小智智能生活助理平台

数据库设计说明书

小智智能生活助理平台 • 数据库设计（GB/T 8567-2006 风格）

文档名称	智行小组_数据库设计说明书
项目名称	小智智能生活助理平台
数据库系统	MySQL 8.0（库名 xzsl_db）
团队名称	智行小组
文档版本	V1.0
编制日期	2026-11-27

文档修订记录

版本	修订日期	修订人	修订说明
V1.0	2026-11-27	本人（智行小组）	完成数据库设计初稿

目 录

文档修订记录 2

1 引言 4

1.1 编写目的 4

1.2 背景 4

1.3 术语与定义 4

1.4 参考资料 4

2 外部设计 4

2.1 标识符和状态 4

2.2 使用它的程序 5

2.3 约定 5

2.4 专门指导 5

2.5 支持软件 5

3 结构设计 5

3.1 概念结构设计 5

3.2 逻辑结构设计 6

3.3 物理结构设计 10

3.4 数据字典摘要 10

4 对象关系映射设计 11

5 安全保密设计 11

6 运行与维护设计 12

1 引言

1.1 编写目的

本文档给出小智智能生活助理平台数据库的完整设计，包括外部设计、概念结构设计（ER 模型）、逻辑结构设计（关系模式）、物理结构设计（存储与索引）、对象关系映射（ORM）以及安全与维护设计，作为数据库实施、编码与运维的依据。

1.2 背景

本系统是具备“能说会做”能力的 AI 智能体平台：用户与智能体对话，智能体调用天气、待办、笔记、出行等工具完成实际任务；管理员在后台维护用户、工具配置并审计日志。系统需要持久化用户、会话、消息、待办、笔记、出行计划、工具配置及各类日志数据，故选用 MySQL 8.0 关系数据库。

1.3 术语与定义

缩写 / 术语	说明
PK	主键（Primary Key）
FK	外键（Foreign Key）
UQ	唯一约束（Unique）
NN	非空约束（Not Null）
JSON	JSON 类型列，存储结构化半结构化数据
逻辑删除	以状态位标记删除而非物理删除，保留审计线索
ORM	对象关系映射，本系统使用 SQLAlchemy 2.0

1.4 参考资料

- GB/T 8567-2006《计算机软件文档编制规范》
- 《智行小组_系统设计说明书》
- MySQL 8.0 Reference Manual
- SQLAlchemy 2.0 官方文档

2 外部设计

2.1 标识符和状态

项目	约定
数据库名称	xzsl_db
字符集 / 排序规则	utf8mb4 / utf8mb4_0900_ai_ci
存储引擎	InnoDB（支持事务、行级锁、外键）
时区	服务器统一 UTC 存储，展示层转 Asia/Shanghai

项目	约定
版本	MySQL 8.0.36

2.2 使用它的程序

模块	访问的表	访问方式
认证服务 AuthService	sys_user、sys_admin	读写
对话服务 ChatService	ai_conversation、ai_message、tool_call_log、tool_config	读写
待办服务 TodoService	todo_item	读写
笔记服务 NoteService	user_note	读写
出行服务 TravelService	travel_plan	读写
管理后台	sys_user、sys_admin、tool_config、tool_call_log、sys_operation_log	读写

2.3 约定

- 表名：小写蛇形命名（snake_case），模块前缀如 sys_（系统）、ai_（智能体）、todo_（待办）；
- 字段名：小写蛇形命名；主键统一为“表名单数 + _id”；时间戳统一为 created_at / updated_at（DATETIME）；
- 布尔型用 TINYINT（0/1）表示；状态字段用 TINYINT + 注释说明取值；
- 删除策略：业务表采用逻辑删除（status 字段），日志表只增不删；
- 所有外键均建立索引；大文本列（content）不建索引。

2.4 专门指导

数据库实行每日全量备份 + binlog 增量备份；备份文件异地保存；每月进行一次恢复演练。

2.5 支持软件

数据库服务器软件 MySQL Server 8.0；客户端工具 MySQL Workbench / DBeaver；应用侧通过 SQLAlchemy 2.0 + PyMySQL 驱动访问。

3 结构设计

3.1 概念结构设计

概念模型包含 10 个实体：用户、管理员、会话、消息、工具配置、工具调用日志、待办、笔记、出行计划、操作日志。主要联系：一个用户发起多个会话；一个会话包含多条消息；一条消息可能产生多条工具调用日志；一个工具配置被多次调用；管理员维护工具配置并产生操作日志。

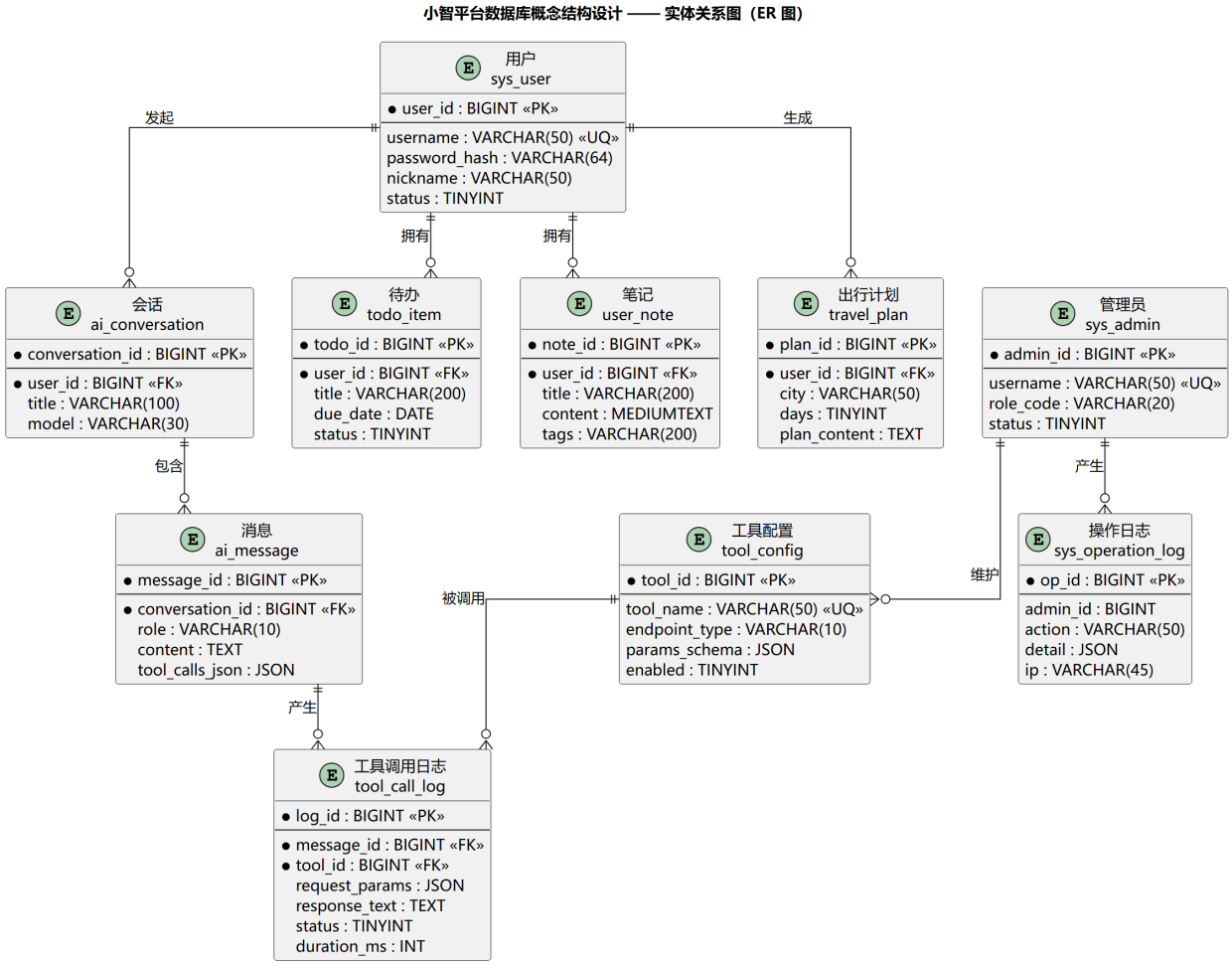


图 3-1 数据库概念结构设计（ER 图）

实体	对应表	核心属性	主要联系
用户	sys_user	username、password_hash、status	1:N 会话 / 待办 / 笔记 / 出行计划
管理员	sys_admin	username、role_code、status	1:N 操作日志；1:N 维护工具配置
会话	ai_conversation	title、model、status	N:1 用户；1:N 消息
消息	ai_message	role、content、tool_calls_json	N:1 会话；1:N 工具调用日志
工具配置	tool_config	tool_name、params_schema、enabled	1:N 工具调用日志
工具调用日志	tool_call_log	request_params、status、duration_ms	N:1 消息 / 工具配置
待办	todo_item	title、due_date、status	N:1 用户
笔记	user_note	title、content、tags	N:1 用户
出行计划	travel_plan	city、days、plan_content	N:1 用户
操作日志	sys_operation_log	action、detail、ip	N:1 管理员

3.2 逻辑结构设计

（1）用户表 sys_user

字段名	类型	约束 / 说明
user_id	BIGINT	PK，自增主键
username	VARCHAR(50)	NN，UQ，登录账号
password_hash	VARCHAR(64)	NN，bcrypt 哈希
nickname	VARCHAR(50)	昵称，默认取 username
avatar_url	VARCHAR(255)	头像地址
email	VARCHAR(100)	邮箱，UQ
phone	VARCHAR(20)	手机号
status	TINYINT	NN，1 正常 / 0 禁用 / 2 未激活，默认 1
created_at	DATETIME	NN，注册时间
updated_at	DATETIME	NN，更新时间

（2）管理员表 sys_admin

字段名	类型	约束 / 说明
admin_id	BIGINT	PK，自增主键
username	VARCHAR(50)	NN，UQ，管理员账号
password_hash	VARCHAR(64)	NN，bcrypt 哈希
real_name	VARCHAR(50)	真实姓名
role_code	VARCHAR(20)	NN，SUPER_ADMIN / OPERATOR
status	TINYINT	NN，1 正常 / 0 禁用，默认 1
last_login_at	DATETIME	最近登录时间
created_at	DATETIME	NN，创建时间

（3）会话表 ai_conversation

字段名	类型	约束 / 说明
conversation_id	BIGINT	PK，自增主键
user_id	BIGINT	NN，FK → sys_user.user_id
title	VARCHAR(100)	会话标题，默认取首条消息前 30 字
model	VARCHAR(30)	使用的模型名，如 gpt-4o-mini
status	TINYINT	NN，1 进行中 / 2 已结束 / 0 已删除
created_at	DATETIME	NN，创建时间
updated_at	DATETIME	NN，最后活跃时间

（4）消息表 ai_message

字段名	类型	约束 / 说明
message_id	BIGINT	PK，自增主键
conversation_id	BIGINT	NN，FK → ai_conversation.conversation_id
role	VARCHAR(10)	NN，user / assistant / tool
content	MEDIUMTEXT	消息文本
tool_calls_json	JSON	模型返回的工具调用指令（assistant 消息）
token_usage	INT	本消息消耗的 token 数
created_at	DATETIME	NN，创建时间

（5）工具配置表 tool_config

字段名	类型	约束 / 说明
tool_id	BIGINT	PK，自增主键
tool_name	VARCHAR(50)	NN，UQ，工具名（get_weather 等）
display_name	VARCHAR(100)	展示名
description	VARCHAR(500)	工具描述（供模型理解）
endpoint_type	VARCHAR(10)	NN，builtin（内置）/ mcp（MCP 服务）
mcp_command	VARCHAR(255)	MCP 启动命令，如 python mcp_server.py
mcp_args	JSON	MCP 启动参数数组
params_schema	JSON	参数 JSON Schema
enabled	TINYINT	NN，1 启用 / 0 停用，默认 1
created_at	DATETIME	NN
updated_at	DATETIME	NN

（6）工具调用日志表 tool_call_log

字段名	类型	约束 / 说明
log_id	BIGINT	PK，自增主键
message_id	BIGINT	NN，FK → ai_message.message_id
conversation_id	BIGINT	会话冗余字段，便于按会话检索
user_id	BIGINT	用户冗余字段，便于按用户统计
tool_name	VARCHAR(50)	NN，调用的工具名
request_params	JSON	调用参数
response_text	TEXT	返回结果（截断至 4KB）
status	TINYINT	NN，1 成功 / 0 失败
duration_ms	INT	耗时（毫秒）
created_at	DATETIME	NN，调用时间

（7）待办表 todo_item

字段名	类型	约束 / 说明
todo_id	BIGINT	PK，自增主键
user_id	BIGINT	NN，FK → sys_user.user_id
title	VARCHAR(200)	NN，任务内容
due_date	DATE	截止日期
priority	TINYINT	NN，1 高 / 2 中 / 3 低，默认 2
status	TINYINT	NN，0 未完成 / 1 已完成，默认 0
source	VARCHAR(10)	NN，manual（手动）/ agent（AI 创建）
created_at	DATETIME	NN
finished_at	DATETIME	完成时间

（8）笔记表 user_note

字段名	类型	约束 / 说明
note_id	BIGINT	PK，自增主键
user_id	BIGINT	NN，FK → sys_user.user_id
title	VARCHAR(200)	NN，笔记标题
content	MEDIUMTEXT	Markdown 正文
tags	VARCHAR(200)	标签，逗号分隔
source	VARCHAR(10)	NN，manual / agent
created_at	DATETIME	NN
updated_at	DATETIME	NN

（9）出行计划表 travel_plan

字段名	类型	约束 / 说明
plan_id	BIGINT	PK，自增主键
user_id	BIGINT	NN，FK → sys_user.user_id
city	VARCHAR(50)	NN，目的地城市
start_date	DATE	出发日期
days	TINYINT	NN，游玩天数 1-3
weather_snapshot	JSON	生成计划时的天气快照
plan_content	MEDIUMTEXT	计划正文
status	TINYINT	NN，1 有效 / 0 已过期
created_at	DATETIME	NN

（10）操作日志表 sys_operation_log

字段名	类型	约束 / 说明
op_id	BIGINT	PK，自增主键
admin_id	BIGINT	FK → sys_admin.admin_id（系统操作为空）
operator_type	VARCHAR(10)	NN，admin / user / system
target_type	VARCHAR(30)	操作对象类型，如 user、tool_config
target_id	BIGINT	操作对象 ID
action	VARCHAR(50)	NN，动作，如 DISABLE_USER
detail	JSON	操作详情
ip	VARCHAR(45)	来源 IP
created_at	DATETIME	NN，操作时间

3.3 物理结构设计

存储参数：全部表使用 InnoDB；表字符集 utf8mb4；主键采用自增 BIGINT；日志类大表预留按月分区能力（当前按月归档）。索引设计如下：

表	索引名	索引列	用途
sys_user	uk_username	username	唯一索引，登录查询
sys_user	idx_email	email	邮箱找回密码
ai_conversation	idx_user_time	user_id, updated_at	会话列表按时间倒序
ai_message	idx_conv	conversation_id	会话消息检索
tool_config	uk_tool_name	tool_name	工具唯一性
tool_call_log	idx_user_time	user_id, created_at	用户维度统计
tool_call_log	idx_tool_time	tool_name, created_at	工具维度统计
todo_item	idx_user_status	user_id, status	待办列表筛选
user_note	idx_user	user_id	笔记列表
travel_plan	idx_user	user_id	计划列表
sys_operation_log	idx_time	created_at	日志按时间查询

容量估算：按 10 万注册用户、人均 50 条消息、日均 1 万次工具调用测算，核心表规模：sys_user 约 10 万行（约 50MB），ai_message 约 500 万行（约 4GB），tool_call_log 约 300 万行（约 2GB），其余表合计小于 2GB；总容量约 10GB 量级，单机即可承载，预留分区与归档策略应对增长。

3.4 数据字典摘要

表名	中文名	记录数估算	说明
sys_user	用户表	10 万	系统用户
sys_admin	管理员表	< 100	后台管理员

表名	中文名	记录数估算	说明
ai_conversation	会话表	100 万	AI 会话
ai_message	消息表	500 万	会话消息
tool_config	工具配置表	< 100	内置/MCP 工具
tool_call_log	工具调用日志表	300 万	调用审计
todo_item	待办表	100 万	用户待办
user_note	笔记表	50 万	用户笔记
travel_plan	出行计划表	50 万	出行计划
sys_operation_log	操作日志表	100 万	管理操作审计

4 对象关系映射设计

应用侧使用 SQLAlchemy 2.0 Declarative 模型，实体类与数据表一一映射；类属性采用与表字段同名的小写蛇形命名，通过 __tablename__ 显式绑定表名。

实体类（模块）	数据表	映射关系说明
User（domain/user.py）	sys_user	一对一映射，status 对应枚举 UserStatus
Admin（domain/admin.py）	sys_admin	一对一映射，role_code 对应角色枚举
Conversation（domain/chat.py）	ai_conversation	一对一映射，user_id 关联 User
Message（domain/chat.py）	ai_message	一对一映射，conversation_id 关联 Conversation
ToolConfig（domain/tool.py）	tool_config	一对一映射，params_schema 序列化为 JSON 字符串
ToolCallLog（domain/tool.py）	tool_call_log	一对一映射，request_params 存 JSON
TodoItem（domain/todo.py）	todo_item	一对一映射，status 对应 TodoStatus 枚举
Note（domain/note.py）	user_note	一对一映射
TravelPlan（domain/travel.py）	travel_plan	一对一映射，weather_snapshot 存 JSON
OperationLog（domain/log.py）	sys_operation_log	一对一映射，detail 存 JSON

ORM 约定：所有查询使用参数化语句防 SQL 注入；JSON 字段在 ORM 层使用 JSON 类型自动序列化；时间字段统一由 ORM 维护（default=now / onupdate=now）；级联删除不物理删除，而是由服务层更新状态位。

5 安全保密设计

- 账号密码仅存 bcrypt 哈希，不存明文；登录态由 JWT + Redis 控制，不落库；
- 数据库账号按最小权限分配：应用账号仅拥有 xzsl_db 的增删改查权限，无 DDL 权限；
- 敏感字段（如请求参数中可能出现的密钥）在日志中脱敏后再入库；

- 管理端操作全部写入 sys_operation_log，支持审计追溯；
- 数据库端口仅对应用服务器开放（安全组 / 防火墙白名单）。

6 运行与维护设计

事项	策略
备份	每日 02:00 全量备份（mysqldump）+ binlog 增量，保留 30 天
恢复	RTO ≤ 2 小时；每季度演练一次全量恢复
日志清理	tool_call_log 超过 180 天归档至历史表后删除
监控	慢查询、连接数、磁盘使用率阈值告警
升级	schema 变更使用迁移脚本（Alembic），支持前向与回滚