题目：基于大模型的英语水平提升系统

# 目录

[1 项目介绍 5](#_Toc73)

[1.1 背景 5](#_Toc14918)

[1.2 愿景 5](#_Toc20005)

[2 需求分析 7](#_Toc14837)

[2.1 用例图 7](#_Toc31088)

[2.2 用例描述 7](#_Toc1710)

[2.3 概念类图 13](#_Toc16016)

[3 系统设计 14](#_Toc28675)

[3.1 概要设计 14](#_Toc10387)

[3.1.1 系统体系结构设计 14](#_Toc20382)

[3.1.2 数据库表设计 14](#_Toc507)

[3.2 详细设计 17](#_Toc26451)

[4 系统实现 22](#_Toc17096)

[4.1 应用的技术介绍 22](#_Toc31050)

[4.1.1 Android 22](#_Toc23491)

[4.1.2 springboot 24](#_Toc13715)

[4.1.3 MVC模式 26](#_Toc7527)

[4.1.4 MySQL 28](#_Toc1858)

[4.1.5百度千帆大模型（ERNIE-Speed-128K） 29](#_Toc9657)

[4.2 各功能模块的实现 31](#_Toc25205)

[4.2.1 基本功能 31](#_Toc12363)

[4.2.2 后端信息 42](#_Toc2081)

[4.2.3 数据库连接 43](#_Toc29576)

[5 系统测试 45](#_Toc31902)

[5.1 功能测试 45](#_Toc14825)

[5.1.1管理端 45](#_Toc2324)

[5.1.2用户登录/注册 48](#_Toc15894)

[5.1.3设定学习计划 49](#_Toc20975)

[5.1.4 单词学习 50](#_Toc789)

[5.1.5 单词复习 50](#_Toc23041)

[5.1.6 学习数据统计 51](#_Toc17624)

[5.1.7 AI对话与语音合成 51](#_Toc26990)

[5.2 性能测试 52](#_Toc28810)

[5.3 其它测试 54](#_Toc26231)

[5.3.1可用性测试 54](#_Toc13703)

[5.3.2兼容性测试 54](#_Toc24830)

[5.3.3安全性测试 54](#_Toc18833)

[6 项目总结 55](#_Toc28707)

# 1 项目介绍

## 背景

近年来，人工智能技术迅猛发展，尤其是以GPT-4等大语言模型为代表的自然语言处理技术，极大地提升了计算机理解和生成自然语言的能力，其能够生成高质量的文本，进行复杂的语言理解和对话。这为各类语言应用，包括自动翻译、语言教学等，提供了强有力的技术支持。

而随着互联网技术的普及，在线教育迅速崛起，成为传统教育的重要补充，甚至在某些领域成为主要学习方式。在线学习打破了地域和时间的限制，使得教育资源更加广泛地传播和利用。尤其是在新冠疫情的推动下，线上学习的需求急剧增加，促进了各种在线学习平台和工具的快速发展和普及。

在全球化进程不断加快的背景之下，英语作为全球通用语言的地位愈加突出。掌握良好的英语能力不仅是个人发展的重要技能，更是参与国际合作、获取国际资讯、融入全球化社会的必备条件。各国政府和教育机构越来越重视英语教育，推动各类英语水平测试和提升项目的发展，以满足日益增长的国际交流需求。《全日制义务教育普通高级中学英语课程标准》明确规定了各个级别的学生应达到的英语水平，英语作为考试的必考科目，也受到社会各界的重视，英语教育培训成为热门行业。

因此基于大模型的英语水平提升系统拥有巨大的市场需求，随着移动设备发展，英语学习类APP充分利用移动设备的便携性优势和大模型技术发展的技术支持，可以为语言学习者提供更有趣味性的学习平台。

## 1.2 愿景

我们的系统旨在利用大语言模型技术，通过生成个性化英文短文和语音，为用户提供定制化的英语学习体验。系统将记录用户的学习频次和成绩，管理用户的词汇学习，并通过循环复习加强记忆。同时，系统将提供一个数据可视化平台，帮助用户实时追踪自己的学习进度和成效。

主要功能：

1. 文本和语音生成： 系统根据用户定义的标准（如包含特定单词、特定主题等），生成对应的英文短文和语音，帮助用户提高语言实际应用能力。
2. 用户管理： 系统将管理用户的个人信息，如用户的学习数据，包括学习频次和评分。
3. 数据记录与分析： 对用户的学习活动进行详细记录，通过数据分析，优化学习过程，提高学习效率。
4. 学习进度可视化： 通过图表和进度条直观展示用户的学习成果和进度，增强学习动力。

我们的系统将成为用户提升英语能力的首选平台，通过高度个性化和科技驱动的学习方式，让用户的英语学习过程更高效、更有成效。

# 2 需求分析

## 2.1 用例图

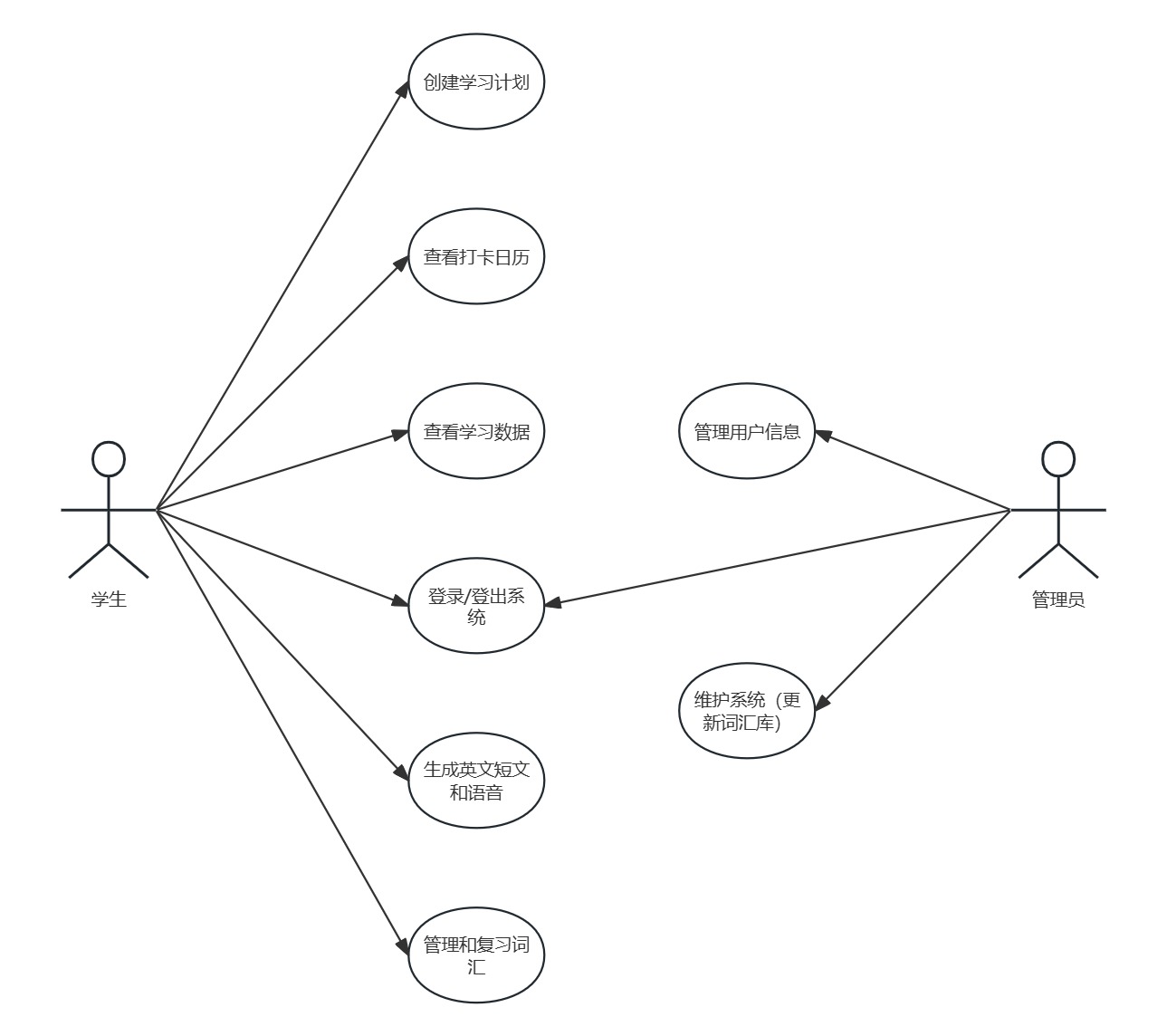
****

图2.1 基于大模型的英语水平提升系统用例图

## 2.2 用例描述

|  |  |
| --- | --- |
| **用例名** | 生成英文短文和语音 |
| **简要描述** | 系统根据用户指定的参数（如包含特定单词，主题等）自动生成一段英文短文及其对应的语音 |
| **参与者** | 用户 |
| **涉众** | 用户：直接受益于生成的内容，用于学习和练习  系统管理员：监控和维护系统性能，确保生成功能正常运行 |
| **相关用例** | 管理和复习词汇：用户可能在生成短文后使用此功能进行复习 |
| **前置条件** | 用户已成功登录系统  用户已输入所有必要的生成参数（如单词限制，主题选择等） |
| **后置条件** | 系统显示生成的英文短文和播放相应语音  用户的活动记录被更新，包括新生成的短文和语音 |
| **基本事件流**   1. 用户登录系统 2. 用户选择生成英文短文和语音的功能 3. 用户输入生成短文的具体参数（例如，包含特定单词，选择主题等） 4. 系统处理输入的参数，生成英文短文 5. 系统使用文本到语音技术生成对应的英文语音 6. 系统展示生成的短文和播放语音 7. 用户可以听取并阅读短文 | |
| **备选事件流**  A-1参数不完整或不符合要求   1. 系统提示参数错误 2. 用户选择重新输入或终止操作   A-2生成过程中发生错误   1. 系统显示技术性错误消息 2. 用户选择重试或终止操作 | |
| **补充约束-数据需求**  D-1所有输入参数必须进行验证，确保它们符合格式和逻辑要求  D-2生成的短文和语音需要存储，以便历史查询和数据分析  **补充约束-业务规则**  B-1生成的短文必须符合教育质量标准，确保其教学适用性 | |
| **待解决问题**  如何处理高峰时段的大量生成请求，确保系统稳定性和响应速度 | |
| **相关图** | |

|  |  |
| --- | --- |
| **用例名** | 创建学习计划 |
| **简要描述** | 用户可以创建个性化的学习计划，包括设定每天学习的单词数量和选择学习内容的主题。 |
| **参与者** | 用户 |
| **涉众** | 用户：直接受益于定制的学习计划，可以系统地学习和复习  系统管理员：监控和维护学习计划创建功能的性能和可用性 |
| **相关用例** | 查看学习数据：用户基于学习数据来调整他们的学习计划。 |
| **前置条件** | 用户必须已经登录系统 |
| **后置条件** | 学习计划被创建并保存在用户的账户中  用户可以开始根据计划进行学习 |
| **基本事件流**   1. 用户登录到系统 2. 用户导航到创建学习计划的界面 3. 用户输入学习计划的详细信息，如每天的学习单词数量和选定的学习主题 4. 系统验证输入的详细信息 5. 系统保存学习计划 6. 系统通知用户学习计划创建成功 7. 用户开始按照计划学习 | |
| **备选事件流**  A-1输入信息不完整或不合逻辑   1. 系统提示错误并要求正确的信息 2. 用户修改信息或取消创建   A-2系统无法保存学习计划   1. 系统显示错误消息 2. 用户选择重试或取消 | |
| **补充约束-数据需求**  D-1学习计划必须包含有效日期和学习目标，以确保其实用性  D-2系统应能处理并存储多个用户的学习计划，保持数据隔离和安全  **补充约束-业务规则** | |
| **待解决问题** | |
| **相关图** | |

|  |  |
| --- | --- |
| **用例名** | 查看打卡日历 |
| **简要描述** | 用户可以查看一个日历界面，显示他们每天的学习活动和完成情况，以便追踪自己的学习进度和连续学习天数 |
| **参与者** | 用户 |
| **涉众** | 用户：直接受益于能够视觉化看到自己的学习连续性和成就，增加学习动机  系统管理员：监控功能的性能，确保数据的准确性和系统的稳定性 |
| **相关用例** | 查看学习数据：打卡数据可以整合到学习数据中，提供更全面的学习进展报告 |
| **前置条件** | 用户必须已经登录系统  用户已经开始一个学习计划，有相关的学习活动记录 |
| **后置条件** | 用户查看他们的打卡日历，并了解自己的学习活动和完成情况 |
| **基本事件流**   1. 用户登录到系统 2. 用户导航到打卡日历的界面 3. 系统显示一个月份视图的日历，每天的学习活动通过不同的颜色或标记显示 4. 用户可以点击任一天，查看那一天的详细学习活动和完成情况 5. 用户根据这些信息，可以调整未来的学习计划或继续维持现有计划 | |
| **备选事件流**  A-1某一天没有学习数据   1. 用户点击一个没有学习记录的日期 2. 系统显示没有学习活动的消息   A-2日历加载失败   1. 系统由于技术问题无法加载日历 2. 系统显示错误消息，并建议用户重试 | |
| **补充约束-数据需求**  D-1系统应实时更新打卡日历，反映用户的最新学习记录  **补充约束-业务规则**  B-1打卡日历中的数据应只对本人可见，保护用户隐私 | |
| **待解决问题**  如何确保在用户学习活动极其频繁的情况下，打卡日历的性能不受影响 | |
| **相关图** | |

|  |  |
| --- | --- |
| **用例名** | 登录 |
| **简要描述** | 用户通过输入用户名和密码来访问系统，以开始使用各项学习功能 |
| **参与者** | 用户 |
| **涉众** | 用户：直接通过登录操作来访问系统中的个人账户和学习材料  系统管理员：监控登录活动以确保系统安全 |
| **相关用例** | 无 |
| **前置条件** | 用户已注册并拥有有效的用户名和密码 |
| **后置条件** | 用户成功登录系统，获得对系统功能的访问权限 |
| **基本事件流**   1. 用户打开登录界面 2. 用户输入用户名和密码 3. 用户点击登录按钮 4. 系统验证输入的用户名和密码的正确性 5. 验证成功后，系统授予用户访问权限，并导向主界面 6. 用户开始使用系统的其他功能 | |
| **备选事件流**  A-1用户名错误或密码错误   1. 系统显示错误消息，提示用户名或密码不正确 2. 用户可以重新输入用户名和密码，也可以选择结束该用例 | |
| **补充约束-业务规则**  **补充约束-非功能需求**  安全性：密码应采用加密方式存储，有关密码的加密算法待定 | |
| **待解决问题**  关于用户名和密码的管理与维护功能还需要进一步明确 | |
| **相关图** | |

|  |  |
| --- | --- |
| **用例名** | 查看学习数据 |
| **简要描述** | 用户可以查看自己的学习历史和进度，包括学习频率、已学单词数量、复习次数以及学习计划的完成情况 |
| **参与者** | 用户 |
| **涉众** | 用户：从中获取个人学习效果的反馈，用于评估和调整学习策略 |
| **相关用例** | 创建学习计划：用户的学习计划完成情况会被记录，并显示在学习数据中 |
| **前置条件** | 用户必须已经登录系统 |
| **后置条件** | 用户查看他们的学习数据 |
| **基本事件流**   1. 用户登录到系统 2. 用户导航到查看学习数据的界面 3. 用户选择查看特定时间段的学习数据 4. 系统提取并显示所请求时间段内的学习数据 5. 用户查看数据，包括图表和统计信息 | |
| **备选事件流**  A-1无法检索数据   1. 系统显示错误消息，说明数据无法加载。 2. 用户选择重试或退出 | |
| **补充约束-数据需求**  D-1系统必须能实时更新并展示学习数据，以反映用户的最新学习活动  D-2数据应在用户界面中以易于理解的格式展示，例如图表和进度条  **补充约束-业务规则**  B-1系统应保证数据的私密性，仅允许用户查看自己的学习数据 | |
| **待解决问题** | |
| **相关图** | |

## 2.3 概念类图

我们的英语学习系统主要面向希望提升英语水平的学生和专业人士，主要实现个性化英语学习内容的生成，包括定制的英文短文和语音生成服务，以及用户学习行为的分析和反馈，具体的概念类图如图2.3所示：

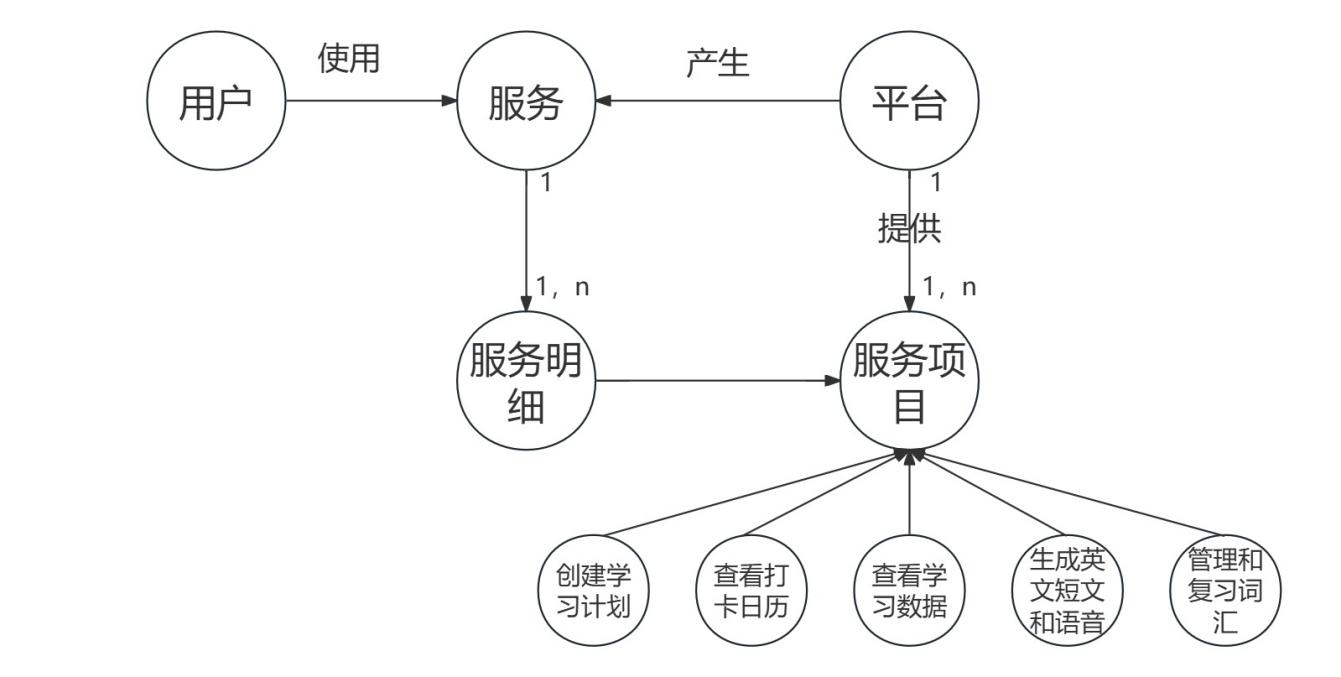


图2.3 概念类图

# 3 系统设计

## 3.1 概要设计

### 3.1.1 系统体系结构设计

系统体系结构设计图如图3.1.1所示：

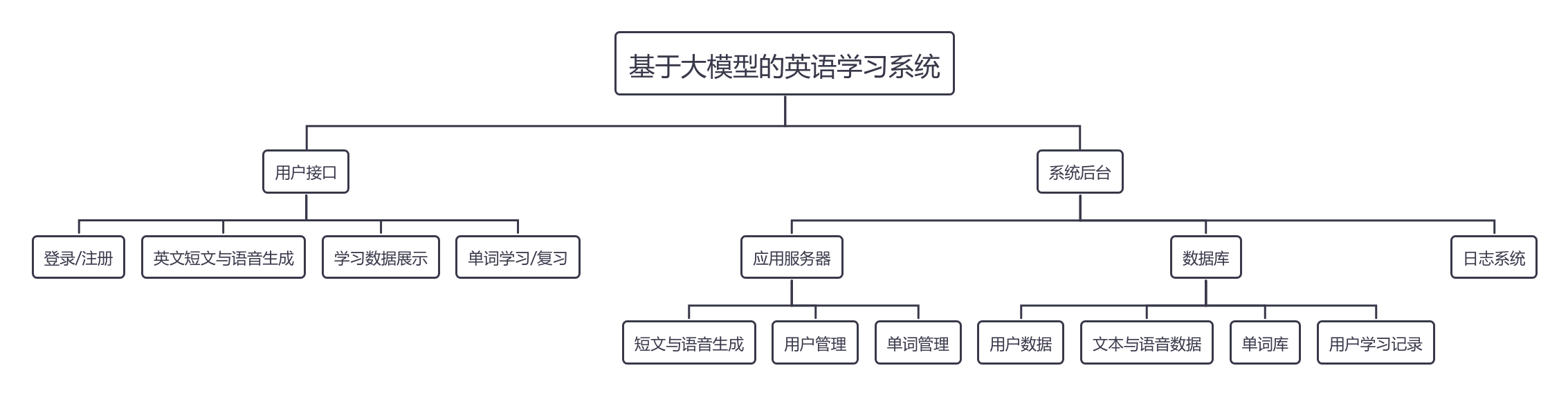


图3.1.1 英语学习系统结构设计图

### 3.1.2 数据库表设计

AI数据表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| Id | Int | Primary key | 唯一标识符 |
| create\_at | Int | Not null | ------------ |
| token | Int | Not null | ------------ |
| user\_id | Int | Not null | 调用者id |

学习记录表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| id | Int | Primary key | 唯一标识符 |
| book\_id | Int | Not null | 属于哪本书 |
| create\_time | Int | Not null | 学习开始的时间 |
| last\_review\_time | Int | Not null | 上次复习的时间 |
| stage | Int | Not null | ------------ |
| strange | Int | Not null | ------------ |
| update\_time | Int | Not null | ------------ |
| user\_id | Int | Not null | ------------ |
| word\_id | Int | Not null | ------------ |

学习进行表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| id | Int | Primary key | 唯一标识符 |
| book\_id | Int | Not null | 属于哪本书 |
| start\_time | Int | Not null | 学习开始的时间 |
| end\_time | Int | Not null | 学习结束的时间 |
| type | Int | Not null | ------------ |
| user\_id | Int | Not null | ------------ |
| word\_id | Int | Not null | ------------ |

用户表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| id | Int | Primary key | 唯一标识符 |
| create\_time | Int | Not null | 创造时间 |
| password | varchar | Not null | 登陆密码（md5加密后） |
| role | Int | Not null | 用户的角色 |
| status | Int | Not null | ------------ |
| username | varchar | Not null | 用户名 |

用户登录表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| user\_id | Int | Primary key | 唯一标识符 |
| gender | Int | 0/1 | 性别 |
| grade | varchar | Not null | ------------ |
| last\_login\_time | Int | Not null | ------------ |
| nickname | varchar | Not null | ------------ |
| school | varchar | Not null | ------------ |

单词书表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| id | Int | Primary key | 唯一标识符 |
| book\_id | varchar | Not null | ------------ |
| pic\_url | varchar | Not null | 单词书封面图片链接 |
| title | varchar | Not null | 标题 |
| word\_num | Int | Not null | 单词数 |
| tags | varchar | Not null | ------------ |

单词表

|  |  |  |  |
| --- | --- | --- | --- |
| 属性 | 数据类型 | 约束条件 | 说明 |
| id | Int | Primary key | 唯一标识 |
| word\_rank | Int | Not null | ------------ |
| head\_word | varchar | Not null | ------------ |
| word | varchar | Not null | ------------ |
| book\_id | Int | Not null | ------------ |

## 3.2 详细设计

英语学习系统的用例设计图如下图所示：

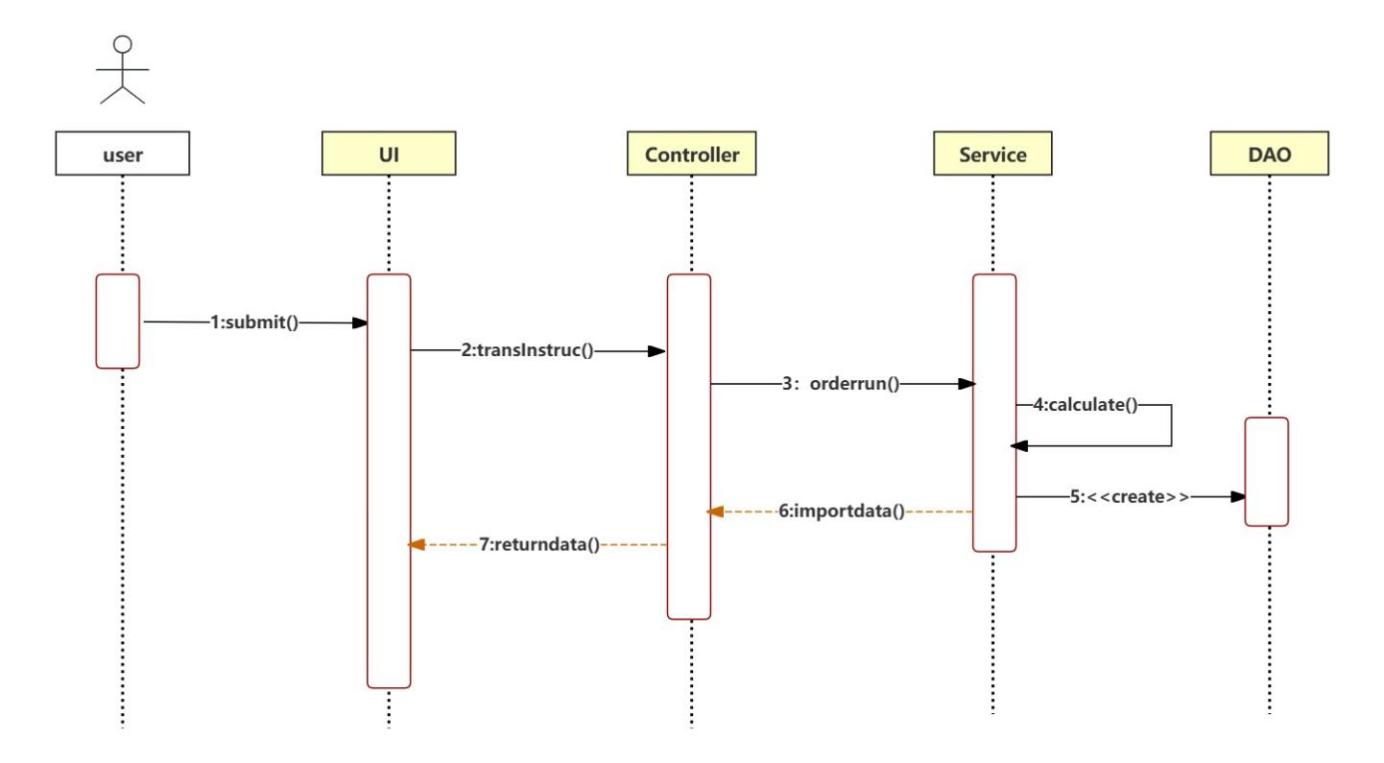


图3.2.1 用例设计总顺序图

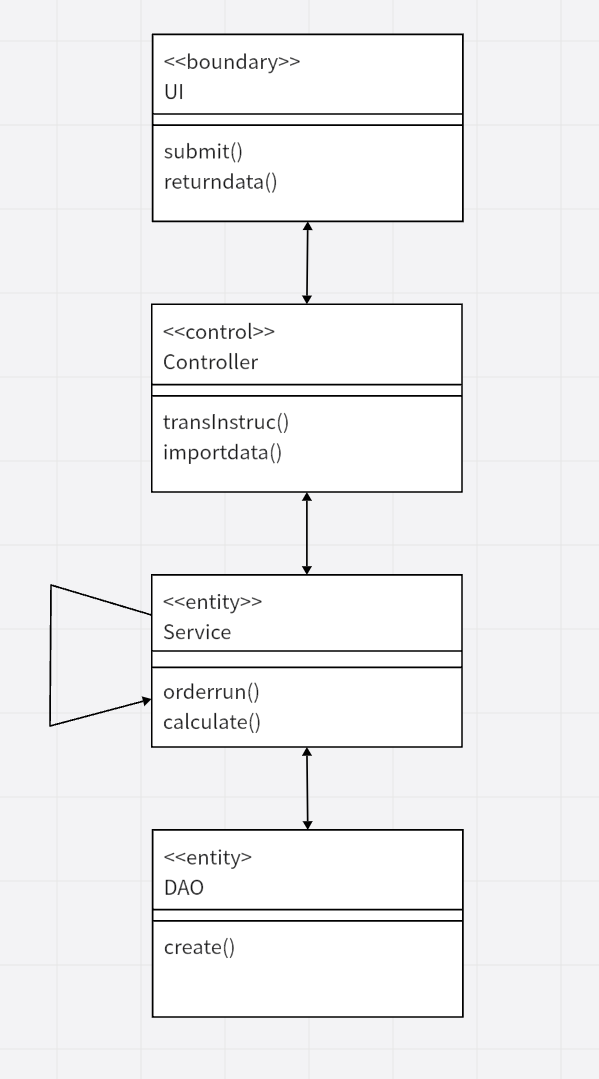
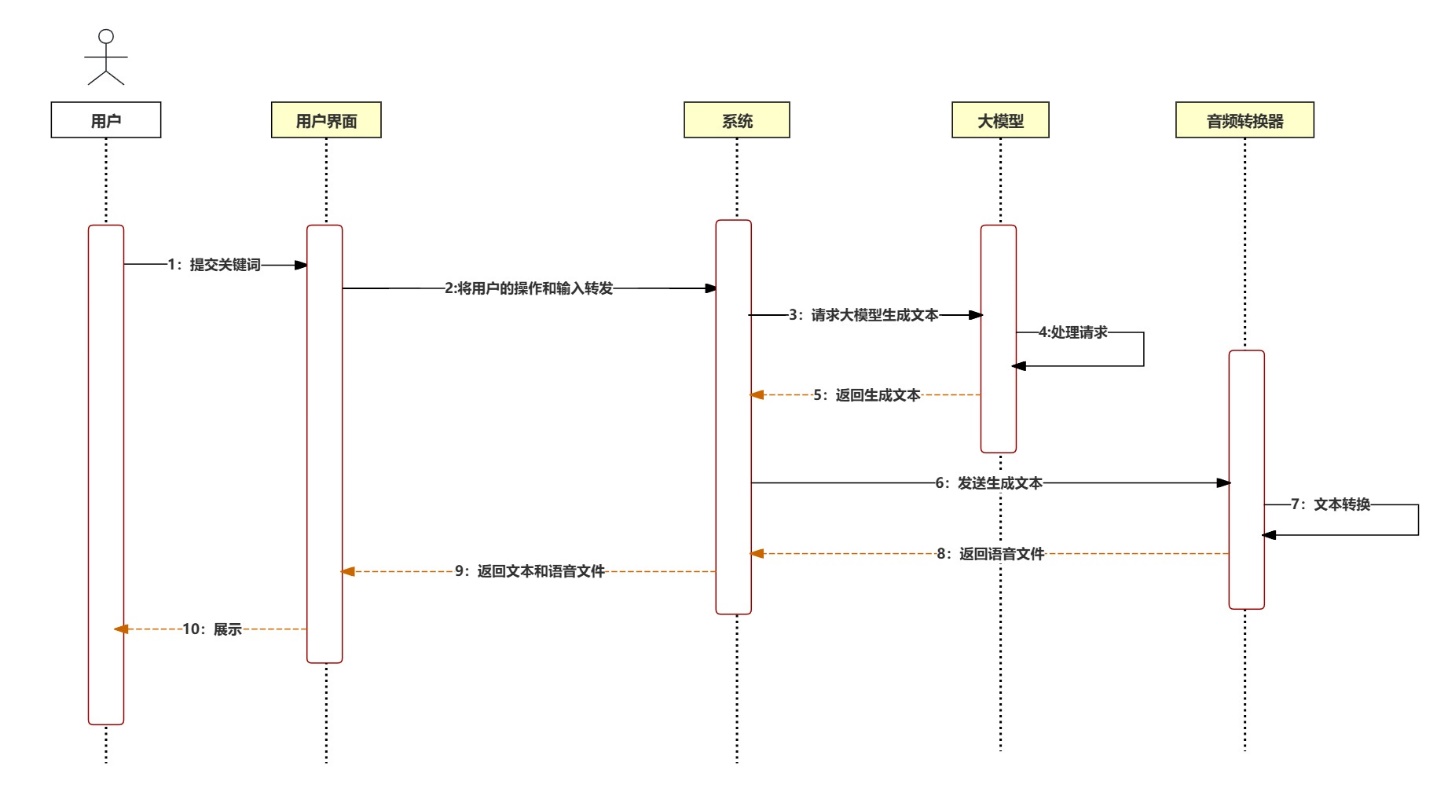


图3.2.2 用例设计总类图

大模型生成短文和语音用例细化顺序图：

图3.2.3大模型短文和语音生成顺序图

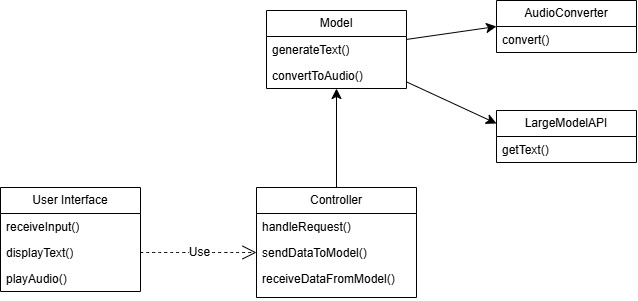


图3.2.4 大模型短文和语音生成类图

图注：

**1. User Interface (用户界面)**

职责：负责与用户互动，接收用户的输入并展示输出结果。

操作：

receiveInput()：接收用户输入的关键词或主题。

displayText()：显示生成的文本。

playAudio()：播放生成的语音。

**2. Controller (控制器)**

职责：协调用户界面与后端模型之间的交互，管理数据流和命令。

操作：

handleRequest()：处理用户的请求。

sendDataToModel()：将数据发送到模型以生成文本。

receiveDataFromModel()：从模型接收生成的文本。

**3. Model (模型)**

职责：封装业务逻辑，负责生成文本和语音的核心处理。

操作：

generateText()：根据用户输入生成文本。

convertToAudio()：将文本转换成语音。

**4. LargeModelAPI (大模型API接口)**

职责：提供访问大型语言模型的接口，用于生成文本。

操作：

getText()：根据提供的参数生成文本。

**5. AudioConverter (音频转换器)**

职责：负责将文本数据转换为音频格式。

操作：

convert()：接收文本并转换成语音文件。

单词学习用例细化顺序图：

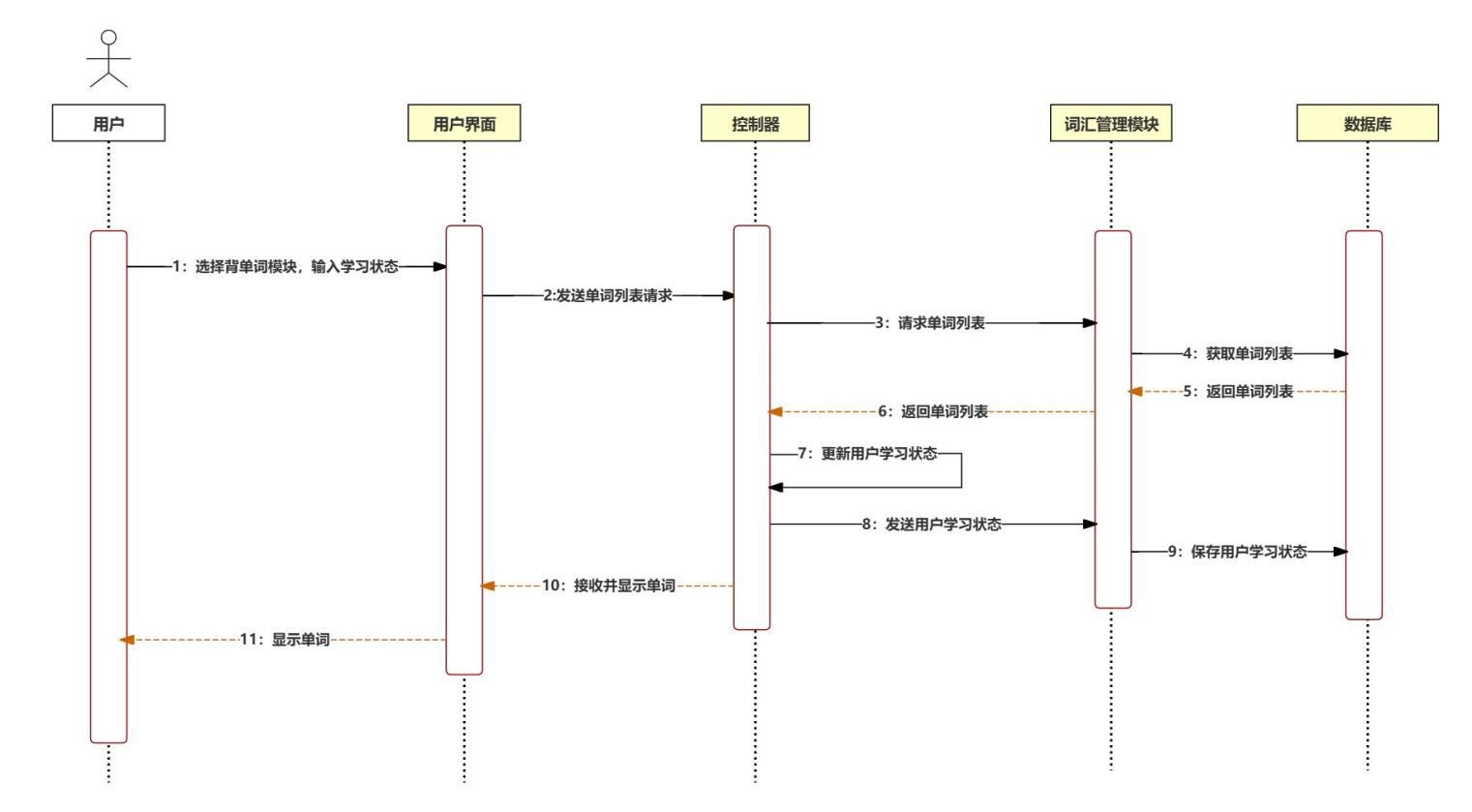


图3.2.5单词学习顺序图

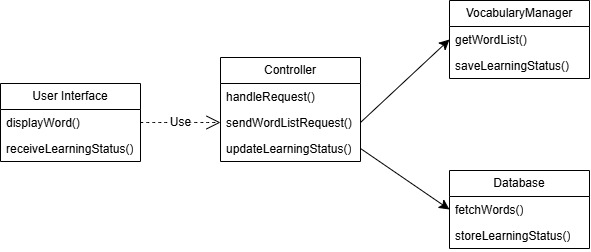


图3.2.6 单词学习类图

图注：

**1、User Interface（用户界面）：**

职责：负责与用户互动，显示单词和释义，并接收用户的学习状态标记。

操作：

displayWord()：显示单词及其释义。

receiveLearningStatus()：接收用户对单词的学习状态标记。

**2、Controller（控制器）：**

职责：协调用户界面与词汇管理模块之间的交互，处理数据流和命令。

操作：

handleRequest()：处理用户的背单词请求。

sendWordListRequest()：向词汇管理模块请求单词列表。

updateLearningStatus()：将用户的学习状态更新到词汇管理模块。

**3、VocabularyManager（词汇管理模块）：**

职责：管理单词数据，提供单词列表并记录用户的学习状态。

操作：

getWordList()：提供需要背诵的单词列表。

saveLearningStatus()：保存用户对单词的学习状态。

**4、Database（数据库）：**

职责：存储和管理单词及其释义，以及用户的学习状态数据。

操作：

fetchWords()：从数据库中获取单词列表。

storeLearningStatus()：将用户的学习状态存储到数据库中。

# 4 系统实现

## 4.1 应用的技术介绍

### 4.1.1 Android

安卓（Android）是一种基于Linux内核（不包含GNU组件）的自由及开放源代码的移动操作系统。

### 1. 开放性和多样性

* **开源特性**：Android 基于 Apache 许可证发布，制造商和开发者可以在不支付许可费的情况下自由地访问和修改源代码。
* **硬件独立性**：Android 能够运行在多种硬件平台上，包括 ARM, x86, 和 MIPS 架构，支持从高性能到低功耗的各类设备。
* **设备多样化**：Android 支持各种形态的设备，如智能手机、平板电脑、可穿戴设备和智能电视，使得整个生态系统丰富多彩，满足不同市场的需求。

### 2. 易于开发

* **Android Studio IDE**：Android Studio 提供了代码编辑、调试、性能监测工具，以及用于设计应用界面的可视化编辑器，极大简化了开发过程。
* **丰富的API库**：Android 提供了广泛的API，这些API让开发者能够访问设备硬件，比如相机、GPS 和加速度计，还包括用于网络通信、数据管理和图形渲染的API。
* **强大的开发者社区**：全球有数百万的开发者和技术论坛支持 Android 开发，新开发者可以轻松找到解决问题的资源和指导。

### 3. 丰富的用户界面设计

* **Material Design**：Android 推出了 Material Design 设计语言，它使用响应式动画、过渡效果、深度效果和设计美学来提高用户体验。
* **自适应布局**：Android 提供了多种布局管理器，支持创建可在不同屏幕尺寸和分辨率上有效显示的用户界面。

### 4. 组件化与模块化

* **组件基础**：Android 应用由不同的组件组成，如活动（Activity）、服务（Service）、内容提供者（Content Provider）和广播接收者（Broadcast Receiver）。
* **Intent 机制**：通过 Intent，应用组件之间不仅可以进行内部通信，还可以与其他应用的组件交互，实现高度的模块化和复用。

### 5. 数据和文件管理

* **SQLite数据库**：Android 内置了 SQLite 数据库，为应用提供轻量级的数据存储方案。
* **权限控制**：Android 6.0 及更高版本引入了运行时权限模型，增强了应用安全，使用户能够控制应用对特定数据的访问权限。

### 6. 性能优化和多任务处理

* **虚拟机优化**：Android 的运行时环境 ART（Android Runtime）使用 Ahead-Of-Time (AOT) 和 Just-In-Time (JIT) 编译策略，优化应用性能和响应速度。
* **后台管理**：Android 限制了后台应用的运行，有效管理系统资源，提升设备性能和电池寿命。

### 7. 快速迭代和广泛的市场接入

* **应用分发**：开发者可以通过 Google Play 和其他 Android 应用市场轻松发布应用，访问全球范围的广大用户群。
* **即时应用技术**：Android Instant Apps 允许用户在不安装完整应用的情况下使用应用的某些功能，从而提升用户的接触和体验。

Android 不仅为用户提供了丰富多彩的移动体验，也为开发者提供了一个功能强大、灵活的开发平台。

### 4.1.2 springboot

Spring Boot 是一个开源框架，其主要目标是简化 Spring 应用的设置和开发过程，特别是微服务架构的应用。Spring Boot 通过提供默认配置和一系列便捷功能，使得开发者能够快速启动并运行基于 Spring 的应用程序，而无需花费大量时间进行复杂的配置。

### 1. 快速启动和易于配置

* **自动配置**：Spring Boot 可以自动配置应用程序的大部分组件。这基于类路径上的 jar 依赖以及已定义的各种属性。例如，如果应用程序在其类路径上包含 Spring MVC，则自动配置一个 Spring 应用程序上下文，包括配置 DispatcherServlet。
* **启动器依赖（Starters）**：Spring Boot 提供了一系列的启动器依赖，这些都是预设的依赖模板，用以简化构建配置。例如，spring-boot-starter-web 包括构建 web 应用所需的所有依赖，如 Spring MVC、Tomcat 或 Jetty 作为默认的嵌入式服务器。

### 2. 综合开发工具

* **Spring Boot CLI**：这是一个命令行工具，可以用来快速开发基于 Spring Boot 的应用程序。它利用 Groovy 语言的简洁性，允许开发者通过最少的代码快速启动新项目。
* **Actuator**：Spring Boot Actuator 提供了一组生产级的特性，帮助你监控和管理你的应用。它暴露了多个 HTTP 或 JMX 端点，用于查看应用详情，如健康状态、环境属性、配置设置、线程详情等。

### 3. 操作简单，提升开发效率

* **内嵌服务器支持**：Spring Boot 支持内嵌的 Tomcat、Jetty 或 Undertow 服务器，使得开发者不必在开发阶段单独设置和管理 Web 服务器。
* **环境管理**：支持通过 application.properties 或 application.yml 文件来配置应用的不同环境设置，如开发、测试和生产环境。这些配置可以包括数据库的连接信息、外部服务的 API 密钥等。

### 4. 丰富的生态系统

* **集成 Spring 生态系统**：Spring Boot 完美集成了 Spring 的其他项目，如 Spring Data, Spring Security, Spring REST, Spring Social, Spring Batch, Spring Data JPA, 等等。
* **社区支持**：拥有庞大且活跃的开发社区，提供大量的教程、指南和第三方插件，这对于解决开发过程中遇到的问题非常有帮助。

### 5. 微服务支持

* **服务发现与注册**：Spring Boot 可以与 Spring Cloud 集成，后者提供了服务发现与注册的功能，非常适合构建基于微服务架构的应用。
* **配置中心集成**：支持与配置中心集成，如 Spring Cloud Config，允许中央管理微服务的所有配置。

### 6. 监控与管理

* **Actuator 监控**：Actuator 的端点可以被用来深入了解和管理应用程序运行时的行为，包括自动检测应用的健康状态、数据库状态、以及与应用相关的各种度量和统计信息。

### 4.1.3 MVC模式

MVC（Model-View-Controller）模式是一种设计模式，广泛应用于构建用户界面的软件架构中。这种模式通过将应用分为三个核心组成部分：模型（Model）、视图（View）和控制器（Controller），来实现关注点分离，这有助于管理复杂的软件设计并提高应用的可维护性、可扩展性和可测试性。

### 1. 模型（Model）

* **定义**：模型代表应用的数据逻辑。它直接管理数据、逻辑和规则，是应用程序中处理数据逻辑的部分，如数据库管理、数据操作逻辑等。
* **功能**：模型负责存取数据，以及对数据进行处理的业务逻辑。它通常响应来自控制器的指令，对数据进行操作，并提供数据状态的更新。
* **独立性**：模型与用户界面无关，它不涉及任何关于数据如何展示的逻辑。

### 2. 视图（View）

* **定义**：视图是应用中用户界面的部分，负责数据的展示。它将模型中的数据用户友好地显示给用户。
* **功能**：视图通过从模型获取数据，并转化成合适的格式显示给用户，形成用户界面。视图的展示方式可以多样，如图表、表格、图形界面等。
* **响应性**：视图通常会在模型数据变化时接收通知，并重新渲染显示界面，这通常通过观察者模式实现。

### 3. 控制器（Controller）

* **定义**：控制器是应用的行为部分，负责接受用户的输入，并调用模型和视图去完成请求的操作。
* **功能**：控制器解释用户输入，将其转换为对模型的命令，从而影响应用的状态。它也可以调用相应的视图来改变用户界面。
* **协调者**：控制器起到了模型和视图之间的中介者角色，确保数据流动和业务逻辑的正确执行。

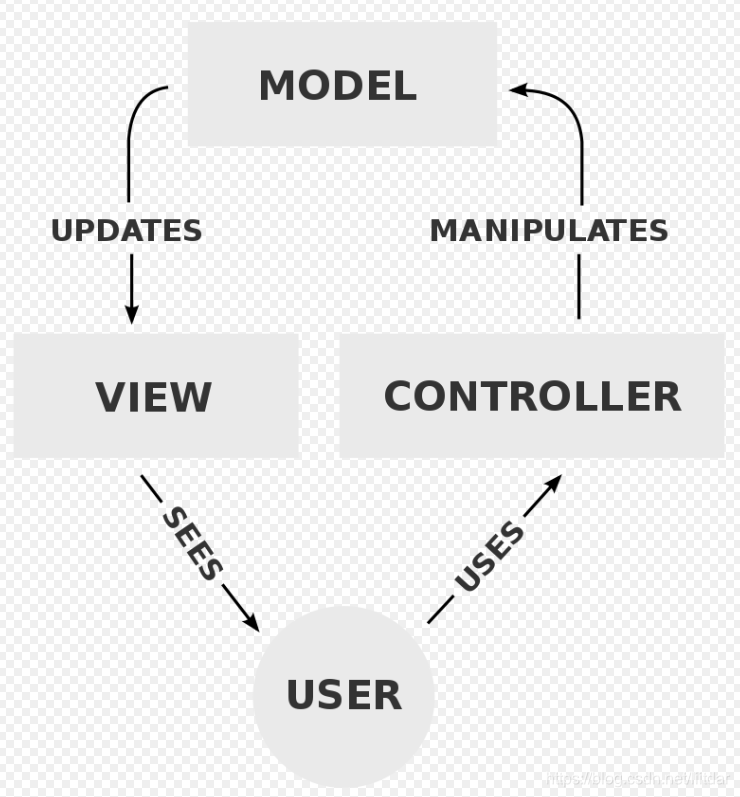


图4.1.3 MVC模式的描述

### MVC 的优势和应用

* **分离关注点**：通过分离数据模型、用户界面和用户输入，MVC 允许开发者独立修改一个部分而不影响其他部分。这种分离使得应用程序更易于管理和扩展。
* **提高组织性**：MVC 提供了一种清晰的架构模式，帮助开发者组织复杂的应用功能，使代码更加模块化。
* **易于测试**：MVC 使得单元测试变得更容易，因为组件之间的接口清晰，可以独立对模型、视图和控制器进行测试。
* **广泛应用**：MVC 模式被广泛应用于各种编程框架中，如 Ruby on Rails、Django（MVT，其中的模板类似于视图）、Spring MVC、ASP.NET MVC 等。

MVC 模式通过其结构化和模块化的设计，提高了大型应用程序开发的效率和质量，是现代软件开发中不可或缺的一部分。

### 4.1.4 MySQL

MySQL 是一个广泛使用的开源关系数据库管理系统（RDBMS），由瑞典的 MySQL AB 公司开发，后来被 Sun Microsystems 收购，最终在 2010 年被 Oracle 公司收购。MySQL 是基于结构化查询语言（SQL）的系统，它是构建动态网站和应用程序的数据存储后端的热门选择。

### 核心特性

* **开源和成本效益：MySQL 是开源的，可以免费使用，这使得它对于初创公司和开发者来说是一个成本效益高的数据库解决方案。尽管如此，也提供了付费版本的 MySQL Enterprise Edition，包含额外的功能和企业级的支持。**
* **性能和可靠性：MySQL 被设计为高度可靠和性能优异的数据库管理系统。它支持多种存储引擎，如 InnoDB（支持事务处理和外键）和 MyISAM（高速存储，但不支持事务）。这些存储引擎使得 MySQL 在各种应用场景下都能提供优异的性能。**
* **易于使用：MySQL 以其简单性和易用性而闻名。它可以在多种操作系统上运行，如 Linux、Windows 和 macOS。MySQL 提供了一个用户友好的命令行界面，并且有许多图形用户界面工具可用，如 phpMyAdmin 和 MySQL Workbench。**
* **高度可扩展：MySQL 支持从小型应用到大型企业应用的扩展需求。它可以处理拥有上千万条记录的大型数据库，同时也适用于只有几百条记录的小型数据库。**
* ****安全性：**MySQL 包含了强大的安全特性，如加密、SSL支持和访问控制列表（ACL）用于更好地管理用户权限和保护数据。**
* **广泛的应用和社区支持：作为全球最受欢迎的数据库之一，MySQL 拥有一个庞大的开发者社区，提供了丰富的文档、论坛、教程和第三方工具，使得解决问题和学习使用 MySQL 更加容易。**

### 应用场景

* **网站和网络应用：MySQL 是许多动态网站的首选数据库，尤其是 LAMP（Linux, Apache, MySQL, PHP/Python/Perl）和 MEAN（MongoDB, Express.js, Angular, Node.js）堆栈的一部分。**
* **企业应用：许多企业级应用选择 MySQL 作为其后端数据库，以处理复杂的业务逻辑和大量数据。**
* **嵌入式应用：MySQL 的轻量级版本（如 MySQL Embedded）被用于设备和其他嵌入式应用中，提供可靠的数据管理而无需专门的数据库管理人员。**

### 4.1.5百度千帆大模型（ERNIE-Speed-128K）

百度千帆大模型（ERNIE-Speed-128K）是百度推出的最新一代大规模自然语言处理（NLP）模型，基于其旗舰的ERNIE（Enhanced Representation through kNowledge Integration）架构。这个模型是专为满足高效的处理速度和高精度的语言理解需求设计的。它结合了最新的机器学习技术和算法优化，以提供在各种NLP任务上的卓越性能。

### 核心技术和架构

**1. 预训练技术：**

ERNIE-Speed-128K 使用了大规模的预训练方法，其中“128K”指的是模型在训练过程中处理的语言模型的规模。预训练包括广泛的文本语料库，覆盖了从新闻文章到社交媒体帖子的各种文本类型。这种广泛的数据输入帮助模型捕捉复杂的语言模式和语义关系。

1. **速度优化：**

模型的设计重点在于优化运行速度，实现了通过多种技术如模型量化、剪枝和高效的前向传播算法来减少计算资源需求和提高响应速度。量化技术减少了模型的内存占用，而剪枝技术则通过去除冗余的网络连接来简化模型结构。

**3. 知识增强：**

ERNIE系列的特色之一是其能够整合外部知识（例如知识图谱），ERNIE-Speed-128K 继续沿用并加强这一特性。通过融合来自特定领域的结构化知识，模型能够更准确地理解专业术语和上下文，特别是在复杂的行业应用中，如金融、医疗和法律。

**4. 多任务处理能力：**

ERNIE-Speed-128K 支持多任务学习，能够在一个统一的框架中同时训练多种任务，如情感分析、文本分类、问答系统和机器翻译。这种能力提高了模型的灵活性和适用性，减少了为每个单独任务训练单独模型的需要。

**5. 自适应调整：**

模型具有高度的自适应性，可以根据不同的业务需求和运行环境进行调整。它支持从云到边缘计算设备的各种部署选项，使其可以在资源受限的设备上运行，同时保持高性能。

### 应用场景和优势

由于其高效的处理能力和广泛的适用性，ERNIE-Speed-128K 特别适合以下应用场景：

* **企业级应用：在企业环境中，该模型可以用于自动化客户服务、提供决策支持系统以及增强业务智能分析。**
* **内容个性化和推荐系统：利用其深度学习能力，可以为用户个性化内容，如新闻、视频和产品推荐，提供更精准的推荐算法。**
* **社交媒体监控和分析：模型能够处理和分析大量的社交媒体数据，用于情绪分析、趋势预测和公众舆情监控。**
* **语言教育和辅助工具：用于开发语言学习应用，帮助用户学习新语言，提供语法校正和写作辅助。**

ERNIE-Speed-128K 的推出，不仅标志着百度在大模型和人工智能技术领域的进一步进展，也为企业和开发者提供了一个强大的工具，以开发更智能、更快速的NLP应用。

## 4.2 各功能模块的实现

### 4.2.1 基本功能：

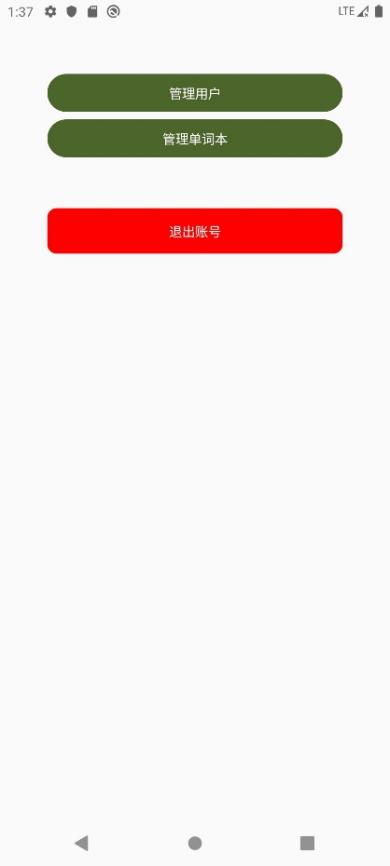
**1、管理员登陆界面**

管理员通过此界面登录进入管理员界面，行使管理员权限，可以查看用户，对用户进行增删和管理、对单词本进行管理



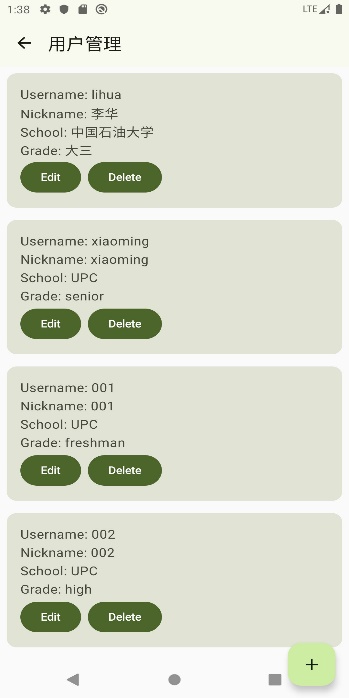
**2、管理员用户界面**

管理员登录成功，可以选择管理用户或者单词本，或者退出账号，通过此页面对整体进行统一的管理，可以保证软件更加方便的进行更新操作。



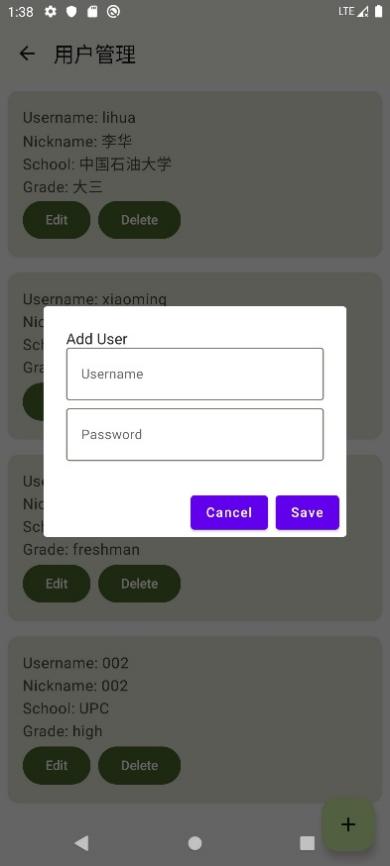
**3、管理员用户管理**

管理员对用户进行操作，可以增加或者删除用户，并且查看所有用户的基本信息，便于对用户进行统一的管理和概览。



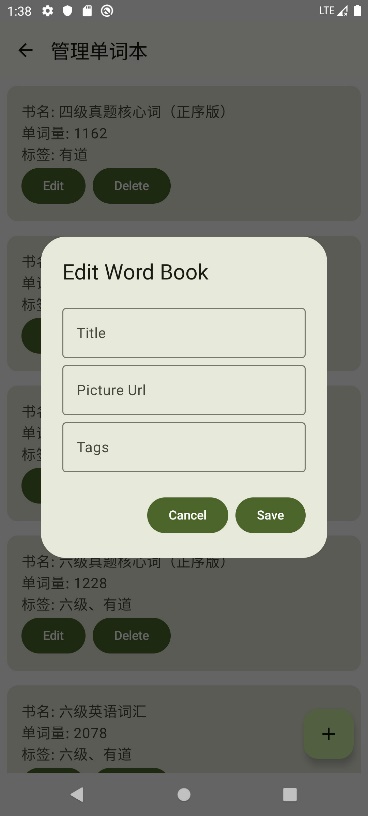
**4、管理员添加用户**

以管理员身份添加用户，设定姓名和密码，可以更方便的对用户进行管理



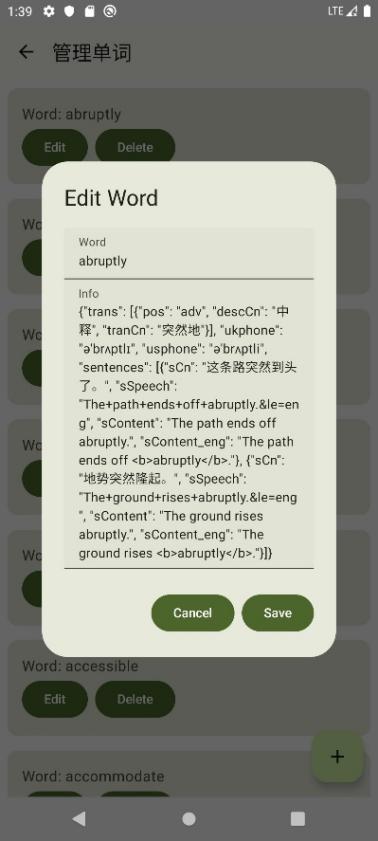
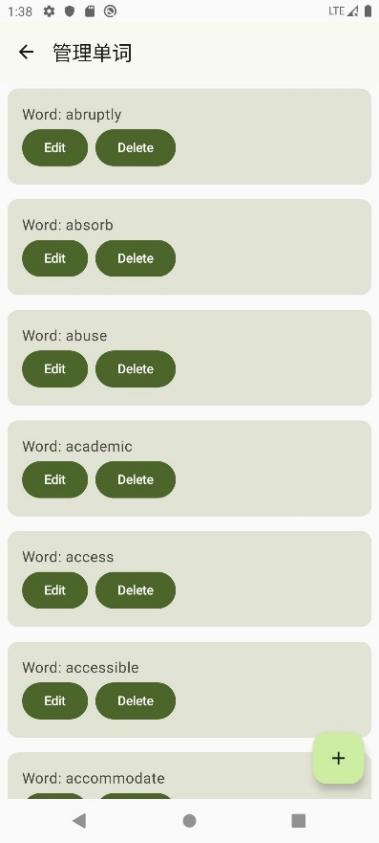
**5、管理员管理单词本**

管理员对单词书进行管理，直接对数据库内的数据进行改动，可以添加书的题目、标签，以便于后续单词库更新后，可以及时让用户使用最新的单词书进行学习



**6、管理员管理单词**

管理员对用户学习的单词进行查看和管理，可以在接收到用户端反馈的错误或者根据最新的内容对单词书内的单词进行及时的修改或者更新



**7、用户登录**

用户登陆界面，用户可以在此界面输入已经注册好的用户名和密码进入用户界面进行软件的使用



**8、用户制定学习计划**

用户在登陆进入界面后，可以选择单词书并根据自己的实际情况制定相关的学习计划，有规律的使用软件进行英语单词的学习，软件会把学习计划进行记录后进行统计每天的使用情况进行汇总。



**9、用户学习计划**

在此页面可以查看已经制定的学习计划，选择已有的单词本进行添加，然后就可以执行自己的学习计划，软件内已经内置了目前热门的单词库，用户可以根据自己的实际情况选择单词书进行学习完善自己的学习计划。



**10、用户单词本**

在制定完学习计划并且选择添加的单词书后，可以在此界面查看自己选择的单词书，如果想要更改已选的单词书，可以在此界面进行移除单词书操作



**11、单词本概览**

点开选择的单词本后，可以在此界面查看单词本内的单词，点击单词可以查看每个单词的中文释义和例句便于记忆和学习。



**12、单词学习**

用户在点击学习单词后，进入此界面进行的单词的学习，可以根据对单词的了解程度选择学会了、模糊、不认识，以便软件对所学过的单词进行分类，便于后续单词的复习，进行进一步的加强记忆



**13、单词复习**

在学习单词后对单词进行复习，可以对于单词的记忆程度，选择学会了、模糊、不认识，便于软件判断是否加入后续的复习中，增强对于单词的记忆效果。



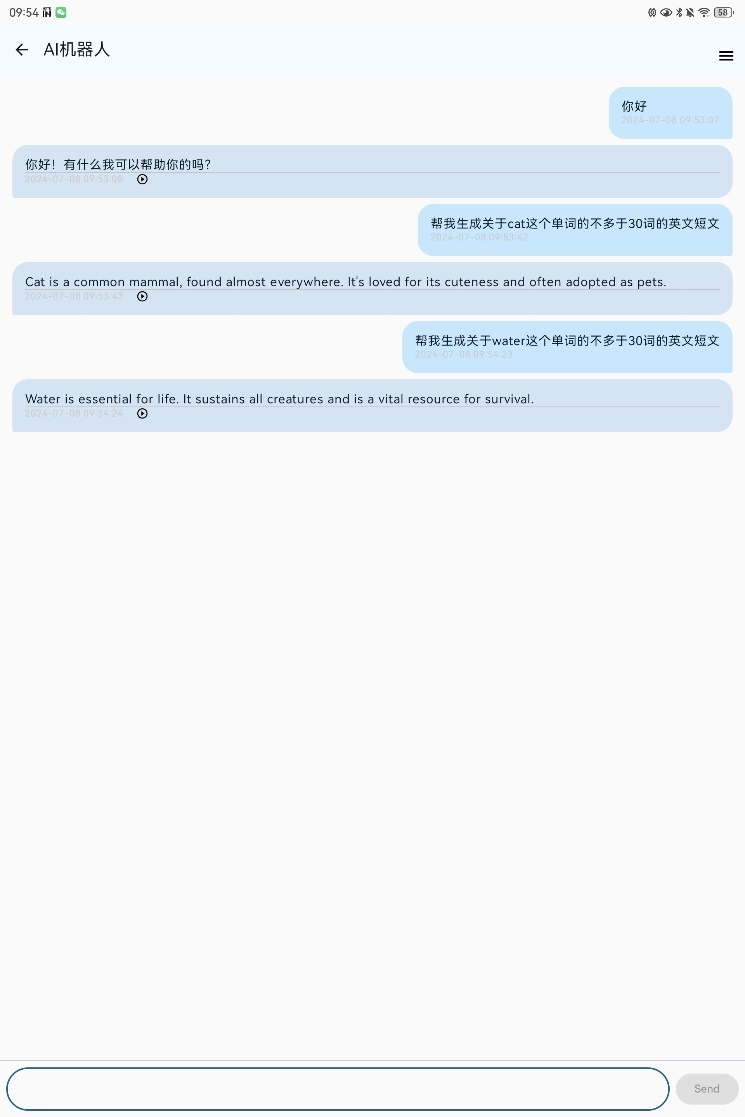
**14、用户数据统计**

软件可以根据用户每天对于单词的学习和复习进行数据统计，并且记录每天的学习时间，也可以根据这些数据，后期对于用户提供更加优化的细节



**15、AI对话**

用户可以在此界面和AI进行对话，可以根据自己的需求向AI提出自己的要求，例如根据某单词生成短文帮自己更加有效的记忆单词，还可以播放AI生成的语音，可以根据AI读音矫正自己的读音，有利于加强口语的练习，全方位的提升自己的英语技能。



**16、AI对话历史记录查询**

在与AI进行对话之后，用户可以在此页面查看与AI对话的记录，便于在学习后忘记进行查看记录和复盘学习内容



**17、用户个人信息**

在用户个人信息页面，用户可以看到自己的基本信息，包括自己的昵称、性别、学校和年级，并且在后续的过程中可以在此界面进行更改个人信息并进行保存

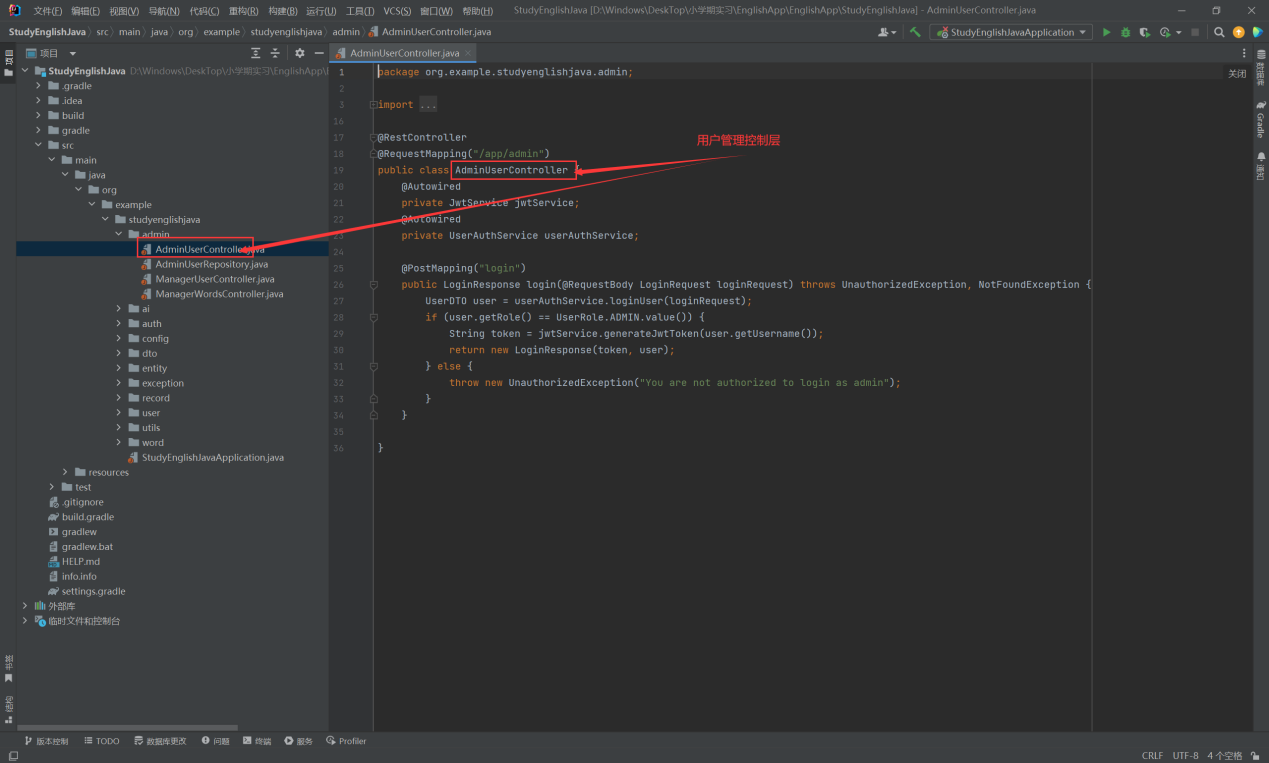


### 4.2.2 后端信息

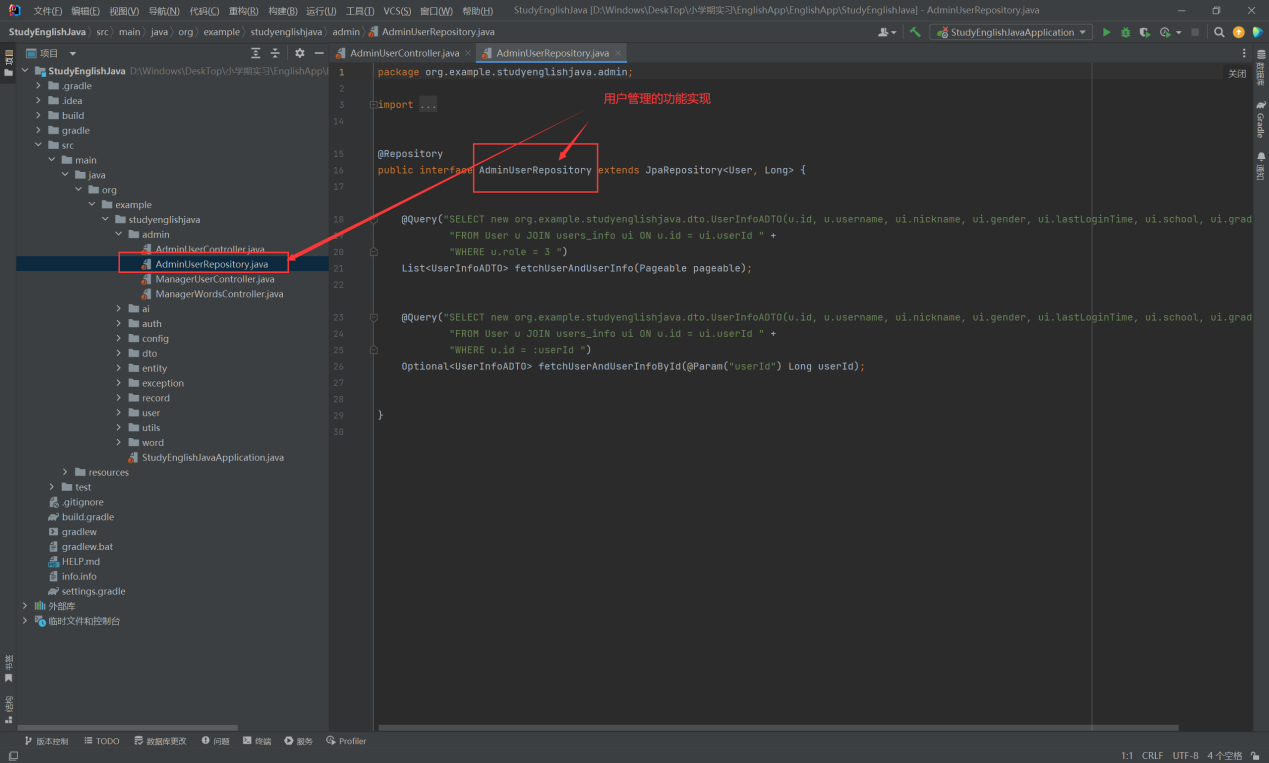
后端使用了SpringBoot框架，使用Java语言实现方法，与前端Android进行接收API进行开发。

**功能代码（部分）：**

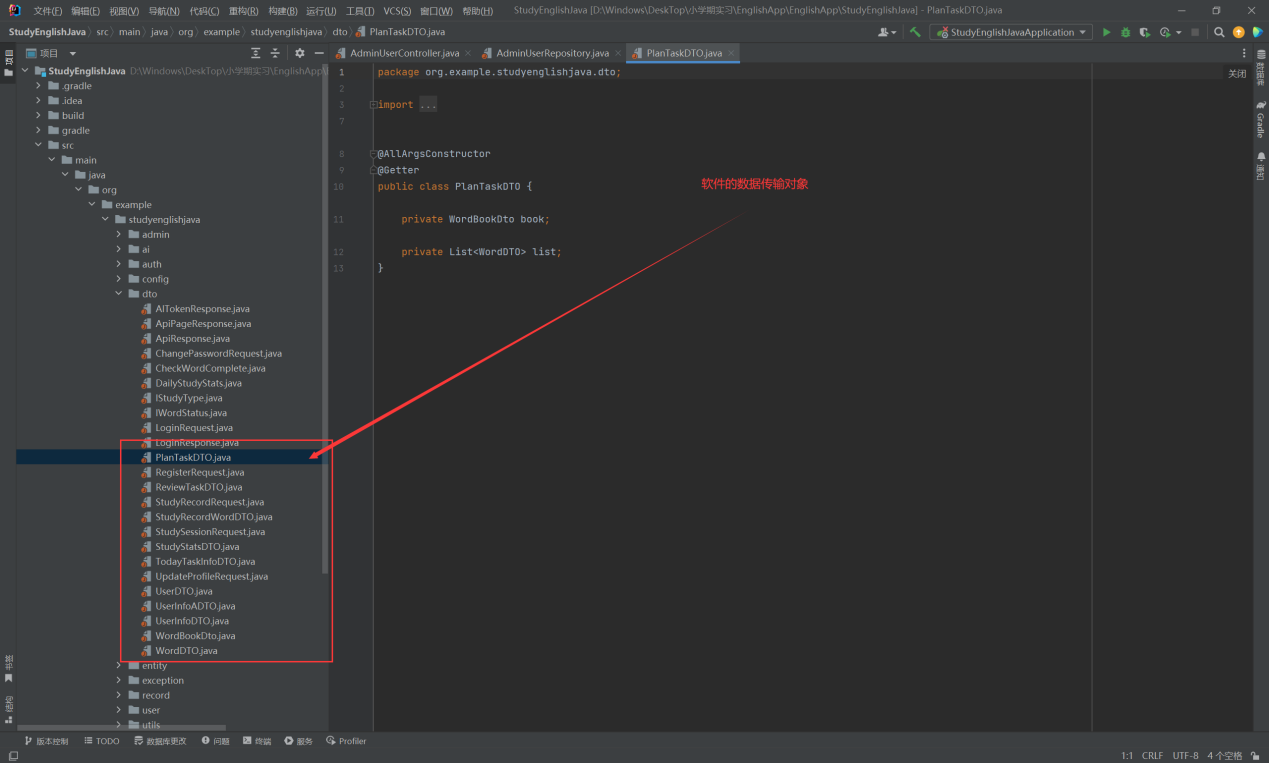
（1）用户管理控制层



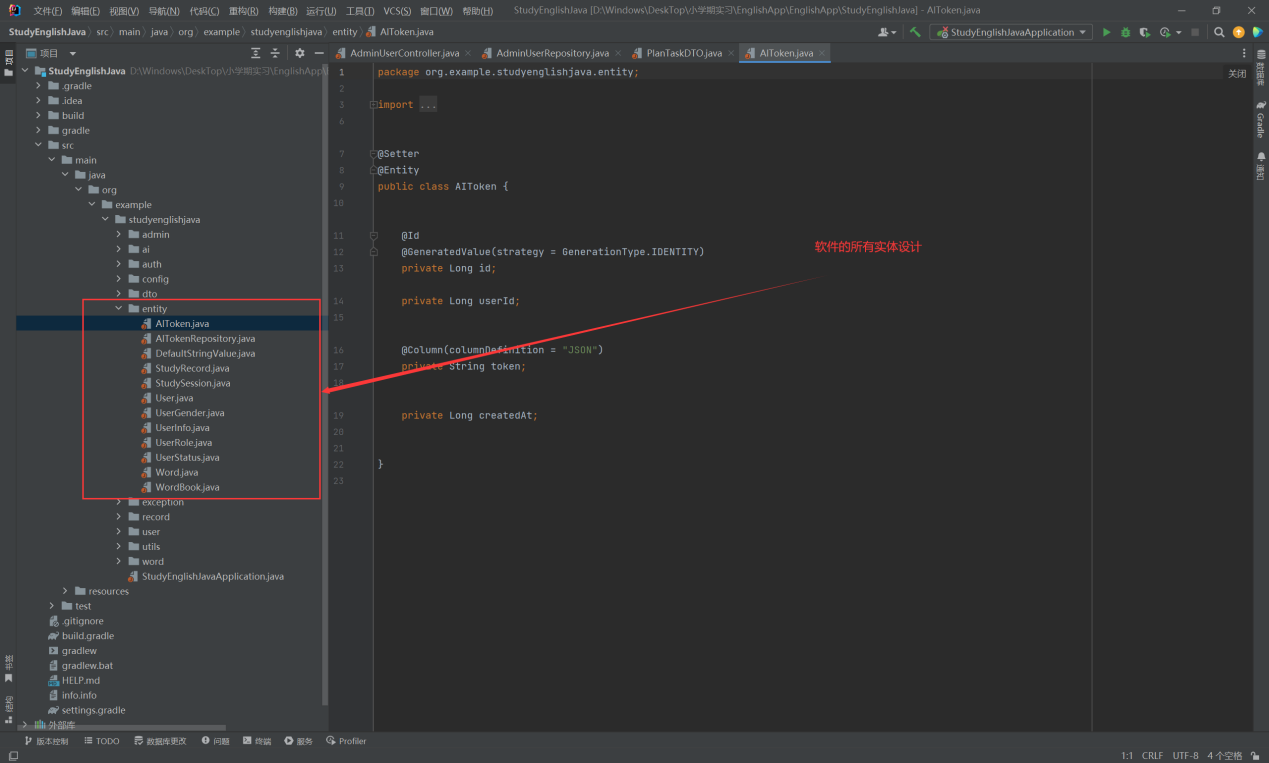
（2）用户管理方法的实现



（3）数据传输对象



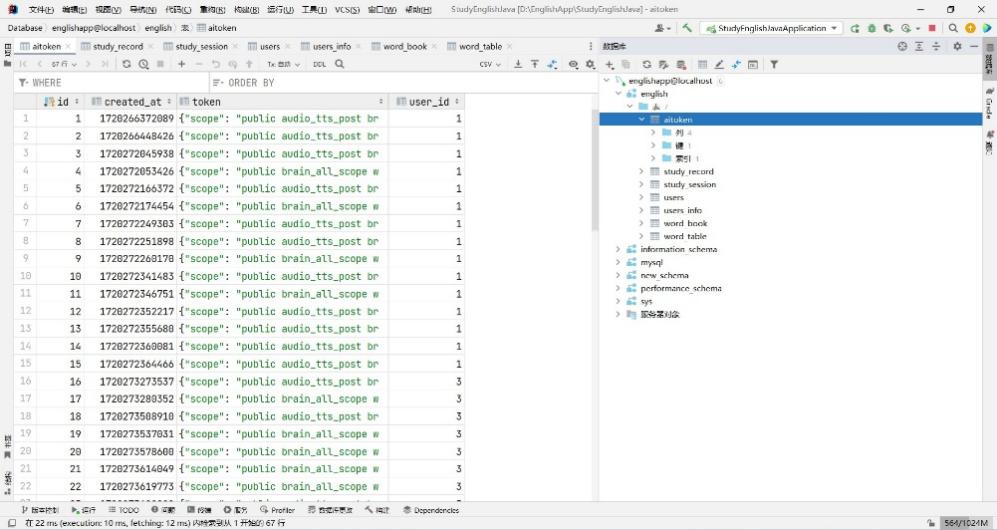
（4）数据的实体设计



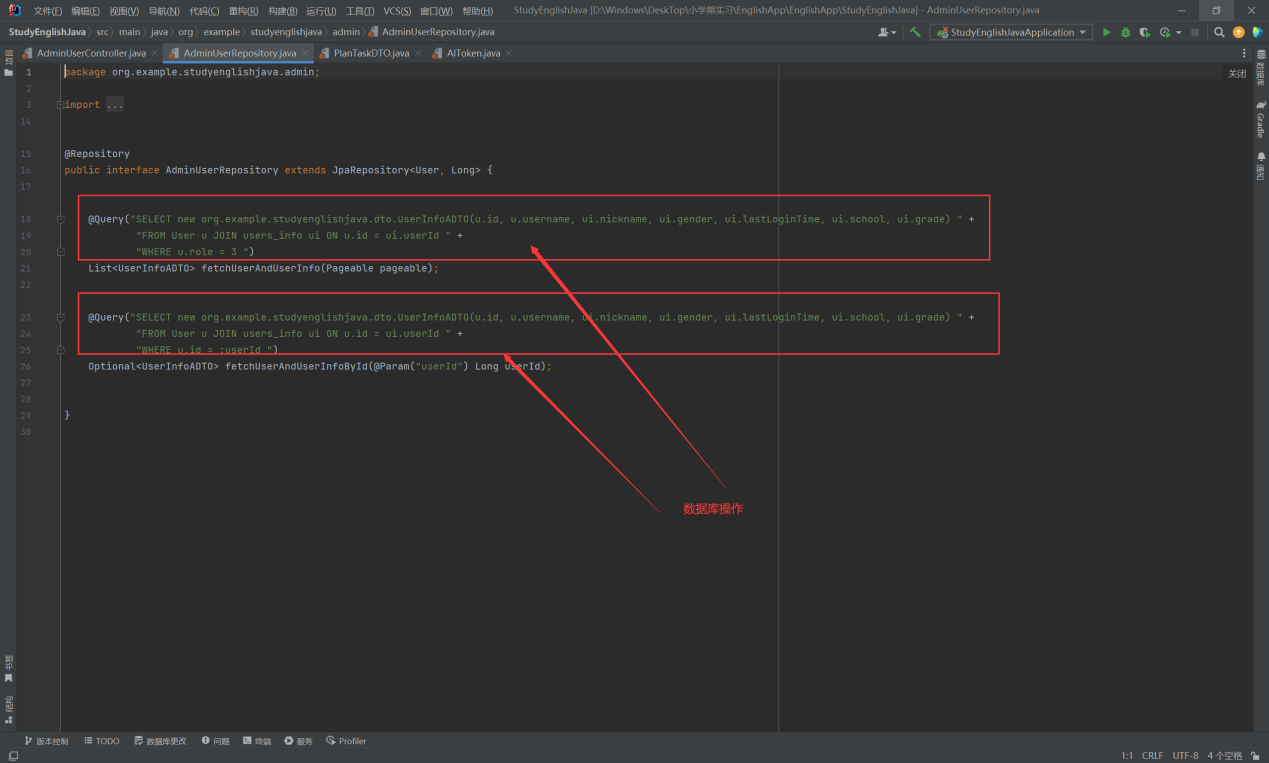
### 4.2.3 数据库连接

运用Mysql数据库对数据进行存取，其中建立了aitoken、study\_record、study\_session、users、user\_info、word\_book、word\_table表项，存取AI模型数据、学习记录数据、学习进行、用户、用户登录、单词书、单词表这些数据

**（1）数据库连接：（部分）**



**（2）数据库操作：（部分）**

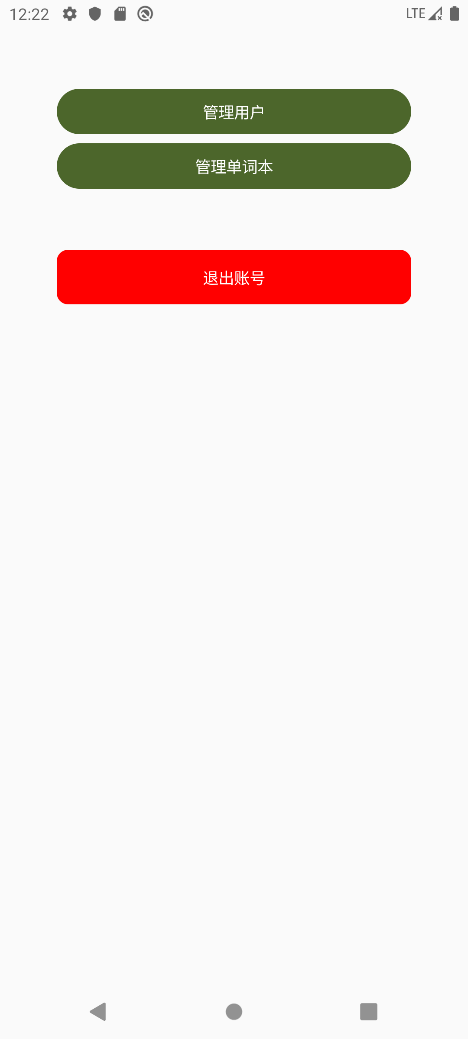


# 5 系统测试

## 5.1 功能测试

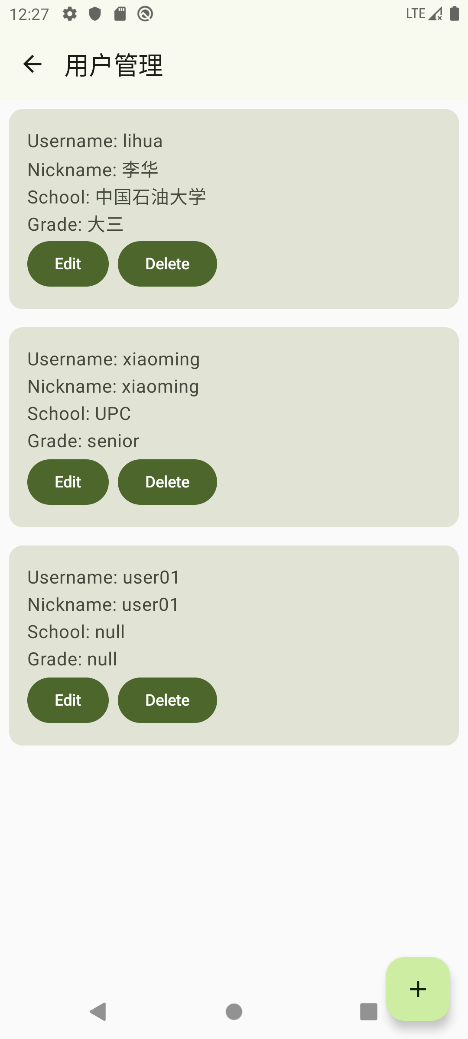
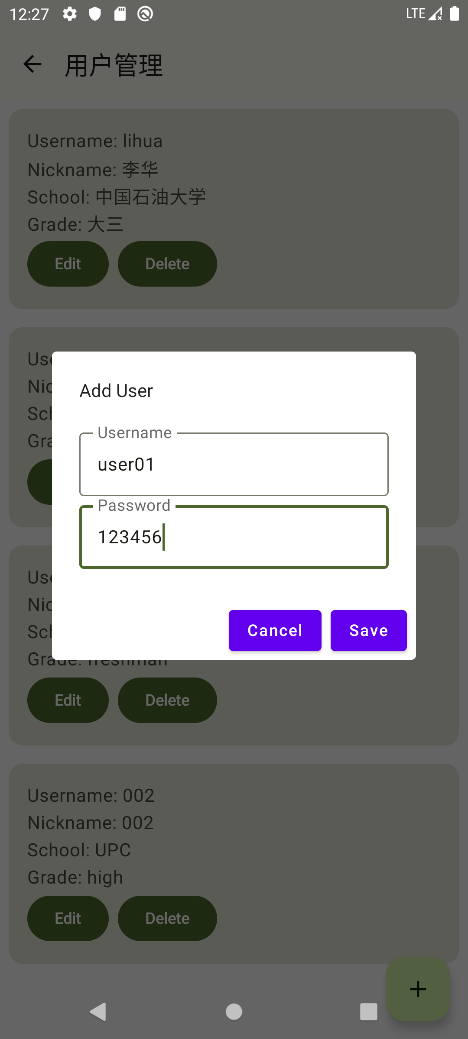
### 5.1.1管理端

1. **管理员登录**

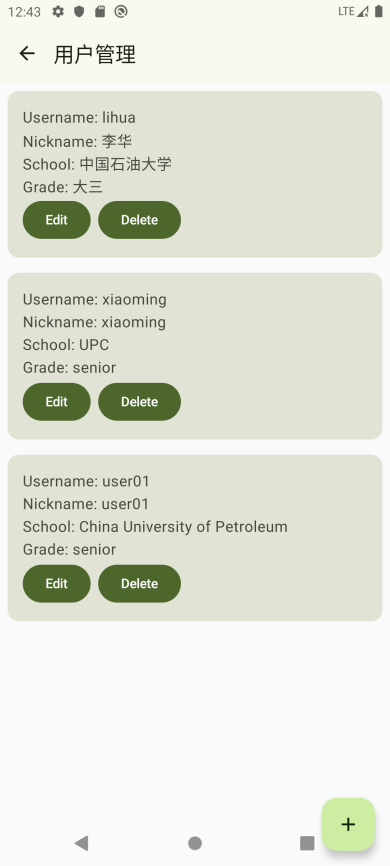
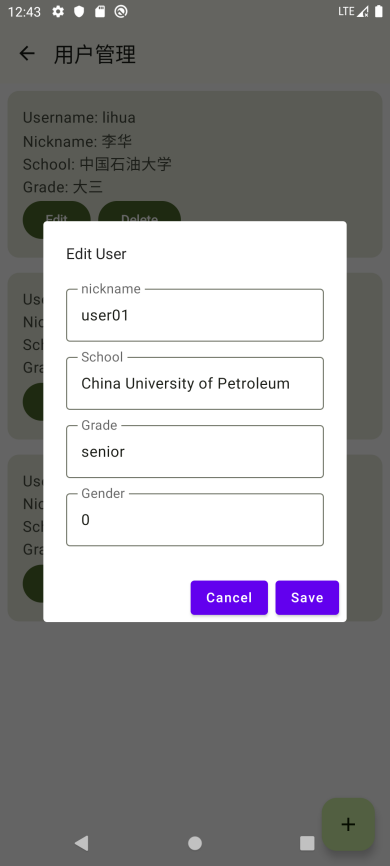


登录成功。

1. **用户管理**

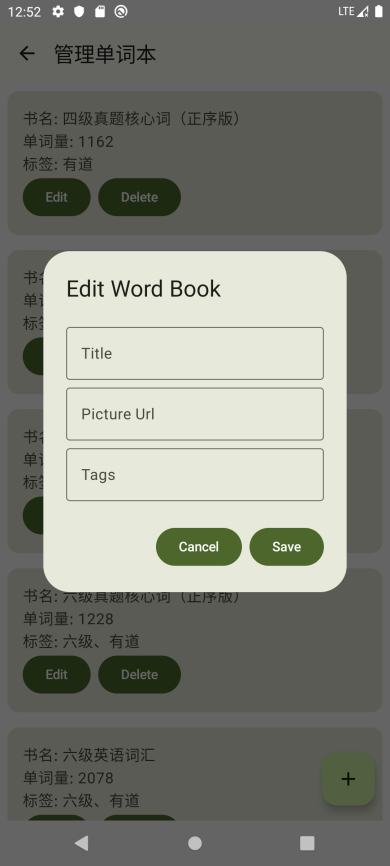
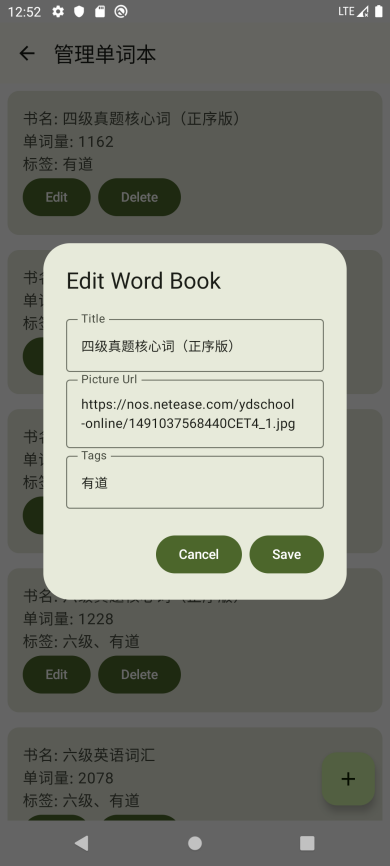


添加用户成功。

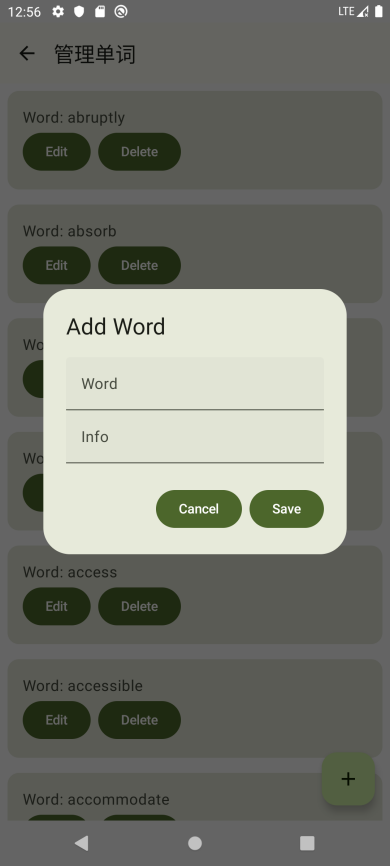
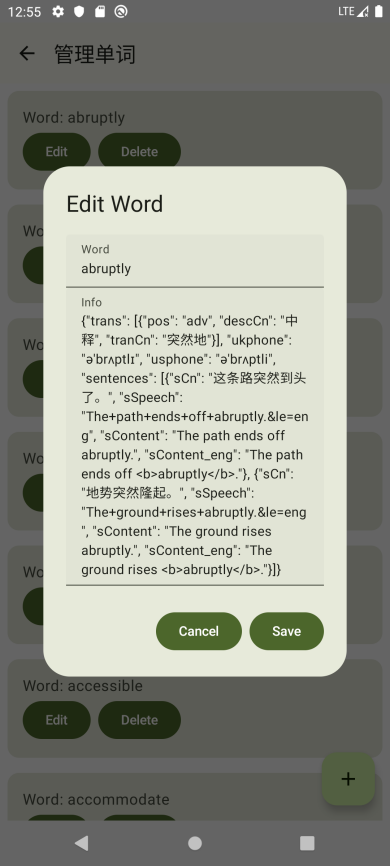
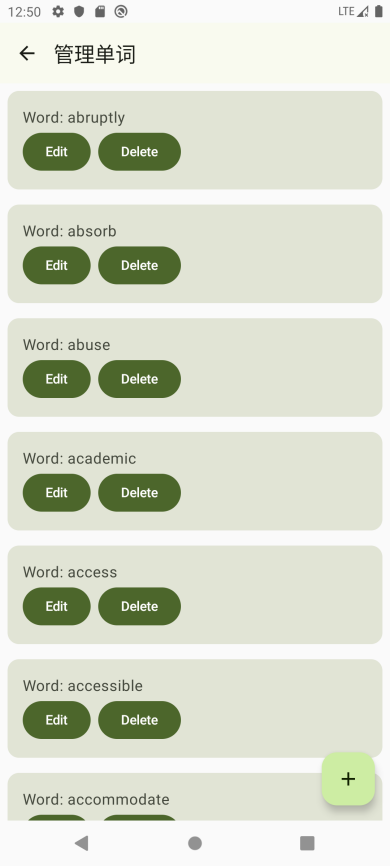


用户信息编辑成功

1. **单词管理**



单词书管理成功



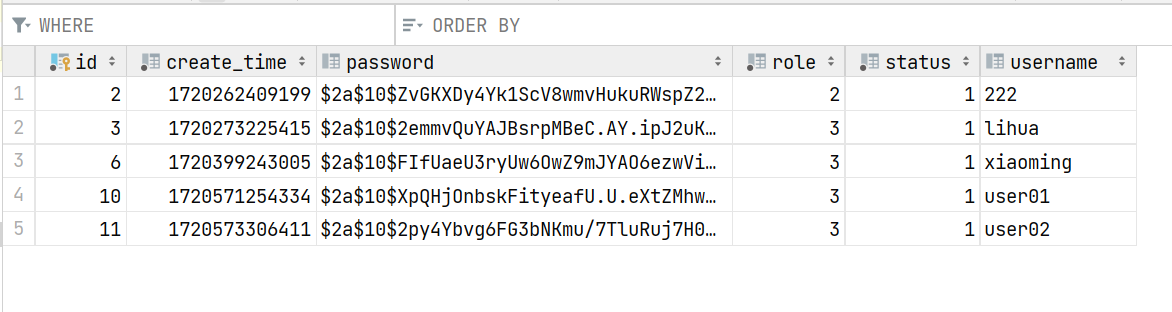
单词管理成功

### 5.1.2用户登录/注册



用户登录成功





用户注册成功

### 5.1.3设定学习计划



学习计划设定成功

### 5.1.4 单词学习



单词学习成功

### 5.1.5 单词复习



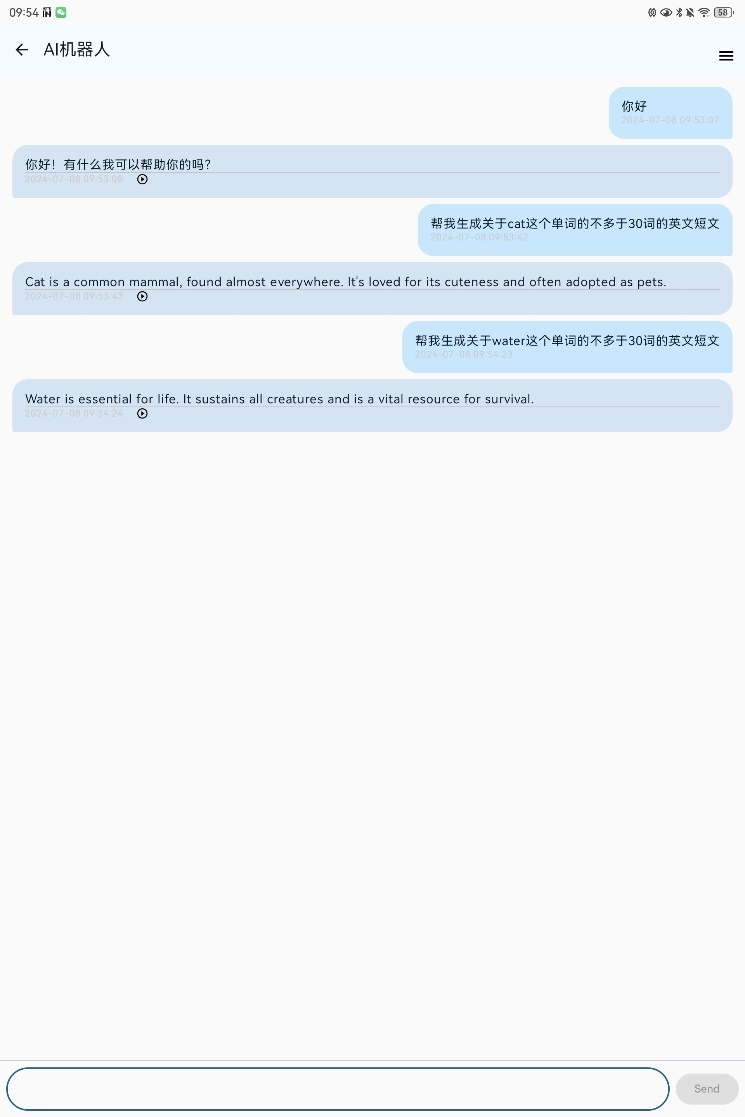
复习单词成功

### 5.1.6 学习数据统计



数据统计成功

### 5.1.7 AI对话与语音合成



**语音合成**

AI对话与语音合成成功

## 5.2 性能测试

### 1. 测试目的

确保基于大模型的英语水平提升系统在高负载下保持稳定运行，并满足性能要求。主要关注响应时间、系统吞吐量和资源消耗。

### 2. 测试对象

* 文本和语音生成
* 用户管理功能
* 学习数据统计

### 3. 性能测试类型

* **基准测试**：在标准负载下测定系统的性能基线。
* **压力测试**：确定系统的最大运行能力，并观察其在极端负载下的表现。
* **容量测试**：评估系统支持的最大用户数量和数据处理量。

### 4. 性能指标

* **响应时间**：系统响应用户请求的时间。
* **系统吞吐量**：系统单位时间内能处理的请求量。
* **资源利用率**：CPU、内存及网络资源的使用情况。

### 5. 测试工具

* **Apache JMeter**：用于模拟多用户并发访问，执行性能测试。
* **Prometheus与Grafana**：监控系统资源使用情况和性能指标。

### 6. 测试场景

#### 场景1：单用户基本操作

* **目的**：测量一个用户执行登陆、生成短文、查看学习数据时的性能。
* **步骤**：
  1. 用户登录。
  2. 生成一篇包含特定单词的短文。
  3. 播放语音。
  4. 查看学习进度和统计数据。
* **结果**：所有操作的响应时间不超过2秒。

#### 场景2：多用户并发访问

* **目的**：评估多个用户同时使用系统时的性能表现。
* **步骤**：
  1. 50个并发用户登录系统。
  2. 同时生成短文和语音。
  3. 同时查看学习数据。
* **结果**：系统无崩溃，平均响应时间不超过5秒。

#### 场景3：持续运行测试

* **目的**：评估系统连续运行24小时的性能稳定性。
* **步骤**：
  1. 系统连续运行，定时执行登录、生成内容和数据查询操作。
  2. 监控系统资源消耗。
* **结果**：没有内存泄漏，性能不下降。

## 5.3 其它测试

### 5.3.1可用性测试

评估用户在使用应用过程中的体验，确保应用易于使用，符合用户期望。

* **用户任务分析**：模拟用户完成特定任务，如创建学习计划、复习单词等。
* **用户界面简洁性**：确保界面不过于复杂，用户能快速理解如何操作。
* **错误消息和帮助文档**：测试错误处理和用户帮助指南的有效性。

### 5.3.2兼容性测试

确保应用在所有支持的设备、操作系统版本和网络环境中正常运行。

* **操作系统版本**：Android 的不同版本。
* **设备类型**：包括手机、平板等。
* **网络环境**：在Wi-Fi、4G和较差网络条件下的表现。

### 5.3.3安全性测试

识别和修复可能的安全漏洞，保护系统和用户数据的安全。

* **身份验证**：测试系统如何处理各种身份验证尝试。
* **数据加密**：验证敏感数据（如密码和用户信息）的加密实现。
* **注入攻击**：确保系统对SQL注入、脚本注入等攻击有充分防护。

# 6 项目总结

### 项目背景与动机

随着全球化进程的加速和在线教育的兴起，掌握英语成为了全球公民的必备技能。在这样的背景下，我们的团队利用人工智能尤其是大语言模型的力量，开发了一套英语水平提升系统。这个系统旨在提供一个个性化、互动性强的英语学习平台，通过最新的技术帮助用户有效提高英语能力。

### 技术概述

项目的技术栈涵盖了前端到后端，以及核心的大模型处理：

* **前端开发**：采用Android平台开发，结合Material Design，为用户提供了一个响应迅速且友好的用户界面。前端不仅支持用户的日常操作，还实现了动态的数据展示，如学习进度条和成绩图表。
* **后端架构**：后端采用Spring Boot，优化了服务的响应时间和处理效率。利用其广泛的库支持，我们快速实现了多种功能，包括用户管理、数据处理和接口整合。
* **大模型集成**：集成了百度的千帆大模型（ERNIE-Speed-128K），该模型支持高效的文本和语音生成，极大地丰富了学习内容和形式。我们通过自定义算法调整模型，以适应教育场景，确保生成内容的教育适用性和多样性。
* **数据库技术**：使用MySQL进行数据存储，设计了严谨的数据库结构来存储用户信息、学习记录和系统日志。我们还实现了高效的数据查询和更新机制，确保系统的稳定性和数据的安全性。

### 功能实现

本系统实现了以下核心功能：

* **文本和语音生成**：用户可以根据自己的需求，生成包含特定单词或主题的英文短文和对应语音，帮助学习和练习。
* **用户管理**：系统提供完整的用户管理功能，包括用户注册、登录、信息编辑等。
* **数据记录与分析**：系统自动记录用户的学习活动，并通过智能分析，为用户提供学习建议和进度反馈。
* **学习进度可视化**：通过图表和进度条，直观展示用户的学习成果和进度，增强学习动力。

### 项目挑战与解决方案

在项目开发过程中，我们面临了多个技术和管理挑战：

* **技术集成**：将百度千帆大模型与现有系统集成时遇到了性能调优问题。通过多次测试和调整模型参数，最终确保了模型的高效运行。
* **用户体验**：为了提高用户体验，我们对APP的交互设计进行了多轮迭代，确保用户操作的直观性和流畅性。
* **数据安全**：在用户数据处理和存储方面，我们实施了多层安全措施，包括数据加密和安全访问控制，以防止数据泄露和未授权访问。

### 结论与未来展望

通过这个项目，我们的团队不仅提高了技术能力，也加深了对教育技术应用的理解。未来，我们计划引入更多AI和机器学习算法，进一步提升系统的智能化水平，为用户提供更加丰富和定制的学习体验。同时，我们也希望将系统扩展到其他语言学习领域，服务更广泛的用户群体。