文件编号：Quickman-SWC2020-[TEAMNUMBER]

受控状态：■受控 □非受控

保密级别：□公司级 □部门级 ■项目级 □普通级

采纳标准：CMMI DEV V1.2





“妙记”自动笔记系统

**Auto Notes**

项目开发文档

**Version 1.0.1**

2019.11.15

**Written by Quickman**



**All Rights Reserved**

目录

[1 项目概述 1](#_Toc24905102)

[1.1 项目背景 1](#_Toc24905103)

[1.2 项目定位 1](#_Toc24905104)

[1.2.1 应用场景 1](#_Toc24905105)

[1.2.2 目标用户 2](#_Toc24905106)

[1.3 项目方案 2](#_Toc24905107)

[1.4 项目目标 2](#_Toc24905108)

[1.5 项目价值 2](#_Toc24905109)

[2 开发计划 2](#_Toc24905110)

[2.1 最终呈现形式 2](#_Toc24905111)

[2.2 主要功能描述 3](#_Toc24905112)

[2.3 运行环境 4](#_Toc24905113)

[2.4 验收标准 4](#_Toc24905114)

[2.5 关键问题 4](#_Toc24905115)

[2.6 进度安排 5](#_Toc24905116)

[2.7 开发预算 5](#_Toc24905117)

[3 可行性分析 5](#_Toc24905118)

[3.1 技术可行性分析 5](#_Toc24905119)

[3.2 资源可行性分析 6](#_Toc24905120)

[3.3 市场可行性分析 6](#_Toc24905121)

[4 需求分析 9](#_Toc24905122)

[4.1 数据需求 9](#_Toc24905123)

[4.1.1 静态数据 9](#_Toc24905124)

[4.1.2 动态数据 9](#_Toc24905125)

[4.1.3 数据词典 9](#_Toc24905126)

[4.1.4 数据采集 9](#_Toc24905127)

[4.2 功能需求 10](#_Toc24905128)

[4.2.1 文字识别功能模块 11](#_Toc24905129)

[4.2.2 文本学习功能模块 12](#_Toc24905130)

[4.2.3 词义感知功能模块 13](#_Toc24905131)

[4.3 性能需求 14](#_Toc24905132)

[4.3.1 时间特性 14](#_Toc24905133)

[4.3.2 适应性 14](#_Toc24905134)

[4.4 界面需求 15](#_Toc24905135)

[4.5 接口需求 17](#_Toc24905136)

[4.5.1 硬件接口 17](#_Toc24905137)

[4.5.2 软件接口 17](#_Toc24905138)

[4.6 其他需求 18](#_Toc24905139)

[4.6.1 可移植性 18](#_Toc24905140)

[4.6.2 可分享性 18](#_Toc24905141)

[5 概要设计 19](#_Toc24905142)

[5.1 处理流程 19](#_Toc24905143)

[5.2 总体结构设计 19](#_Toc24905144)

[5.3 功能设计 19](#_Toc24905145)

[5.4 用户界面设计 19](#_Toc24905146)

[5.5 数据结构设计 19](#_Toc24905147)

[5.6 接口设计 19](#_Toc24905148)

[5.6.1 外部接口 19](#_Toc24905149)

[5.6.2 内部接口 19](#_Toc24905150)

[5.7 错误/异常处理设计 19](#_Toc24905151)

[5.7.1 错误/异常输出信息 19](#_Toc24905152)

[5.7.2 错误/异常处理对策 19](#_Toc24905153)

[5.8 系统配置策略 19](#_Toc24905154)

[5.9 系统部署方案 19](#_Toc24905155)

[5.10 其他相关技术与方案 19](#_Toc24905156)

[6 数据库设计 19](#_Toc24905157)

[7 详细设计 19](#_Toc24905158)

[7.1 \*\*功能模块 20](#_Toc24905159)

[7.1.1 功能描述 20](#_Toc24905160)

[7.1.2 性能描述 20](#_Toc24905161)

[7.1.3 输入 20](#_Toc24905162)

[7.1.4 输出 20](#_Toc24905163)

[7.1.5 程序逻辑 20](#_Toc24905164)

[7.1.6 限制条件 20](#_Toc24905165)

记录更改历史

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **更改原因** | **版本** | **作者** | **更改日期** | **备 注** |
| 1 | 创建 | V0.1.0 | 队员1 | 2019.11.6 | 项目概述和开发计划 |
| 2 | 创建 | V0.2.0 | 队员1 | 2019.11.7 | 可行性分析和需求分析 |
| 3 | 更新 | V0.2.1 | 队员1 | 2019.11.10 | 项目概述 |
| 4 | 更新 | V1.0.0 | 队员1 | 2019.11.13 | 修改和审查前四个（初赛）部分 |
| 5 | 更新 | V1.0.1 | 队员1 | 2019.11.15 | 修改部分术语、修改界面需求 |

# 项目概述

## 项目背景

市场上听写知识型软件非常少，依托于快应用平台上的同类软件更是寥寥无几，造成相关产业领域存在巨大的社会痛点，也缺乏对应的应用解决用户的相关需求。市场上现有的手机端APP一般只是使用NLP技术实现语音转文字功能，且仅支持单人使用和记录，带来了识别效果不佳、用户体验差、使用流程繁琐等问题。“妙记”，不仅将语音转化成文字，还支持用户上传照片，并将照片中的文字提取出来并且进行自动整理，感知词义，然后提取出关键字，知识图谱呈现在用户面前。

“妙记”主要针对参加会议、讲座、访谈、课堂的参会用户群体，用户通过使用面对面建群功能可以开展即时课堂或会议，旨在帮助用户记录和补充遗漏的会议要点和精彩瞬间，实时记录参会内容，构建知识图谱反馈给用户，实现智能摘要生成、词义自动搜索、笔记总结和知识点汇总，借助深度学习技术实现信息的二次提取，构建完整的智能摘要型一体化解决方案。同时，“妙记”支持多人同时记录，可以从多份样本记录中提取效果最佳的记录结果，大幅提高了识别率，对于用户的使用也更加友好。

## 项目定位

### 应用场景

1. 小王是是一个非常在意课堂笔记质量的同学。每次上课时，他都会把精力花在记录老师讲解的要点和PPT上的关键内容。但是课堂结束后，只有留下了笔记，对上课的内容一无所知。
2. 小丽是一个学霸，上课时积极和老师互动，课堂上的效率非常高。然而，小丽在课后复习的过程中有些困难，因为上课时没有记录关键的笔记，要复习的内容实在太多了。
3. 小张是一个商务狂人。每天的商务会议接连不断，因此会议迟到对他来说是家常便饭了。小张的苦恼是如果迟到太久，往往会错过会议的关键内容，导致小张对之后的会议内容云里雾里。
4. 小华是一个专心学术的研究生。每次兴高采烈地去听学术报告或者参加学术会议时，对汇报者说出的一些专业术语表示很无奈，碰到了自己的知识盲区。

在以上的场景中，例如会议、访谈、课堂、讲座等等，启动“妙记”的语音识别功能，助你凝神倾听主讲人所讲内容并及时记录下主讲人的讲话要点；点击拍照，助你你记录下所有的精彩瞬间；不知内容概要？“妙记”为你实时提取关键词并构建知识图谱，助你轻松理解讲话的中心和关键；想要复习讲话内容？“妙记”在讲话结束后为你生成一篇完整规范的文档，并自动摘要；一些生僻词令人困惑？“妙记”感知生僻词，自动为你搜索其词义，并带来丰富的拓展功能。

### 目标用户

需要在会议、课堂、访谈、讲座中实时记录笔记的所有与会人员，包括但不局限于学生、老师教授、专家学者、商业人士、企业高管等众多群体。

## 项目方案

“妙记”部署于快应用平台，区别于以往传统的APP应用存在用户留存度低等问题，采取轻量级、高便捷、低成本的方式给予用户更良好、更便捷的体验，旨在通过这样一款轻量级移动应用产品，切实解决用户在相关应用场景中的痛点问题。用技术改善生活，让应用改变世界。

对于会议、访谈等实时内容记录的场景，用户一键开启面对面建群即可马上开始实时记录，解放双手压力，减轻大脑负荷量。

对于课堂、讲座等内容驱动的场景，“妙记”通过构建知识图谱、展示关键词，和感知词义帮助用户快速整理出完整的摘要记录并快速高效理解当前的知识，扫除用户在理解知识方面的障碍。

## 项目目标

“妙记”以“即时感知，即时记录，即时理解”为宗旨，旨在提高会议、课堂、访谈、讲座效率。让用户可以在无论何时进入会议和课堂中通过查看记录和关键词能马上回顾错过的内容，在使用过程中查看提取出的关键字和知识图谱和词义感知出的名词解释从而即时理解，并且在整个使用过程结束后能生成一份会议访谈摘要，从繁重的手写记录中解放人们的双手。同时其一键面对面开启记录体现了快应用“快速”，“敏捷”的特性，参与课堂，会议，讲座访谈者们的手机构成了一个完整的语音、视觉传感器网络，体现了大赛“万物互融”的主题。

## 项目价值

基于万物互融的理念，旨在通过移动互联网应用的模式，创造全新的用户体验。通过精准定位目标用户群体，“妙记”可以大大提升参与会议、课堂、访谈的效率，使并对内容进行二次加工，使用户可以快速了解会议、课堂、访谈、讲座内容，将用户从记录笔记的工作中解放，留出更多的时间和精力进行思考学习。

# 开发计划

## 最终呈现形式

秉承基于万物互融的理念，创造全新的用户体验，方便人们的生活的原则， “妙记”旨在成为一个帮助人们快速和即时开展会议、课堂、访谈、讲座记录的实时听写知识性软件。

参与会议、课堂、访谈、讲座的用户可以通过手机端一键开启面对面建群的功能，进入该群组后，该群组内的用户的手机会自动将记录到的声音上传到服务端，服务端自动生成记录和关键词，知识图谱显示在用户的下拉框中，用户还可以将拍摄到的PPT内容上传至服务器，服务器会将照片中的内容自动转化成文本并追加到记录和关键词，知识图谱中。基于知识图谱，词义感知可以给用户全新的智能体验，在会议、课堂、访谈、讲座结束之后，每个用户会收到整理好的摘要记录和知识图谱，还可以查看历史来回顾完整的会议或者课堂记录。

因此，“妙记”的最终使用流程如下:

1. 一个团队(团队人数不限甚至可能是一个人)决定开始会议或者课程，通过输入4位代号开启共同群组(面对面建群)，该群内的成员会分享彼此的数据
2. 使用过程中通过输入相同的4位代号可以加入新的成员，各个成员可以通过拍照上传的方式加入记录。并且可以随时查看当前的关键词，知识图谱历史记录，搜索结果。
3. 客户可以随时结束，并在整个会议，课堂，访谈结束后生成一个完整的摘要和知识图谱。

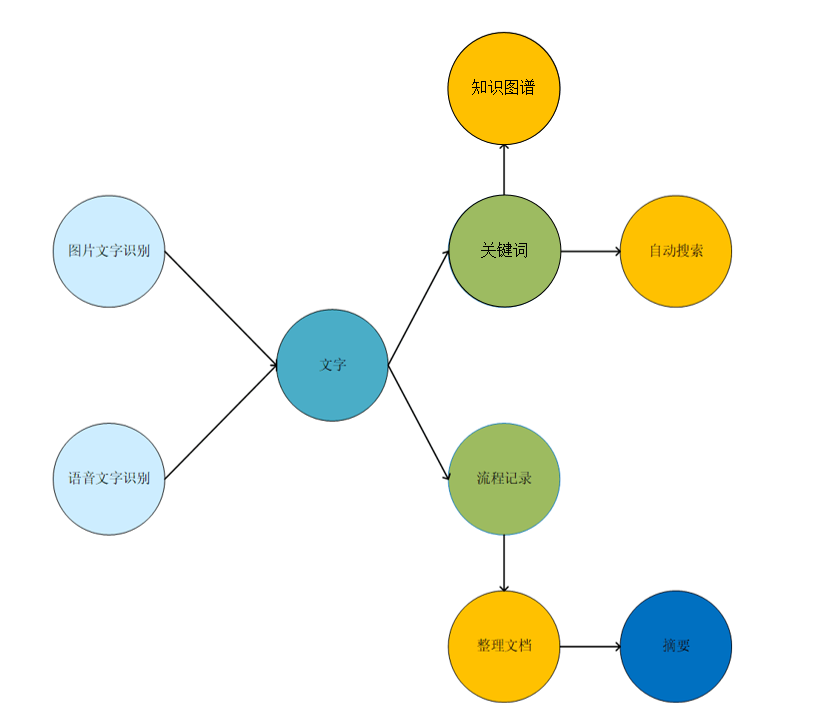


图 2‑1 流程示例

## 主要功能描述

本项目的主要面向人群是参会人员、学生和老师，使用场景是当用户需要“妙记”，系统可以实时地识别语音和图像，方便地让用户接受实时信息。

基于以上描述的使用场景，将系统主要功能划分为文字识别、文本学习、和词义感知三大模块，每个模块下面有若干个功能。下表即为系统的主要功能描述。

表 2‑1 主要功能描述

|  |  |
| --- | --- |
| 模块 | 功能 |
| 文字识别 | 识别语音和图片中的文字 |
| 文本学习 | 学习文本以提取关键词，构建知识图谱，自动生成 |
| 词义感知 | 从文本中感知不常见词和专业术语，对这些词解释词义并联想到其他词 |

## 运行环境

1. 客户端运行于当前主流支持快应用平台的安卓智能手机。



图 2‑2 快应用生态成员

1. 服务端运行于各大发行版Linux操作系统，如Ubuntu 16.04及以上版本，CentOS7及以上版本等等。

## 验收标准

1. 功能验收：功能实现正确、完整，达到目标结果，用例符合文档需求。
2. 性能验收：项目性能达到开发文档需求部分说明的要求，可以最终上线，且具有可靠性、易用性。
3. 文档验收：文档准备齐全，内容全面详尽，包含项目所需信息和所有功能模块。文档清晰，容易理解，没有语病和歧义，能通过适当的术语、图形表示。
4. 代码验收：代码完整，书写风格统一，采用标准规范，可读性强，无数据丢失、运行过慢等问题。

## 关键问题

在项目开发前，需要详细完整地考虑到开发过程中将会遇到的问题。我们将问题分为三 类：设备条件问题、技术问题、其他风险。设备条件问题主要是在项目运行阶段，可能会由 于设备造成的问题。技术问题是为了完成项目，将会主要解决的技术难点。下表展示了我们项目的关键问题以及对应的解决方案。

表 2‑2 关键问题

|  |  |  |
| --- | --- | --- |
|  | 问题描述 | 相关对策 |
| 设备条件 | 由于服务器的问题，用户在上 传语音或图片时，上传或分析失败 | 在软件方面，失败后提醒用户 再次进行操作。 在硬件方面，使用可靠性高的服务器。 |
| 在最糟糕的情况下，由于硬件 的原因，造成存储的用户数据丢失 | 对用户数据进行相应的备份 |
| 技术焦点 | 如何识别语音 | 通过不断训练网络，提高识别准确性 |
| 如何识别图片中的文字 | 训练CRNN模型 |
| 其他风险 | 项目进度把控不当，无法按时 交付项目 | 每隔一段时间对项目进行一个阶段性评估，当发现项目进 度并不良好的时候，及时对项 目进度进行适当的调整 |

## 进度安排

表 2‑3 进度安排

|  |  |
| --- | --- |
| 项目安排 | 预计完成时间 |
| 项目相关文档完成 | 2019-10-16 |
| 模型构建 | 2019-12-01 |
| 快应用设计 | 2019-12-06 |
| 服务端开发、测试阶段 | 2020-01-15 |
| 模型优化 | 2020-01-31 |
| 快应用二次设计、开发 | 2020-02-13 |
| 系统完整测试阶段 | 2020-03-01 |
| 迭代开发 | 2020-05-09 |

## 开发预算

表 2‑4 开发预算

|  |  |  |
| --- | --- | --- |
| 项目类别 | 项目内容 | 费用预算（元） |
| 设备 | 服务器 | 500 |
| 设备 | 开发和测试硬件 | 10000 |
| 软件 | API接口 | 100 |

# 可行性分析

## 技术可行性分析

本项目目前用到的主要技术有：

前端：HTML、CSS和JavaScript以及快应用框架；

后端：Python、Flask；

数据库：MySQL；

服务器：Nginx；

配置管理工具：Git；

开发工具：Visual Studio Code；

其他详细技术：

1. 项目中的文字提取，关键词提取，文本相似度功能可以通过调用已有的API接口完成。
2. 对于知识图谱的构建暂时还没有相关资料，但是可以根据数据词云的思想来模拟成树状图。
3. 对于自动摘要模型的构建，可以先使用已完成的TEXTRANK4zh来简单实现，或者使用TEXTRANK和WordNet的自动摘要。
4. 多位用户开启面对面群组加入同一频道，手机同时开始上传语音数据。

## 资源可行性分析

1. 资金：项目只需要租用一台服务器就能够运行，对资金要求不高。
2. 系统资源:快应用可以获得系统的录音，拍照等权限。
3. 对于API的使用，可以每人注册一个账号，然后轮流使用账号资源。
4. 对于自动摘要模型训练的数据集，可以使用LCSTS，一个大型中文短文本数据集，也可以使用《文本摘要常用数据集和方法研究综述》提到的一些数据集。

## 市场可行性分析

1. 市场上同类的软件非常少， 同样在可以在快应用平台上使用的同类软件更是寥寥无几。在手机端的app一般是都是使用NLP技术将记录到的语音转化成文字功能。而我们的产品不仅将语音转化成文字，还支持用户拍照上传，并将照片中的文字提取出来然后自动整理这些文字，之后提取出知识图谱和关键字及其搜索结果展示在用户面前，最后得到一份整理文档和摘要。
2. 用户可以使用一键面对面建群功能即时开展即时课堂，会议目前市面上的同类软件比如搜狗听写是支持单人使用进行记录这样导致有时候识别效果不佳，而我们的产品支持多人同时记录，这样可以从多份样本得到较好的语音，提高了识别率。这样可以最大化的满足用户对识别准确程度的需求。
3. 为了更好地了解市场需求，我们团队就身边的人进行了问卷调查，调查结果如下：

本次调查共包含110人，其中学生居多（70人），其次是公司职员及管理人员（18人），之后是教师（10人）以及经营商铺的个体户（2人）。

由上图可以看出，频繁需要记录的的人数占62%，18%的人经常使用，16%的人偶尔使用，只有4%的人从不使用。

很多人比较青睐知识图谱，和自动文档整理的功能，这说明大部人在进行记录也会遇见这几个问题，我想专心听讲，不想摘抄PPT上的内容怎么办？这个讲座的知识好专业呀我听不懂怎么办?主讲人的关键内容我不小心走神错过了怎么办？我想快速了解整个记录的流程怎么办？在这个会议，讲座，课堂，访谈结束之后我想留存一份文档记录怎么办？

针对上述问题，“妙记”有一套完整的解决方案。

# 需求分析

## 数据需求

### 静态数据

表 4‑1 静态数据

|  |  |  |  |
| --- | --- | --- | --- |
| 分类 | 名称 | 定义 | 类型 |
| 数据库 | DATABASE\_URI | 数据库连接信息 | String |
| TRACK\_MODIFICATIONS | 数据库修改跟踪选项 | String |
| 基本配置 | SERVER\_NAME | 服务器域名 | String |
| SECRET\_KEY | 服务器密钥 | String |
| 用户设置 | Nick | 用户在群组中的昵称 | String |
| WordSensor | 是否开启词义感知 | Boolean |

### 动态数据

表 4‑2 动态数据

|  |  |  |  |
| --- | --- | --- | --- |
| 分类 | 名称 | 定义 | 类型 |
| 基本数据 | GroupId | 群组ID | String |
| Location | 用户位置信息 | String |

### 数据词典

表 4‑3 数据词典

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 允许为空 |
| 1 | Speech | 用户记录的语音 | Sound | 否 |
| 2 | Photo | 用户拍下的照片 | Picture | 是 |
| 3 | Text | 从声音和图像中识别出的文本 | String | 否 |
| 4 | Keywords | 从文本中提取的关键词 | Array | 是 |
| 5 | Graph | 知识图谱 | Picture | 是 |
| 6 | Abstract | 自动摘要 | String | 是 |
| 7 | Document | 整理后的文档 | Document | 是 |
| 8 | Meaning | 感知词义 | Array | 是 |

### 数据采集

1. 在场所有用户话筒采集的声音信息。
2. 用户拍照上传的图片中包含的文字信息。
3. 公开的LCSTS数据集

## 功能需求



图 4‑1 功能模块结构图

表 4‑4 核心功能模块描述

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| **文字识别** | 语音整合与识别 | 多个用户开启语音后，系统实时地转化语音为文字，并整合多个音源提高识别率 | 10 |
| 图像处理与识别 | 用户拍照上传图片后，系统处理图像并识别出其中的文字 | 9 |
| **文本学习** | 关键词提取 | 系统从文本中查找出文本中及其有代表性的关键词 | 8 |
| 知识图谱构建 | 系统从文本中抽取实体和关系并完成知识推理 | 8 |
| 文本摘要生成 | 系统通过学习文本，生成摘要 | 7 |
| 文档整理 | 系统从杂乱无章的文本中寻找规律，整理出一篇规范的文档 | 5 |
| **词义感知** | 生僻词感知 | 系统从文本中感知不常用词或者专业术语 | 6 |
| 词义解释 | 系统将感知而出的词语解释含义，并联想到其他词语，帮助用户理解局部文本 | 6 |

### 文字识别功能模块



图 4‑2 文字识别用例图

表 4‑5 文字识别用例规约

|  |  |
| --- | --- |
| 用例名称 | 识别文字 |
| 功能简述 | 用户开启录音功能并随时拍照，系统即可实时识别其中的文字 |
| 用例编号 | UC000 |
| 执行者 | 用户 |
| 前置条件 | 录音源和拍照质量良好 |
| 后置条件 | 系统从录音和拍照中识别出文字 |
| 涉众利益 | 用户记录音频和图像中的文字 |
| 基本路径 | 1. 用户开启录音 2. 系统实时识别音频中的文字 |
| 扩展路径 | 1a. 用户随时拍照  系统自动优化图片，并提取其中的文字 |
| 字段列表 | 语音、拍照图片 |
| 设计规则 | 独立模块 |
| 未解决的问题 | 从多个用户上传的语音中择优或者整合以提高识别率 |
| 备注 | 识别出的文字若不符合语法则认为识别失败 |

### 文本学习功能模块



图 4‑3 文本学习用例图

表 4‑6 文本学习用例规约

|  |  |
| --- | --- |
| 用例名称 | 文本学习 |
| 功能简述 | 从文本提取关键词、形成知识图谱、生成文本摘要和整理后形成规范的文档 |
| 用例编号 | UC001 |
| 执行者 | 系统 |
| 前置条件 | 系统需成功提取文本 |
| 后置条件 | 系统将从文本中学习得到的关键词，知识图谱，文本摘要和文档展示给用户 |
| 涉众利益 | 用户了解文本中概要知识和关联知识 |
| 基本路径 | 1. 系统识别文本 2. 系统学习文本 |
| 扩展路径 | [待定] |
| 字段列表 | 关键词、知识图谱、文本摘要、文档 |
| 设计规则 | 独立模块 |
| 未解决的问题 | 发现关键词之间的关联关系 |
| 备注 | 整理所得文档若无可读性则认为功能有缺陷 |

### 词义感知功能模块



图 4‑4 词义感知用例图

表 4‑7 词义感知用例规约

|  |  |
| --- | --- |
| 用例名称 | 词义感知 |
| 功能简述 | 感知文本中的不常用词或者专业术语，并展示出其词义 |
| 用例编号 | UC002 |
| 执行者 | 系统 |
| 前置条件 | 系统需成功提取文本 |
| 后置条件 | 系统展示生僻词的词义（对用户来说） |
| 涉众利益 | 用户理解文本中不常见词语或专业术语 |
| 基本路径 | 1. 系统识别文本 2. 系统感知生僻词并解释含义 |
| 扩展路径 | [待定] |
| 字段列表 | 生僻词、生僻词的词义 |
| 设计规则 | 独立模块 |
| 未解决的问题 | 判断生僻词 |
| 备注 | 系统识别出的不常用词或专业术语若有悖常识则认为识别失败 |

## 性能需求

### 时间特性

1. 响应时间：

因为本项目是以人工智能为主题，要涉及大量的计算和分析，需要在准确率和相应时间二者之间做折衷。在实时语音转化方面，我们要求用户在快应用中做出操作后代码更改响应时间在 0.4 秒内。在文本学习模块中，我们要求响应时间在 5秒以内。同样地在词义感知模块中，我们要响应时间不超过1秒。

1. 数据传输时间：

数据获取和传输的速度也影响着用户的体验。我们要求数据从获取到显示在页面的时间 不超过 0.8 秒。

### 适应性

由于我们的产品是收集在场参与所有用户的声音信息，因此一般来说，参与用户越多，得到的声音质量越好，但是也会增大时延，这两者需要综合考虑，可以在以后的设计中，进行进一步的优化。

## 界面需求

1. 入口界面

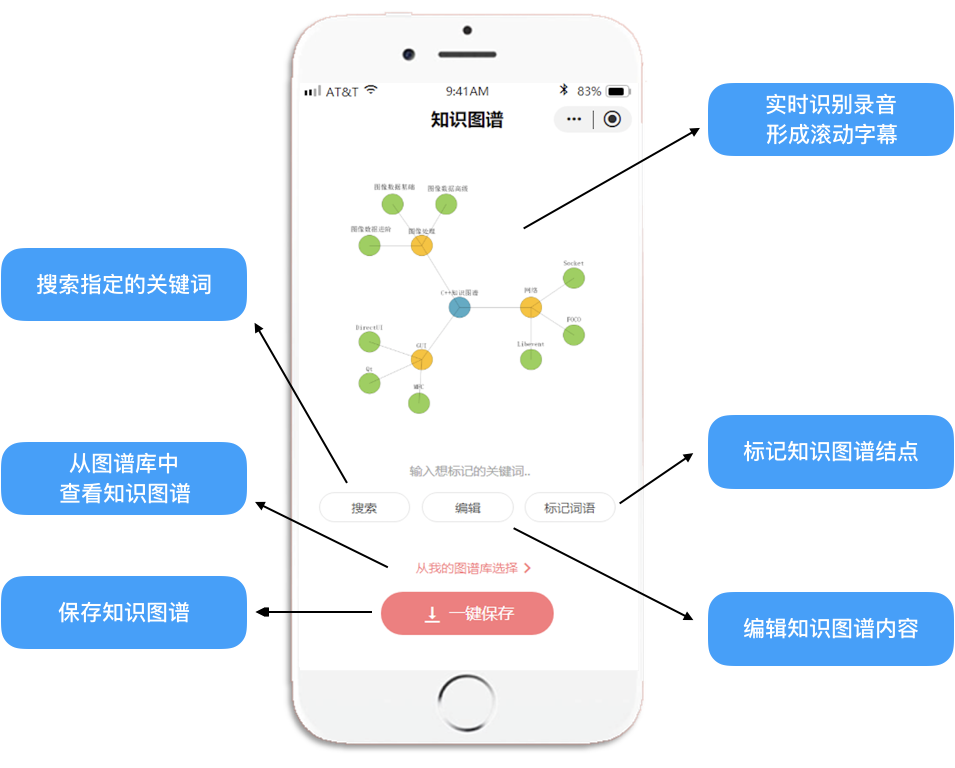




1. 主界面



1. 知识图谱页面



1. 文档和摘要界面



## 接口需求

### 硬件接口

无

### 软件接口

表 4‑8 软件接口

|  |  |
| --- | --- |
| 对外服务依赖 | 交互方式 |
| 科大讯飞语音听写 | HTTP请求 |
| 科大讯飞关键词提取 | HTTP请求 |
| 文本相似度 | HTTP请求 |

## 其他需求

### 可移植性

为了方便用户使用，我们的产品确保在各大系统平台都能够准确的运行。

### 可分享性

为了方便即时编辑处理文本，我们产品确保用户可以随时分享保存的文本给其他应用。

# 概要设计

## 处理流程

## 总体结构设计

## 功能设计

## 用户界面设计

## 数据结构设计

## 接口设计

### 外部接口

### 内部接口

## 错误/异常处理设计

### 错误/异常输出信息

### 错误/异常处理对策

## 系统配置策略

## 系统部署方案

## 其他相关技术与方案

# 数据库设计

# 详细设计

## \*\*功能模块

### 功能描述

### 性能描述

### 输入

### 输出

### 程序逻辑

### 限制条件