目 录

[第一章 需求分析 3](#_Toc6230586)

[第二章 概要设计 3](#_Toc6230587)

[第三章 详细设计 3](#_Toc6230588)

[第四章 测试报告 3](#_Toc6230589)

[第五章 安装及使用 3](#_Toc6230590)

[第六章 项目总结 3](#_Toc6230591)

# 需求分析

1. **项目背景**

随着全球化的不断深入，各国之间社会、经济、文化和科技领域交流不断深入，对于当代人才的语言要求越来越高。而英语作为世界上流通最广泛的语言，学好英语就相当于多了一份眼界和格局。在英语中，人们日常使用最多的就是英语口语，故其重要性不言而喻。  
 现在市场上存在的英语口语练习软件较少，且都是一个独立的APP，难免会为用户推送一些与学习口语无关的内容，使得用户难以专心学习。而本项目中所要做的英语口语练习微信小程序，依托微信开发者平台和科大讯飞开放平台，旨在打造一款轻量化的学习工具，使得用户获得轻松、高效的学习体验。

1. **应用场景**

本项目的市场定位是一款免费的口语练习微信小程序，轻量化的程序和依托于微信的特点决定了该程序易于被大众接受，拥有可观的用户群体。用户可以利用本小程序轻而快的特点得到较好的英语口语学习体验。应用场景主要为学校、公司、交通工具等地点。应用时间主要是零散的碎片化时间。

1. **目标人群**

该程序的目标客户群主要为学生和有练习英语口语需求的商务人士，以及对英语口语感兴趣，想要学习的社会人士。

1. **主要功能**

本项目希望通过利用小程序这一轻量化的载体和语音测评这一先进的技术去承载更具有价值的知识。用户完全可以通过此小程序扩展英语练习的维度——即不再是最基本的记忆单词，而是通过跟读或自主阅读的形式做到听与说兼备。同时通过语音测评的功能给用户反馈，让用户可以更好地练习口语发音；通过数据统计让用户更加直观地了解自己口语方面的不足，加以改进，真正做到一款利于用户提升口语能力的小程序。

1. **本项目与市面同类别APP比较:**

|  |  |  |  |
| --- | --- | --- | --- |
| **软件名称** | **本软件** | **雅思哥** | **英语流利说** |
| **软件形式** | 微信小程序 | APP | APP |
| **占用空间大小** | 无需安装 | 48.7M | 51.1M |
| **所包含单词列表** | 四六级、雅思 | 无 | 无 |
| **所包含句子列表** | 四六级、雅思等 | 无 | 无 |
| **所包含相关课程** | 包含标准读音句子课程 | 包含雅思考试相关课程 | 包含从低年级至高年级众多课程 |
| **是否包含读音评分** | 包含流畅度评分，完整度评分，准确度评分以及漏读提醒 | 仅包含整体评分以及漏读提醒 | 仅包含整体评分以及漏读提醒 |
| **是否包含标准读音** | √ | √ | √ |
| **是否包含各种直播及课程推销** | × | √ | √ |
| **是否包含交流论坛** | × | √ | × |

# 概要设计

**1.总体结构设计：**

本项目访问层，即前端交互为微信小程序；功能层为可与用户交互供其实用的各功能模块；标准层，即后端，为springboot项目，后端持久层选用mybatis框架，WEB应用服务器选用springboot默认内置的Tomcat。项目后端部署在服务器上。

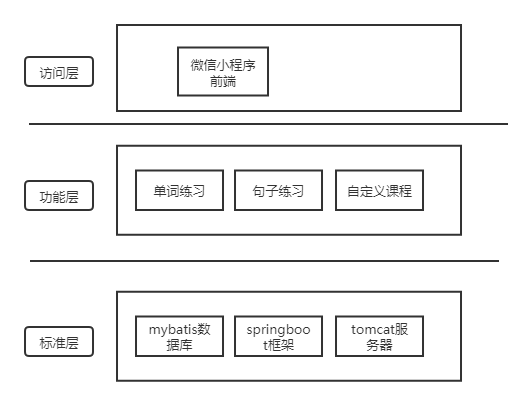


图2-1 项目总体架构设计

**2.功能模块简介：**

（1）**英语口语练习功能**，使用语音测评功能对用户的读音进行评测。

（2）**课程选择功能**，用户可根据自己需求选择对应的课程并自定义学习计划量。

（3）**签到打卡功能。**

（4）**数据统计功能**，将用户历史得分和平台平均得分进行图表展示。

（5）**排行榜功能**，即将使用小程序所有用户的平均分分数进行排名，可以清晰看到自己所处的位置，激励用户多加练习。

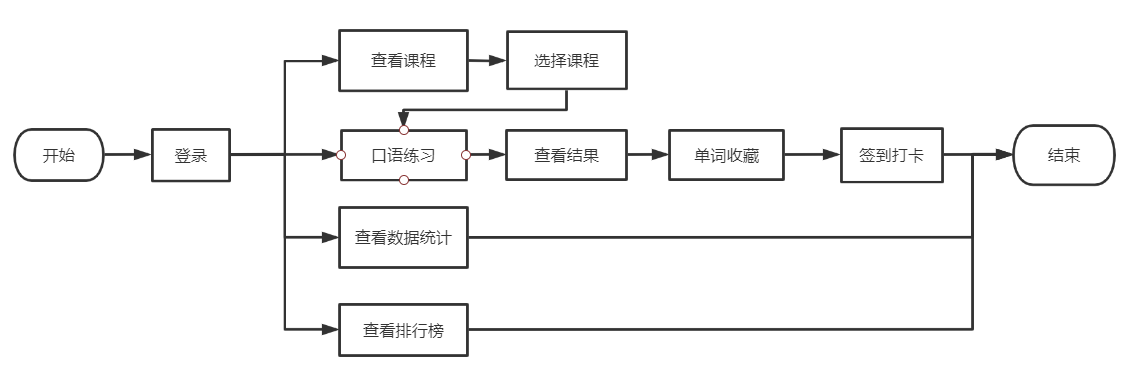


图2-2程序运行流程图

# 详细设计

**1.关键技术介绍：**

针对第一章所列出的对比表格，本项目进行了一定程度的改进。本项目采用了讯飞开放平台语音测评功能。此功能可将上传的英语语音与其对应的内容进行匹配和测评，在较短的时间内返回语音阅读的各项指标的json格式数据：如准确度得分、完整度得分、流畅度得分，以及句子中各单词得分，有效地解决了现在市面上的口语练习小程序评分标准模糊、宽泛、没有较好的指导作用等问题。同时通过对数据的全面展示，方便用户找出自己的弱项并加以练习，使用小程序这一较为轻量化的工具，实现碎片化时间的有效利用。

1. **界面设计：**
   1. **登录界面（主界面）**



该界面为用户第一次使用小程序时的登录界面。在初始情况下，该界面上方为默认背景，并且有微信小程序的名称。在背景图片的下方显示有已学天数、连续打卡和剩余时间栏。此时由于用户未登录，默认置为0。在界面的中下部为赞赏支持和意见反馈按钮，用户在登陆后可以点击这两个按钮对本小程序的进一步优化提供意见。界面底部为点击登录按钮。用户通过点击此按钮进行微信小程序的授权，然后完成登录。最下方的导航栏三个按钮分别为主页、单词列表和统计，登陆后点击按钮便可以进入相应界面。

图3-1 登录界面



该界面为用户登录后的原始界面状态，同时也是小程序的主页。在界面的上部有通过用户授权得到的用户微信头像，加上橙色的背景图，共同作为用户的一个个人名片。在界面中部显示的是用户的已学天数、连续打卡和剩余时间，单位都为天数。在界面的中下部为赞赏支持和意见反馈按钮，用户可以点击这两个按钮对本小程序的进一步优化提供意见。对于已经选择了课程的用户可以点击界面底部的开始学习按钮进直接入练习界面，而没有选择课程的用户点击开始学习按钮会进入课程模块以便用户选择课程。最下方的导航栏三个按钮分别为主页、单词列表和统计，登陆后点击按钮便可以进入相应界面。

图3-2 主界面

1. **课程模块界面**

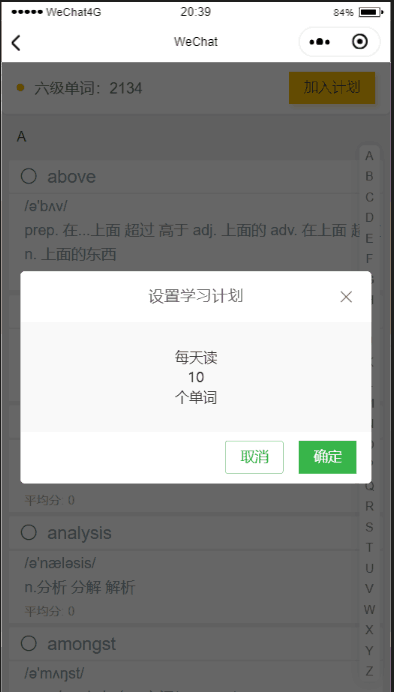
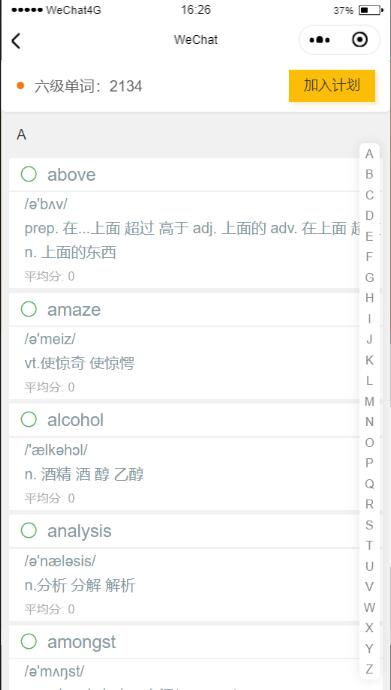


图3-3 课程列表 图3-4 课程详情 图3-5 设置计划量

图3-3为课程列表，该界面也为课程模块的主界面。该界面的最上方为搜索栏，用户可以通过搜索栏搜索自己想要学习的课程和其他信息。在这个界面，用户可查看所有单词和句子、自己收藏的单词和句子以及所有课程。用户点击具体课程可进入图3-4课程详情查看该课程单词。点击右上角加入课程可设置每日学习计划量，再添加到自己的学习计划中。

1. **练习界面**



图3-6 单词练习界面 图3-7 句子练习界面

该界面为英语口语练习小程序的主要功能性界面。界面中最上方部分为需要练习的单词或句子的详情，单词的话包括有单词本身、单词音标以及该单词常用的中文含义。如果是练习一段句子，则会显示该句及其中文翻译。下方分别有：“标准读音”按钮、上一次用户发音的“我的发音”按钮，以及下方的麦克风按钮用以录下用户发音。对于返回结果，我们会将本次用户得分、历史用户得分以及平台得分都记录下来，对于句子，我们会用红色标注出用户读音较差的单词。

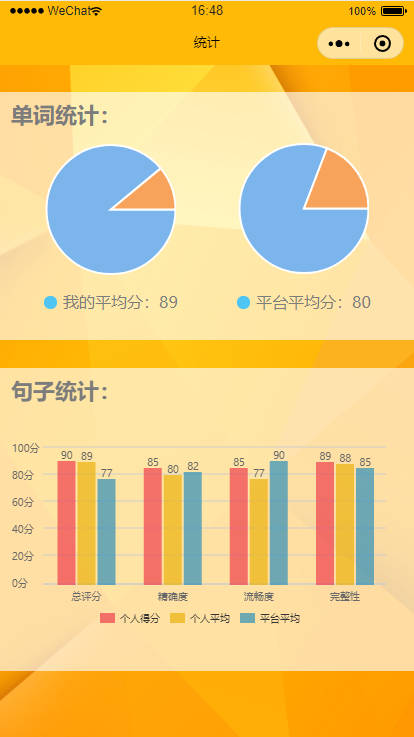
1. **签到界面**



签到界面为用户完成了每天的计划量的练习之后自动弹出的一个界面。整个界面使用明黄色的背景和整个小程序的风格保持一致。签到界面上方为用户名称，界面中部为连续打卡和累计打卡天数。用户可以通过点击签到按钮进行当日的签到打卡。

图3-8 签到界面

1. **统计界面**



该界面用户可以通过点击主界面中的统计按钮进入，主要目的是让用户对于自己得分的情况有一个详细的了解。上方饼状图统计了用户、平台单词发音得分；下方柱状图统计了用户、平台句子发音得分，使用户可以直观地看到自己的弱项。

图3-9 统计界面

**（6）排行榜界面**



排行榜界面主要是为了给全平台的用户一个比较、展示的机会。用户可以通过不断练习跻身排行榜的上游获得成就感和荣誉感。同时也可以激励用户不断学习、练习来提高自己的英语水平。排行榜的前三名分别使用金银铜色作为列表背景，更有仪式感。

图3-10 排行榜界面

1. **功能模块设计**
2. **口语练习功能模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| 口语练习功能 | 单词练习功能 | 此功能是根据用户选择的单词练习课程进行单词的口语训练。小程序显示单词和音标，用户点击开始按钮进行练习，并显示得分情况。 | 高 |
| 句子练习功能 | 此功能是根据用户选择的句子练习课程进行单词的口语训练。小程序显示句子，用户点击开始按钮进行练习，并显示得分情况，并且还会根据句子中每个单词的单独得分标注出低于平均分的单词，对漏读的单词也进行单独的标注。 | 高 |
| 文章跟读功能 | 此功能根据用户选择的文字跟读课程进行文字跟读的训练，小程序显示文章以及标注出当前跟读的进度，用户点击开始按钮进行跟读，小程序显示跟读的得分并对低于平均分的单词进行标注。 | 中 |

1. **课程功能模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| 课程功能模块 | 显示课程列表 | 用户点击小程序底部导航栏“课程”进入课程列表查看所有课程 | 高 |
| 显示课程详情 | 点击课程列表中的课程，即可进入课程详情界面。课程详情界面显示课程名称、难度、词汇量等具体信息 | 高 |
| 添加课程 | 用户可在课程详情界面点击“添加到我的课程”来将此课程添加到用户学习课程列表中 | 中 |
| 我的课程 | 通过进入“用户中心”的“我的课程”界面，用户可以查看当前已选择的所有课程 | 中 |

1. **签到打卡功能模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| 签到打卡功能模块 | 签到 | 用户完成当日学习计划量后，可以进行打卡。 | 中 |

1. **统计功能模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| 统计功能模块 | 查看平台与个人历史数据 | 用户点击小程序底部导航栏“统计”进入统计界面，在其中可以查看自己各项指标的平均分以及平台平均分 | 中 |

1. **排行榜功能模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| 排行榜功能模块 | 查看平均得分排行榜 | 用户点击小程序底部导航栏“排行榜”可以查看小程序用户平均得分总榜，并且可以知道自己所处位置。 | 中 |

**4.数据库设计**

**（1）ER图**

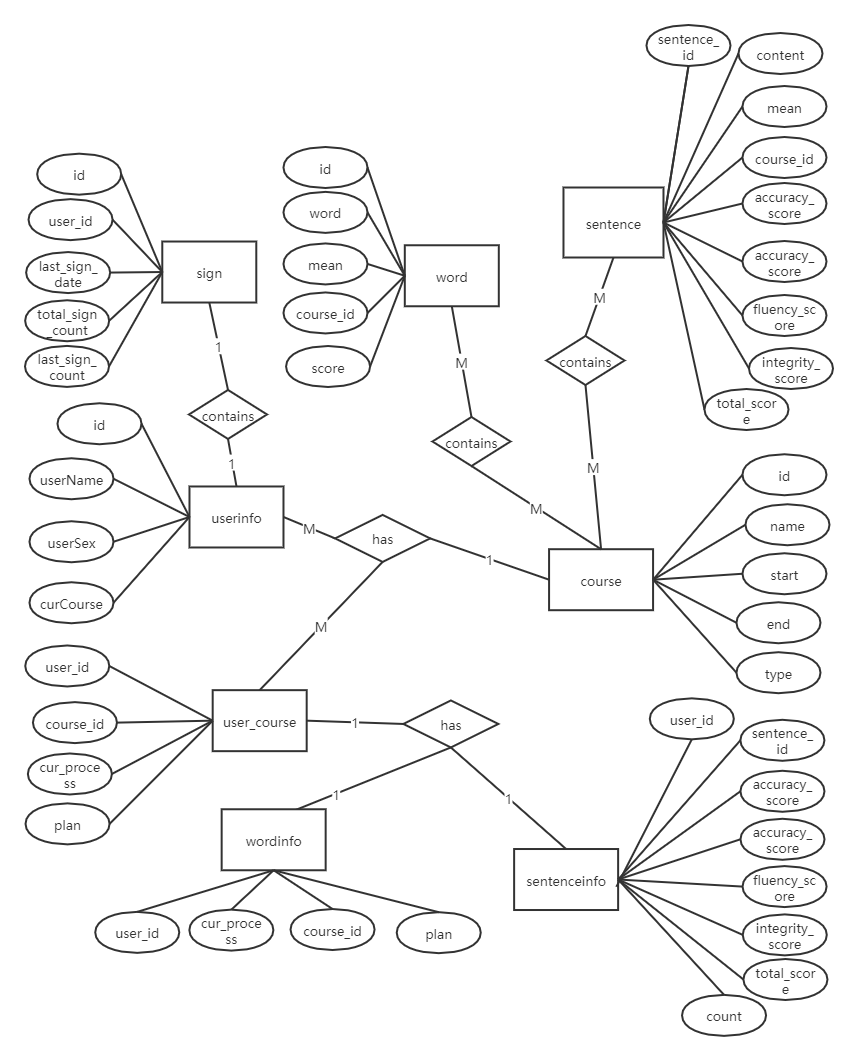


图4-1 数据库ER图

1. **数据库设计基本信息：**

本系统数据库管理系统为Mysql 5.7.2，后端数据库框架使用Mybatis。数据库表组成如下：

|  |  |  |
| --- | --- | --- |
| 序号 | 表名 | 功能说明 |
| 1 | userinfo | 用户基本信息数据表 |
| 2 | course | 课程基本信息数据表 |
| 3 | user\_course | 用户选择的当前课程数据表 |
| 4 | sign\_in | 用户签到数据表 |
| 5 | word | 单词数据表 |
| 6 | word\_info | 用户所学单词数据表 |
| 7 | sentence | 句子数据表 |
| 8 | sentence\_info | 用户所学句子数据表 |

1. **数据库各表设计：**

**1).用户信息表(userinfo)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 编号 | id | varchar(100) | 非空，主键 |
| 2 | 用户名 | userName | varchar(20) |  |
| 3 | 用户性别 | userSex | varchar(10) |  |
| 4 | 电子邮件 | email | varchar(20) |  |
| 5 | 简介 | intro | varchar(255) |  |
| 6 | 教育 | education | varchar(10) |  |
| 7 | 当前课程 | curCourse | int(10) |  |

**2).课程信息表(course)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 编号 | id | int(11) | 非空，主键 |
| 2 | 姓名 | name | varchar(20) | 非空 |
| 3 | 课程图片 | course\_image | varchar(400) |  |
| 4 | 难度 | difficulty | int(5) |  |
| 5 | 数量 | count | int(11) |  |
| 6 | 类型 | type | int(1) |  |
| 7 | 简介 | intro | varchar(400) |  |
| 8 | 起始位置 | start | int(11) |  |
| 9 | 结束位置 | end | int(11) |  |

**3).用户\_课程信息表(user\_course)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 用户编号 | user\_id | varchar(100) | 非空，联合主键 |
| 2 | 课程编号 | course\_id | int(11) | 非空，联合主键 |
| 3 | 当前进度 | cur\_process | int(5) |  |
| 4 | 计划 | plan | int(3) |  |

**4).签到信息表(sign\_in)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 编号 | id | int(10) | 非空，主键 |
| 2 | 用户编号 | user\_id | varchar(100) | 非空 |
| 3 | 最后签到日期 | last\_sign\_date | data |  |
| 4 | 总共签到次数 | total\_sign\_count | int(5) |  |
| 5 | 最近连续签到次数 | last\_sign\_count | int(5) |  |

**5).单词表(word)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 编号 | id | int(11) | 非空，主键 |
| 2 | 单词 | word | varchar(20) | 非空 |
| 3 | 中文 | chinese | varchar(20) | 非空 |
| 4 | 课程编号 | course\_id | int(11) | 非空 |
| 5 | 分数 | score | decimal(5,2) |  |
| 6 | 种类 | flag | int(1) |  |
| 7 | 音频路径 | video\_path | varchar(255) |  |
| 8 | 次数 | count | int(5) |  |

**6).单词信息表(word\_info)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 用户编号 | user\_id | varchar(100) | 非空，联合主键 |
| 2 | 单词编号 | word\_id | int(11) | 非空，联合主键 |
| 3 | 总分 | total\_score | decimal(5,2) |  |
| 4 | 次数 | count | int(5) |  |

**7).句子表(sentence)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 编号 | id | int(20) | 非空，主键 |
| 2 | 内容 | content | varchar(400) | 非空 |
| 3 | 中文 | chinese | varchar(400) |  |
| 4 | 课程编号 | course\_id | int(11) |  |
| 5 | 准确度得分 | accuracy\_score | decimal(5,2) |  |
| 6 | 流畅度得分 | fluency\_score | decimal(5,2) |  |
| 7 | 完整度得分 | integrity\_score | decimal(5,2) |  |
| 8 | 总分 | total\_score | decimal(5,2) |  |
| 9 | 种类 | flag | int(1) |  |
| 10 | 音频路径 | video\_path | varchar(255) |  |
| 11 | 各单词得分 | word\_score | varchar(1000) |  |
| 12 | 次数 | count | int(5) |  |

**8).句子信息表(sentence\_info)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **中文描述** | **列名** | **类型和精度** | **数据说明** |
| 1 | 用户编号 | user\_id | varchar(100) | 非空，联合主键 |
| 2 | 句子编号 | sentence\_id | int(20) | 非空，联合主键 |
| 3 | 准确度得分 | accuracy\_score | decimal(5,2) |  |
| 4 | 流畅度得分 | fluency\_score | decimal(5,2) |  |
| 5 | 完整度得分 | integrity\_score | decimal(5,2) |  |
| 6 | 总分 | total\_score | decimal(5,2) |  |
| 7 | 次数 | count | int(5) |  |

# 测试报告

【填写说明：包括测试报告和技术指标。为了保证作品质量，建议多进行测试，并将测试过程、测试结果、修正过程或结果形成文档，也可以将本标题修改为主要测试，撰写主要测试过程结果及其修正；根据测试结果，形成多维度技术指标，包括：运行速度、安全性、扩展性、部署方便性和可用性等。本部分简要说明即可，减少常识性内容。】

1. **接口测试(基本功能测试）**

为保证**后端逻辑运行**正常且正确返回，采用Postman对后端进行接口测试。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **访问URL** | **接口描述** | **请求类型** | **参数列表** | **返回是否正确** | **错误原因** | **是否改正** |
| localhost:8080/login | 用户登录接口 | POST | code,userName,userSex,iv,encrypterData | 否 | 后端中访问微信登录后台时，appid与小程序appid未对应。 | 是 |
| localhost:8080/getMyCourse | 用户获取当前已选课程 | GET | user\_id,course\_id | 是 |  |  |
| localhost:8080/getCourseList | 用户获取课程列表 | GET | 无 | 是 |  |  |
| localhost:8080/getAllWord | 获取所有单词 | GET | 无 | 是 |  |  |
| localhost:8080/getWordList | 获取指定课程的所有单词 | GET | course\_id | 是 |  |  |
| localhost:8080/getWordCollect | 获取用户收藏的单词 | GET | user\_id | 否 | SQL语句有误 | 是 |
| localhost:8080/selectCourse | 用户选择课程 | POST | user\_id,course\_id,plan | 是 |  |  |
| localhost:8080/getAllSentence | 获取所有句子 | GET | 无 | 否 | 后端通过首字母对句子进行分组时，按照小写ASCII进行分组。 | 是 |
| localhost:8080/getSentenceList | 获取指定课程的所有句子 | GET | course\_id | 否 | 后端通过首字母对句子进行分组时，按照小写ASCII进行分组。 | 是 |
| localhost:8080/getSentenceCollect | 获取用户收藏的单词 | GET | user\_id | 是 |  |  |
| localhost:8080/sign | 用户签到接口 | POST | user\_id | 否 | 前后端日期格式不一致 | 是 |
| localhost:8080/getRank | 获取排行榜 | GET | user\_id | 是 |  |  |
| localhost:8080/getStat | 获取统计数据 | GET | user\_id | 否 | 后端处理数据时出现除数为0的异常 | 是 |

1. **压力测试（性能测试）**

本项目虽不涉及高并发的情形，但需要保证一般情况下的多人访问正常。压力测试采用Postman压测功能，对日常使用较为耗时的功能进行压测，评判标准为请求到回复的时间以及返回结果的正确与否。

压测请求迭代次数设置为20，使用tests脚本对状态码、响应速度、返回字段进行监控。访问的后端接口已部署到腾讯云服务器，网络带宽为1M，已使用SSL证书进行加密。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **压测URL** | **状态码是否为200** | **响应平均速度** | **返回字段是否正确** | **错误/不足原因** | **改进措施** |
| https://scunewwave.xyz:8080/getALLWord | 是 | >3s | 是 | 由于返回的单词列表大概有4000个，且一个单词包含较多数据，因此响应速度过慢 | 对单词列表进行了分页请求与显示，最终响应速度可以降至1s以内。 |
| https://scunewwave.xyz:8080/  getALLSentence | 是 | 500ms | 是 |  |  |
| https://scunewwave.xyz:8080/  getCourseList?course\_id=1 | 是 | 100ms | 是 |  |  |
| https://scunewwave.xyz:8080/  practice/user\_id=og7kd5HTQjjRn5TimXB0dXyXX1As&&word\_id=1&&multiFormdata=1.wav | 是 | <5s | 是 | 此处是用户进行语音测评功能接口，需要向后端传输录音文件，并且将录音和内容再次上传到讯飞平台，因此时间较长，也是本项目需要时间最长的功能 | 受制于云服务器网络带宽，因此此问题可以通过增加云服务器配置解决。 |
| https://scunewwave.xyz:8080/ |  |  |  |  |  |

# 安装及使用

本项目基于微信小程序平台开发，后端已上传至云服务器并已注册、备案正规域名开放后端访问接口，因此不需要使用者进行繁琐的程序安装。

**运行环境要求：**安卓和IOS系统，微信版本至少需要更新至7.0.7，其以上更佳。网络连接正常。

**使用流程：**

1. 微信小程序运行方式十分简便，在小程序搜索框搜索或者扫码即可运行。
2. 打开小程序后，需要进行授权。点击首页下方“点击登录”按钮进行授权和登录。
3. 登录完成后，可以进入课程选择界面进行课程查看和选择。
4. 进入到口语练习界面，屏幕上方显示内容和翻译，屏幕下方可以听标准读音，接 着按住“麦克风”按钮可以进行阅读，松开后等待口语测评接口返回结果，并可以按“我 的读音”按钮听自己的发音。
5. 完成设定的计划量的阅读后，可以进行签到。
6. 用户可以在“统计”界面查看自己历史得分、平台平均得分等数据。

# 项目总结

1. **.项目优缺自查：**

本项目解决的问题比较切合实际，即针对我们学习英语时口语方面的欠缺而设计。采用的语音测评技术可以在很大程度上弥补现阶段各口语APP评分不详细的弱点。同时小程序这个载体应用面广，并且实用起来十分简便，可以在碎片化的时间为用户带来便捷、高效的使用体验。在基础功能上，本项目完成度较高。

但由于小组比较缺乏实际开发经验，因此在性能或防错方面没有做到最好。现阶段数据库中数据不算庞大，但如果使用人数增多或者课程内容增加，那么时间性能是一个亟待提升的问题之一；受制于小程序的规定，本项目在商业推广上有一定局限性。

1. **.后续升级思路：**

性能方面，可以考虑加入缓存来提高响应性能。在数据库数据过大时，可以对数据库进行优化，提升查询的性能。

解决商业推广问题，可以将小程序作为手机APP引流的工具之一，在APP上实现商业价值。

1. **.总结：**

项目致力于通过轻量化的工具和先进的技术来承载更有价值的内容。口语一直是我国英语教较为缺失的一部分，也是英语学习中比较有难度的一部分，因此通过本项目可以提高口语学习在大家英语学习中的效率和比重。在项目中，我们通过比较标准的开发流程去进行项目开发，在迭代中不断找出之前的不足，提升了工程能力；在沟通中找到更多想法，锻炼了交流能力；在试错与改错中，发现了基础知识的欠缺点。项目尚有一些问题，但我们会不断地完善，真正成为英语学习者手中的一把“利器”。