

系统数据库设计说明书

版本: 1.0

作者: [爱玛单车队]

日期: 2024-10-28

1. 引言

1.1 编写目的

在完成 Speak with AI 小程序系统需求分析的基础上, 我们依据《系统设计说明书》对该系统进行了设计。为了文档化对该系统的设计, 编写了本文档。

1.2 命名规范

数据库表明、字段名用大写开头的英文命名。

私有变量用“_”开头的小写英文命名, 其他变量用大写英文字母开头。

所有的函数用大写英文字母开头。

1.3 术语定义

总体结构: 软件系统的总体逻辑结构, 本系统采用面向对象的方法对系统进行设计。

概念数据模型(CDM): 关系数据库的概念设计模型, 主要表现为 E-R 图。

物理数据模型(PDM): 关系数据库的物理设计模型。

1.4 参考资料

[1] 吕云翔. 软件工程实用教程. 北京: 清华大学出版社, 2015.

1.5 相关文档

系统设计说明书

1.6 版本更新信息

版本号	创建者	创建日期	维护者	维护日期	维护纪要
0.0	曾庆徽	2024.10.21	爱玛单车队 小组成员	2024.10.21-10.29	设计原型, 创建数据库

2. 总体设计

2.1 硬件运行环境

1. 服务器

CPU: Intel Xeon E5-2680 2.50GHz 以上 内存: 1GB 及以上 网络配置: 100M 网卡及以上

2. Web 浏览 PC

CPU: AMD 或 Inter 1.6GHz 以上 内存: 1GB 及以上 网络配置: 100M 网卡及以上

2.2 软件运行环境

1. 服务器

操作系统: Linux (Kernel 4.0 以上)

数据库: MySQL-8.0.15

Web 服务器: nginx、uWSGI

2. 客户机

操作系统无限制, 有网络, 浏览器即可

2.3 子系统清单

子系统清单图如下

子系统编号	子系统名称	子系统功能描述
1	个人信息管理系统	注册登录功能的实现, 个人信息维护更新, 管理员管理网页用户
2	人机交流系统	进行 AI 语音交流
3	英文翻译系统	对于给出的英语段落进行翻译
4	个人中心系统	用户查看学习记录, 个人信息等

2.4 功能模块清单

功能模块图如下

模块编号	名称	模块功能描述
101	注册账号	用户从游客注册为用户
102	登录系统	已注册用户登录系统
103	找回密码	用户丢失密码后, 通过审核重新获取
104	查看个人信息	用户查看个人信息
105	修改个人信息	会员登录后对资料进行管理
106	用户注销	用户离开系统时, 进行注销
201	语音输入	进行英文语音输入, 从而完成对话
202	翻译	对给出英文段落进行翻译
203	场景选择	给 AI 设计对话场景

模块编号	名称	模块功能描述
204	设置	设置对话难度
205	实例回答	通过 AI 给出适合的答案
301	英语水平	存储用户英文水平
302	学习记录	记录用户学习信息
303	个人资料编辑	设置个人资料
304	反馈情况	总结用户的发音情况
305	系统评分	对用户语音对话进行评分

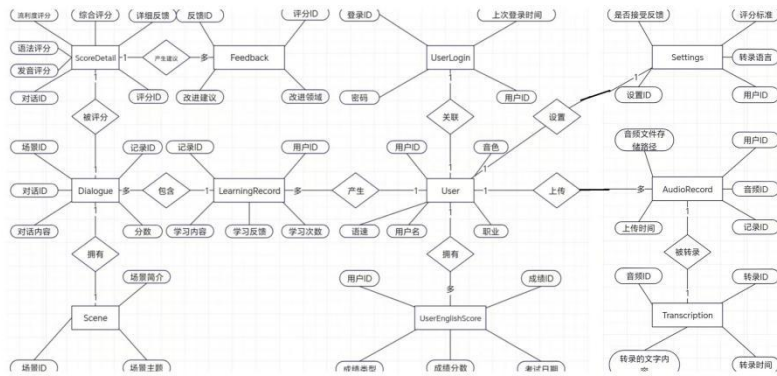
3. 数据库设计

3.1 数据库中表名列表

表名	用途
User	用户表
UserLogin	用户登录表
UserEnglishScore	用户英语成绩表
LearningRecord	学习记录表
Scene	场景表
Dialogue	对话表
AudioRecord	音频记录表
Transcription	转录表
ScoreDetail	评分详情表
Feedback	反馈表
Settings	用户设置表

3.2 数据库中的表关系

该关系可用 E-R 图来呈现（具体主键外键约束参考详细清单）：



3.3 数据库表的详细清单

3.3.1 用户表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	user_id	用户 ID	INT	否	主键
2	username	用户名	VARCHAR(50)	否	
3	profession,	职业	VARCHAR(50)	是	
4	voice	音色	VARCHAR(50)	是	
5	Speech_speed	语速	VARCHAR(50)	是	

3.3.2 用户登录表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	login_id	登录 ID	INT	否	主键
2	user_id	用户 ID	INT	否	外键
3	password	密码	varchar(50)	否	
4	last_login	上次登录时间	TIMESTAMP	是	

3.3.3 用户英语成绩表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	score_id	成绩 ID	INT	否	主键
2	user_id	用户 ID	INT	否	外键

序号	英文字段名	中文字段名	数据类型	是否允许为空	主 键 / 外键
3	score_type	成绩类型	VARCHAR(50)	是	
4	score	成绩分数	INT	是	
5	exam_date	考试日期	DATE	是	

3.3.4 学习记录表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键/外键
1	record_id	记录 ID	INT	否	主键
2	user_id	用户 ID	INT	否	外键
3	study_content	学习内容	TEXT	是	
4	study_feedback	学习反馈	TEXT	是	
5	study_times	学习次数	INT	是	

3.3.5 场景表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主 键 / 外键
1	scene_id	场景 ID	INT	否	主键
2	scene_topic	场景主题	VARCHAR(100)	是	
3	scene_description	场景简介	TEXT	是	

3.3.6 对话表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键/外键
1	dialogue_id	对话 ID	INT	否	主键
2	record_id	记录 ID	INT	否	外键
3	scene_id	场景 ID	INT	否	外键
4	dialogue_content	对话内容	TEXT	是	
5	score	分数	INT	是	

3.3.7 音频记录表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	audio_id	音频 ID	INT	否	主键
2	user_id	用户 ID	INT	否	外键
3	record_id	记录 ID	INT	否	外键
4	audio_path	音频文件存储路径	VARCHAR(255)	是	
5	upload_time	上传时间	TIMESTAMP	是	

3.3.8 转录表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	transcription_id	转录 ID	INT	否	主键
2	audio_id	音频 ID	INT	否	外键
3	transcribed_text	转录的文字内容	TEXT	否	
4	transcription_time	转录时间	VARCHAR(255)	是	

3.3.9 评分详情表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	score_id	评分 ID	INT	否	主键
2	dialogue_id	对话 ID	INT	否	外键
3	pronunciation_score	发音评分	INT	是	
4	grammar_score	语法评分	INT	是	
5	fluency_score	流利度评分	INT	是	
6	overall_score	综合评分	INT	是	
7	feedback	详细反馈	TEXT	是	

3.3.10 反馈表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键 / 外键
1	feedback_id	反馈 ID	INT	否	主键
2	score_id	评分 ID	INT	否	外键
3	suggestion	改进建议	TEXT	是	
4	improvement_area	改进的具体领域	VARCHAR(100)	是	

3.3.11 用户设置表

序号	英文字段名	中文字段名	数据类型	是否允许为空	主键/外键
1	settings_id	设置 ID	INT	否	主键
2	user_id	用户 ID	INT	否	外键
3	transcription_language	转录语言	VARCHAR(50)	是	
4	scoring_standard	评分标准	VARCHAR(50)	是	
5	notification_preference	是否接收反馈通知	BOOLEAN	是	

4. 典型的子系统设计

本节以 AI 辅助英语口语练习中口语交流典型功能子系统为例，从功能子系统的类属性方法设计、功能活动设计、处理时序设计三个层面进行详细设计描述。

4.1 AI 辅助英语口语系统设计

4.1.1 类设计

放在这里显示不清楚，详情见我们的类图.png文件（或者博客园）。

4.1.2 活动图设计

详情见我们的活动图.png文件。

4.1.3 时序图设计

详情叫我们的时序图.png文件。

5. 功能模块设计

1. 登陆页面：

调用背景：

1. 手机号注册以及登陆
2. 微信登录（备选）

页面组成：软件名称以及两个登录方式

调用描述：无

2.主页：

调用背景：通过对话按键与 AI 进行对话，为软件的主要功能

页面组成：实例回答，个人信息，场景选择，翻译，设置，点击说话

3.个人信息页面：

调用背景：存储用户的各类型的信息

页面组成：语言水平，学习记录，个人信息设置

6. ORM 模型设计

我们使用的是 Java ORM 模型开发：

具体模型实例代码框架参考我们小组文件：

ORM.txt。

7. 接口设计

7.1 用户接口

使用 Vue 作为前端框架，结合 Java 后端 提供数据支持，通过 CSS 增强动画效果，提供清晰、简洁、易用且美观的用户界面，并使用 AJAX 实现局部刷新

7.2 外部接口

数据储存：本系统涉及大量数据的存储和处理，通过 JAVA 操纵 MySQL-8.0.15 数据库进行交互

7.3 内部接口

本系统以数据为中心，网站各模块均通过 ORM 模块与数据库服务器进行交互，页面跳转时，通过 session 以及 GET 请求和表单等传递参数

8. 系统错误处理

8.1 出错信息

1. 对用户输入的各项内容进行有效性、安全性检查，减少错误发生的几率；
2. 对程序运行中的异常均进行捕获，按统一的方式将出错提示信息提供给用户；
3. 当游客、会员访问自身权限以外的信息时，将其导航到统一的出错提示页面。

8.2 故障预防与补救

以统一的机制进行网站权限的控制，对程序中用到的数据尽量进行加密，以减少黑客攻

击的危害。定期对数据库中的数据进行海量备份及增量备份。

8.3 系统维护设计

1. 编码实现时应采用模块化和分层的思想，提高模块内部的内聚，减少模块间的耦合。使系统逻辑结构清晰,从而增强可读性和可维护性。
2. 面向数据与面向对象相结合，模块划分符合面向对象思想。
3. 在编码过程中注意标识符命名的意义，添加适量注释。

9. 测试计划

具体的测试计划将围绕单元测试以及实体测试进行。单元测试将使用 `jest` 框架等来进行，实体测试将围绕开发人员对各个环节进行真机模拟来测试。

试用水印