domains、user_data）。

参考资料

- 项目需求文档
- MySQL 官方手册（建议 MySQL 5.7+ 以支持 JSON）
- 团队编码与数据库设计规范

外部设计

标识符和状态

- 数据库标识：awesome_cs。
- 部署状态：可在开发、测试、生产环境分别部署，需在文档中注明有效时间范围与版本。

使用它的程序

- AwesomeCS 后端服务（Spring Boot，按项目中声明的版本）
- WebSocket 服务模块（用于实时聊天）
- 后台任务、定时任务及管理后台

约定

- 表与字段使用下划线风格（snake_case）；后端实体使用驼峰风格。
- JSON 字段应存储标准 JSON，写入时保持小写与去重（视业务需要）。
- 日期时间统一使用 DATETIME/TIMESTAMP，API 层格式化输出。

专门指导

- 数据导入格式：确保文本与 JSON 字段的格式正确；字符集使用 utf8mb4。
- 数据库操作规范：通过后端 Service 层或专用脚本执行批量导入/更新，生产环境操作需经过变更控制流程。

支持软件

- 建议 DBMS：MySQL 5.7+ 或 MariaDB 支持 JSON 的版本。
- 存储引擎：InnoDB。
- 字符集：utf8mb4。

结构设计

概念结构设计

本数据库反映的现实世界实体与关系包括：

- 用户 (users)
- 博客文章 (posts) 及标签、分类
- 文章点赞 (post_likes)
- 文章评论 (comments) 及回复
- 咨询关系 (consultation_relation) — 专家发布的咨询项
- 咨询会话 (consultations) — 具体的咨询订单/会话
- 咨询支付流水 (consultation_payments)
- 咨询消息 (consultation_messages) — 可选的聊天消息持久化
- 模拟面试记录 (mock_interviews)
- 学习路径推荐 (study_path_recommendations)

对于每一实体，字段类型、约束与说明见下层数据字典条目。

逻辑结构设计

- 关键字段与关系：
 - `posts.user_id` → `users.id`
 - `post_likes.post_id` → `posts.id`
 - `comments.post_id` → `posts.id`, `comments.parent_id` → `comments.id`
 - `consultation_relation.user_id` → `users.id`
 - `consultations.expert_id` / `consultations.seeker_id` → `users.id`
 - `consultation_payments.consultation_id` → `consultations.id`
 - `consultation_messages.consultation_id` → `consultations.id`
- 视图与 API：后端通过 DTO/响应类控制对外数据结构，数据库作为持久化层。

物理结构设计

- 内存与缓冲：数据库参数 (innodb_buffer_pool_size 等) 应根据数据量与并发调整。
- 外存组织：使用 InnoDB 页与事务日志，考虑分表/分库策略用于水平扩展 (未来需求)。
- 访问方式：主从复制用于读写分离；关键写操作 (支付、点赞) 通过事务或幂等设计保证一致性。

运用设计

数据字典设计（概要）

以下为主要表的字段说明：

- users
 - id BIGINT PK （用户 id 主键）
 - phone VARCHAR(20) UNIQUE （用户手机号，唯一）
 - nickname VARCHAR(50) UNIQUE （用户昵称，唯一）
 - password VARCHAR(1024) （用户密码，使用 bcrypt/argon2 等不可逆加密存储）
 - avatar VARCHAR(255) （用户头像 URL）
 - bio VARCHAR(500) （用户个人简介）
 - user_data JSON （用户相关复合信息，如掌握的技术栈，JSON 格式）
 - status TINYINT DEFAULT 1 （用户状态，1-正常，0-禁用）
 - created_at DATETIME, updated_at DATETIME （创建与更新时间）
- posts
 - id BIGINT PK （博客 id 主键）
 - user_id BIGINT （发布用户 id，外键 -> users.id）
 - category VARCHAR(64) （文章分类）
 - tags JSON （标签数组，JSON 格式）
 - title VARCHAR(200) （文章标题）
 - content LONGTEXT （文章内容，可能为 Markdown/HTML）
 - summary VARCHAR(500) （文章摘要）
 - status TINYINT DEFAULT 0 （文章状态，0-草稿，1-已发布，2-归档）
 - view_count INT DEFAULT 0 （浏览量）
 - created_at DATETIME, updated_at DATETIME （创建/更新时间）
- post_likes
 - id BIGINT PK （点赞记录 id）
 - user_id BIGINT （点赞用户 id，外键 -> users.id）
 - post_id BIGINT （被点赞文章 id，外键 -> posts.id）
 - created_at DATETIME （点赞时间）
 - 索引：(post_id)、(user_id)
- comments
 - id BIGINT PK （评论 id）
 - post_id BIGINT （所属文章 id，外键 -> posts.id）
 - user_id BIGINT （评论用户 id，外键 -> users.id）
 - parent_id BIGINT NULL （父评论 id，若为 NULL 则为顶级评论）
 - content TEXT （评论内容）
 - created_at DATETIME （评论时间）
- consultation_relation
 - id BIGINT PK （咨询服务项 id）
 - user_id BIGINT NOT NULL （发布者/专家 id，外键 -> users.id）

- price DECIMAL(10,2) NOT NULL (单次咨询价格)
- domains VARCHAR(1024) 或 JSON (服务领域或技能标签)
- description VARCHAR(1000) NULL (服务描述/说明, 可选)
- created_at DATETIME (创建时间)
- consultations
 - id BIGINT PK (咨询会话/订单 id)
 - expert_id BIGINT (专家用户 id, 外键 -> users.id)
 - seeker_id BIGINT (求助者/下单用户 id, 外键 -> users.id)
 - status TINYINT DEFAULT 0 (会话状态, 例: 0-pending, 1-active, 2-completed, 3-cancelled)
 - scheduled_at DATETIME NULL (预约/开始时间, 可选)
 - created_at DATETIME (创建时间)
 - 索引: expert_id、seeker_id、status
- consultation_payments
 - id BIGINT PK AUTO_INCREMENT (支付记录 id)
 - consultation_id BIGINT (关联咨询会话 id, 外键 -> consultations.id)
 - amount DECIMAL(10,2) (支付金额)
 - status TINYINT DEFAULT 0 (支付状态: 0-pending, 1-success, 2-failed)
 - provider VARCHAR(64) (支付提供商, 如 alipay/wechat/stripe)
 - transaction_id VARCHAR(128) (第三方交易号)
 - created_at DATETIME (支付时间/记录创建时间)
 - 唯一约束: UNIQUE(provider, transaction_id) (保证回调幂等)
- consultation_messages
 - id BIGINT PK AUTO_INCREMENT (消息 id)
 - consultation_id BIGINT (所属咨询会话 id, 外键 -> consultations.id)
 - sender_id BIGINT (发送者 id, 外键 -> users.id)
 - content TEXT (消息内容)
 - message_type VARCHAR(32) DEFAULT 'text' (消息类型, 如 text/image/system 等)
 - created_at DATETIME (消息时间)
 - 索引: consultation_id
- mock_interviews
 - id BIGINT PK (模拟面试记录 id)
 - user_id BIGINT (参加用户 id, 外键 -> users.id)
 - domain VARCHAR(32) (面试领域/方向)
 - style VARCHAR(32) (面试风格, 如 behavior/technical)
 - recording_url VARCHAR(1024) (录音/视频存储地址)
 - score INT NULL (面试评分, 可选)
 - created_at DATETIME (创建时间)
- study_path_recommendations
 - id BIGINT PK (推荐记录 id)

- user_id BIGINT （关联用户 id，外键 -> users.id）
- content TEXT （推荐内容）
- created_at DATETIME （创建时间）

安全保密设计

- 访问控制：按角色分配数据库账号权限，原则最小权限。
- 敏感数据：密码使用不可逆加密存储（bcrypt/argon2）；生产环境启用 TLS 连接。
- 审计日志：记录支付回调、订单状态变化等关键日志以便追溯。

物理设计与索引建议

- 引擎：InnoDB
- 字符集：utf8mb4
- 索引：
 - posts(user_id), posts(status), posts(created_at)
 - post_likes(post_id), post_likes(user_id)
 - comments(post_id), comments(parent_id)
 - consultation_payments(consultation_id)
 - consultation_messages(consultation_id)
- tags/domains 查询建议：
 - 使用 JSON 类型并配合 JSON_CONTAINS 查询，或在写入时同步生成标准化字段并为其建立索引

事务与并发控制

- 点赞：使用事务保证插入/删除与计数一致；高并发下使用 Redis 缓存计数并定时落盘。
- 支付流程：支付回调必须幂等处理，使用 consultation_payments 的 (provider, transaction_id) 唯一约束保证幂等性，并在事务中更新 consultations 状态。
- view_count：高并发场景下使用 Redis 累积并周期性同步到数据库。

运维与专门指导

- 备份策略：定期全量备份 + binlog 增量；灾备演练。
- 数据迁移：使用有序的 SQL 迁移脚本（Flyway 或 Liquibase）。
- 日志与监控：监控慢查询、连接数、IO、buffer pool 使用率；设置慢查询阈值并优化。

安全与保密（补充）

- 加密：对敏感字段进行加密存储或在应用层加密。
- 访问日志：对支付、提现、数据导出等敏感操作记录操作人、时间与变更内容。