

appid	wx8b7a09b5490be970
-------	--------------------

# 微信小程序应用开发赛

## 介 绍 文 档

参赛类别: 微信小程序

项目名称: 师生互选助手

队伍名称: 湖南大学 NEST 实验室

团队学生: 汪铭林 曹航程 胡靖阳 沈湘宇

指导老师: 蒋洪波教授

所属高校: 湖南大学

电子邮件: wminglin@hnu.edu.cn

2019 年 5 月

# 目录

第 1 章	团队组成及分工 .....	3
1.1	团队组成 .....	3
1.2	成员分工 .....	4
第 2 章	项目介绍 .....	5
2.1	项目背景 .....	5
2.1.1	社会背景 .....	5
2.1.2	相关领域背景 .....	5
2.2	项目主要内容 .....	6
2.2.1	项目内容总览 .....	6
2.2.2	项目主要功能介绍 .....	6
2.3	项目意义 .....	7
2.4	项目创新性 .....	7
第 3 章	项目可行性分析 .....	9
3.1	市场需求分析 .....	9
3.2	目标用户分析 .....	9
3.2.1	参加研究生（硕博生）入学考试的学生 .....	9
3.2.2	获得入学资格的准研究生（硕博生） .....	9
3.2.3	高校的硕博研究生导师 .....	10
3.3	技术可行性分析 .....	10
第 4 章	产品介绍 .....	11
4.1	产品界面与功能总览 .....	11
4.2	产品子界面与功能详细介绍 .....	12
4.2.1	用户登录与导师信息搜索引导界面及功能 .....	12
4.2.2	个人基本信息与身份修改界面及功能 .....	13
4.2.3	使用帮助与意见反馈界面及功能 .....	14
4.2.4	导师信息搜索界面与功能 .....	14
4.2.5	所选院系导师推荐界面 .....	15
4.2.6	所选导师个人信息界面与功能 .....	16
4.2.7	导师非负面评价界面与功能 .....	16
4.2.8	基于深度学习人脸对比认证界面与功能 .....	17

4.2.9 师生内部系统通信界面与功能.....	18
第 5 章 技术方案及细节.....	20
5.1 小程序中涉及到的前端技术.....	20
5.1.1 身份认证模块.....	20
5.1.2 导师信息搜索模块.....	21
5.1.3 热门导师推荐模块.....	23
5.1.4 个人信息模块.....	23
5.2 小程序所涉及到的后端技术及数据库设计.....	25
5.2.1 Spring boot 框架简介.....	25
5.2.2 Spring boot 的特性.....	25
5.2.3 MyBatis 框架简介.....	25
5.2.4 后端模块简介.....	26
5.3 小程序数据爬取涉及技术.....	29
5.4 小程序数据库设计概述.....	30
5.5 小程序数据库详细设计.....	31
第 6 章 系统运维方案.....	34
6.1 运行维护目标.....	34
6.2 运行维护内容.....	34
6.3 程序异常事件处理流程.....	34
第 7 章 总结及展望.....	35
7.1 功能拓展展望.....	35
7.2 《师生互选助手》推广方式.....	35
7.3 小程序总结与展望.....	35

# 第1章 团队组成及分工

本章摘要：本章主要介绍本次队伍组成人员的基本信息及参赛学生的各项分工，其中包括参赛学生及指导老师的个人简历信息。

## 1.1 团队组成

指导老师	姓名	蒋洪波	已获学位	博士	职称	教授
	所属单位	湖南大学信息科学与工程学院		电子邮箱	hongbojiang@hnu.edu.cn	
	导师简介	蒋洪波教授 2008 年毕业于美国 Case Western Reserve University，获计算机科学博士学位。2008 年至 2017 年曾任华中科技大学教授。目前为湖南大学“岳麓学者”特聘教授，信息科学与工程学院院长助理，博士生导师。中国计算机学会 CCF 杰出会员、ACM 高级会员、IEEE 高级会员、湖南省计算机学会常务理事兼副秘书长。研究方向为移动计算与网络大数据，在 IEEE/ACM Transactions 期刊发表论文 47 篇，国际会议 ACM MOBICOM 论文 2 篇、ACM SIGMETRICS 论文 1 篇、ACM CoNEXT 论文 1 篇、ACM MOBIHOC 论文 2 篇、IEEE INFOCOM 论文 10 篇等。受邀担任 CCF A 类期刊 IEEE/ACM Transactions on Networking 的编委、CCF A 类期刊 IEEE Transactions on Mobile Computing 的编委，SCI 一区期刊 IEEE Communications Magazine 的编委。				
学生成员 1	姓名	汪铭林	已获学位	硕士在读	研究方向	移动计算与智能感知
	所属单位	湖南大学信息科学与工程学院		电子邮箱	wminglin@hnu.edu.cn	
	学生简介	本人掌握 java 语言基础、java web 后台开发，MVC 设计模式、Springboot 框架以及安卓开发。在本科期间曾获得“专业一等奖学金”、“国家励志奖学金”等多项奖励。本人本科期间积极参加学科相关竞赛，并获得“蓝桥杯 JAVA 软件大赛”安徽省二等奖、安徽省“炜煌杯”单片机应用技能大赛一等奖、万方杯数据库检索大赛三等奖等。				
学生成员 2	姓名	曹航程	已获学位	硕士在读	研究方向	移动计算与智能感知
	所属单位	湖南大学信息科学与工程学院		电子邮箱	chc_hnu@hnu.edu.cn	
	学生简介	本人掌握 java 语言基础、网络爬虫基本框架、java web 后台开发及相关 mvc 框架。在本科及研究生期间曾获得“国家励志奖学金”、“学业单项奖学金”、“美国数学大学生数学建模比赛国际一等奖”、“全国大学生数学建模比赛省二等奖”、“百度-西交大·大数据竞赛 2018——商家招牌的分类与检测比赛 70(排名)/1139(参赛队伍数)”、“省级优秀本科毕业生”及“全国大学生创业创新项目”等奖励或荣誉。				
学生成员 3	姓名	胡靖阳	学位	硕士在读	研究方向	移动计算与智能感知
	所属单位	湖南大学信息科学与工程学院		电子邮箱	fbhhjy@hnu.edu.cn	
	学生简介	本人掌握 java 语言基础，安卓开发及相关的 mvc 框架，目前专注于研发基于声学信号的新型人机交互系统。				

学 生 成 员 4	姓名	沈湘宇	学位	博士在读	研究方向	移动计算与智能感知
	所属单位	湖南大学信息科学与工程学院		电子邮箱	shenxiangyu@hnu.edu.cn	
	学 生 简 介	本人掌握 python 语言基础，了解人脸识别相关模型及人脸追踪相关模型，目前专注于基于计算机视觉的新型推荐系统。				

## 1.2 成员分工

学生成员 1	主要负责项目开发，其中包括微信小程序前端设计（界面布局设计、逻辑交互设计）及微信小程序后端开发（后端采用 SpringBoot+MyBatis 框架来处理前端请求以及数据库交互）
学生成员 2	主要负责导师信息的数据爬取，实现导师身份认证接口，提供相应功能的设计方案及技术方案，完成最终产品文档的撰写和整合。
学生成员 3	主要负责主界面以及用户首次使用的功能提醒界面设计，主要包括用户指南和反馈部分的页面设计。同时在后期的文档编写过程中，负责功能界面及相关功能介绍。
学生成员 4	主要负责系统数据库中的数据收集以及前期预处理。

## 第2章 项目介绍

本章摘要：本章主要对项目背景及项目内容进行了详细介绍，其中项目背景包括构建该项目的社会背景和相关的领域背景，并且本节中简明的概括了项目的主要功能模块。

### 2.1 项目背景

#### 2.1.1 社会背景

建设教育强国是中华民族伟大复兴的基础工程。优先发展教育，加快了我国教育现代化建设的进程。进入 21 世纪以来，我国国民的高学历比例显著提高，国民文化水平结构不断得到优化。特别是近几年，我国研究生数量随时间大幅增长。据教育部数据统计，2016 年至 2019 年全国硕士研究生报名人数分别为 177 万，201 万，238 万和 285 万，相对增长率分别为 7.34%，13.56%，18.41%和 19.75%。2016 年至 2019 年，全国高校的研究生招生人数每年大约以 4 万递增，2019 年招生人数突破 70 万，这意味着参加并通过研究生考试的考生数量不断增加。

对于数量庞大的研究生备考群体及准研究生群体而言，合理规划并度过在校学习时光，不仅仅要通过自己的努力，还需要导师的耐心及细心指导。研究生导师对于研究生学习生活的指引至关重要，寻找适合自己未来规划及发展的导师是正确开启研究生涯的第一步。作为研究生导师，同样希望寻找适合本实验室研究方向及发展的学生，帮助学生在实现自我人生价值的基础上，不断的推进实验室建设。所以在学生寻找心仪导师及导师寻找心仪学生的双向选择过程中，找到一个快捷高效的通讯平台，从而实现双方之间的高效交流是十分重要的。

师生间进行交流的触发点为该生通过了解该导师的个人信息，然后确定其为心仪导师并主动进行联系。对于新生而言，如何去充分且准确查询所在院系的师资信息是联系导师的首要环节。导师根据与其联系的学生基本学习情况及未来发展规划，进行选择性的沟通，最终确定个人招生名单。从小的方面而言，高质量的进行师生互选对学生个人未来发展及实验室建设起到了关键性作用。从大的方面而言，高质量的进行师生互选对我国的研究生教育及培养起到正面的推动作用。综上，为学生提供全面详细的导师信息，并为师生间提供快捷高效的交流平台是十分重要的。

#### 2.1.2 相关领域背景

目前存在少量的网站系统及手机应用，可以为学生提供导师的个人信息，为在进行导师选择的时候提供导师个人信息，但是这些应用存在着严重的不足，我们总结为如下几点：

- 提供的导师信息不全面且不准确。这些产品的数据主要来源为院校的官网，即并未对导师信息进行细粒度收集及优化整理，只是进行简单的原始数据展示。并且现有产品的导师个人数据长期不进行更新，即高校官网师资信息进行更新而产品数据依旧不变。对于学生进行导师数据收集而言，这些未经优化处理的原始数据可能并无太大价值，不及时更新数据的产品甚至会对用户产生误导。
- 交互效果差。在移动端产品快速发展的时代，大多数基于 PC 端的网站系统并未实现移动端的界面动态适配，这对于用户而言操作体验差。现存相关的移动端产品将导师信息查询仅作为一个小的功能接口，用户很难快速查询到相关导师信息，这类产品

会大大降低用户的信息检索效率。

目前学生与心仪导师在互选交流过程中,大多使用电子邮件、移动电话及实时聊天软件,但这三种通讯方式针对现有的使用场景存在着一些缺陷,我们总结为如下几点:

- 电子邮件代替了纸质邮件,为人与人之间的交流提供了极大的便捷,但是对于师生互选而言这个方式并不是最有效的。部分老师和学生,并不习惯经常性的去查询是否有新的邮件到达,这可能导致双方无意的遗忘个别邮件,断点式的交流并不利于沟通。一个学生可能因为邮件信息的无意遗漏错过其心仪的导师,一个导师也可能因为邮件信息的无意遗漏错过满意的学生。
- 移动电话可以完成长距离的实时通讯,但是这是在保证接收双方都有时间且都有意愿进行通话的条件下。硕博导师需要完成繁重的教学工作和学术研究工作,所以对于老师而言,抽取大量的时间与未达成明确互选意向的学生进行电话交流是不现实的。师生初步进行相互交流时,这并不是一个很好的沟通途径。只有在双方有了基本的了解和初步达成互选意向后,才可能通过电话沟通进行深入了解。
- 实时聊天软件具有一定的隐私性,师生双方只有完成初步了解过程后,才会有互加好友进行交流的意愿和可能性。对于导师而言,每年招收硕士、博士名额有限,但联系其的学生众多,通过聊天软件互加好友进行初步交流无疑会导致大量的社交冗余信息,对于导师而言这无疑增加了社交负担。

## 2.2 项目主要内容

### 2.2.1 项目内容总览

微信小程序《师生互选助手》,主要用于解决硕博入学考试备考者和准硕博群体收集导师完整信息难及师生互选初期通讯交流缺乏专业平台两大问题。本小程序为学生提供完整详细的导师信息,为师生双方提供专用内部通讯系统,为通过人脸认证的导师提供实时修改个人主页信息及招生信息的权限,同时系统内的导师正面评价系统可为学生在进行导师选择时提供正确指引。师生用户无需安装任何应用,通过微信进入到小程序端便可获取便捷高效的服务体验。

### 2.2.2 项目主要功能介绍

本次设计的微信小程序《师生互选助手》,主要实现了以下几大功能:

- 为相关学生群体提供所需院校导师的详细信息。为了能够给学生提供导师的详细信息,小程序数据库中的导师信息除了获取自高校官网,还获取自教师个人学术主页及所在实验室主页等。系统内通过多途径的信息融合,使学生以高效率完成导师信息的收集。本功能促进了导师端信息向学生端流通,尽可能缓解信息不对称给师生初步互选带来的影响。
- 微信小程序中构建师生互选专用的通讯平台。为了师生间进行迅速快捷的交流,小程序中构建的通讯系统实现了实时通讯及文件收发的功能。打开小程序中个人主页界面,便可看到消息互发图标,并且系统会提示未读消息数,这尽可能保证了师生双方的通讯消息不会被遗漏。师生建立通讯关系之后,两人之间将会建立聊天模式,可进行双方之间的实时通讯。
- 高校导师的快速人脸识别认证。微信小程序中,会为每个进入小程序的用户提供唯一标

识码, 该标识码与用户微信号一一对应。构建师生通讯系统的前提是系统内部能够确定导师微信号与标识码的对应关系。小程序内完成导师的人脸认证后, 便可将导师微信号绑定相应标识码。学生可通过师生通讯系统完成与心仪导师的实时交流及文件发送。为了使用目前掌握的导师信息完成认证, 本系统中使用了人脸对比算法。打开认证界面, 认证者利用前置摄像头完成拍照, 然后将认证照片与数据库中相应的导师照片提交到后台服务器完成特征对比。导师通过人脸认证便可进入导师端界面, 并被授权相应的个人主页信息修改权限。导师未通过人脸认证则会返回人脸认证的失败原因, 如未通过活体认证、匹配相似度低等。

- 通过认证后的导师可在小程序中修改个人主页信息, 系统实时进行导师个人主页信息的更新。在本次的系统设计中, 为了更为合理便捷的获取导师个人信息, 我们为通过认证的导师赋予修改个人简历信息的权限, 每一个导师可以根据当前自身的研究状况更新个人信息。导师对个人数据的更新, 是对系统数据库原有导师个人数据很好的补充。
- 导师评价子系统, 为学生在进行导师选择时提供指引。该评价系统希望通过在读学生对导师的正面评价(如科研能力强、项目多、教学严谨负责和实验室氛围好等等), 实现对新生在选择导师时的良性引导。学生可以依据对该导师的了解, 进行点赞或文字评价。系统会对学生的评价进行筛选, 筛选出非负面的评价展示在评论区。剔除负面评价的目的在于更好的利用正面评价引导新生进行导师选择, 而不是用导师间的正反面评价形成反差引导学生进行导师选择。

该系统主要包括导师详细信息提供、系统内部师生通讯、导师人脸认证、导师个人主页信息实时修改及导师正面性描述评价五个子功能, 来助力师生双方的互选。

## 2.3 项目意义

对于备考研究生入学考试的硕博学生及准研究生这两个群体而言, 师生互选助手为每个使用者提供了一个查询导师详细信息及师生高效交流的专业平台, 缓解了师生互选过程中信息不对称的问题。微信小程序的灵活性强, 用户无需再下载额外的第三方插件或手机应用。通过这个小程序, 导师信息无障碍的被学生用户获取, 从而解决了信息资源闭塞给学生带来的影响。

对于工作繁忙的硕博导师而言, 通过微信进入小程序, 便可实现与前来联系的学生使用内部通讯系统进行交流。这简化了师生交流过程中信息查询平台和通讯平台之间的跳转流程, 给导师在选择学生的过程中带来极大的便利。同时, 导师可以随时修改个人主页信息以完成对个人数据的补充及更新。

综上所述, 师生互选助手解决了师生间信息不对称的问题, 并且为师生交流提供了一站式的通讯平台。

## 2.4 项目创新性

当前我国研究生数量不断增长, 而相关的用于研究生导师信息查询及师生互选初期通讯的产品并不成熟。本次我们设计的师生互选助手, 解决了现有产品所表现出的种种缺陷, 在解决实际问题的过程中, 我们提出了一些新的设计概念和方法, 即创新性总结如下:

- 系统中第一次提出利用导师的学校官网信息、实验室个人信息及学术主页信息完成导师个人信息的补充, 把信息收集及整理过程留给系统设计人员。多方数据融合给学生用户



提供高质量的导师信息，解决了由于信息资源的不对称带来的信息闭塞，使用户快速高效得到需要的信息

- 系统中第一次提出将导师信息检索及师生互选初期通讯两大功能构建在同一个产品中。学生在进行导师信息查询时，可以通过微信小程序中的通讯系统实现与心仪导师的交流，从而实现将多项功能的多平台多步操作集成在一个产品中。
- 系统中第一次提出利用人脸对比认证实现导师微信号标识码与系统个人数据进行关联。本文中，为了实现系统内部通讯，必须先实现导师的实名认证，即找出导师微信的标识码和导师的一一对应关系。我们提出利用导师个人简历中的照片及其他相关个人照片与认证时拍摄的照片进行人脸对比，从而实现导师实名认证。
- 系统中第一次提出为通过身份认证的导师提供实时修改个人主页信息的权限，这将小程序构建为全国高校教师的信息发布及查询平台。导师本人对个人信息的修改，与系统收集到的数据完美互补，进一步提升了该导师个人信息的完整性。

## 第3章 项目可行性分析

本章摘要：本章主要对项目的实施可行性进行了分析，主要包括师生互选助手的市场需求分析、目标用户分析及技术可行性分析，其中目标用户分析包括参加研究生考试的硕博生和硕博研究生导师两部分分析。

### 3.1 市场需求分析

近四年，我国高校研究生报考人数从 177 万增长至 290 万，录取人数由 59 万增长至 74 万，报考人数增长率约为 18%，这意味着目标用户人群在不断的扩大。通过层层选拔最终通过研究生入学考试的学生，需要有着充分的准备、扎实的专业知识基础和较强的心理素质。所以对于他们而言，能够取得入学资格十分不易，也会格外慎重的对待接下来的每一个入学环节。

研究生新生入学前需经历导师信息收集、联系心仪导师、确定导师并进行师生互选四个基本过程。但是当前市场中相关的导师信息提供产品并不成熟，主要体现在提供的导师信息过时且信息缺失严重，这意味着现有的产品并不能为用户提供优质的服务。当前主流的通讯工具，如移动电话、电子邮件及实时聊天应用，并不满足师生进行初步互选通讯交流的特定要求。现有的产品将导师信息查询及师生通讯两大功能完全分离，并不能为师生间互选提供一体化的服务。一款产品，若拥有高质量信息查询和便捷式师生交流两大功能，无疑会给目标用户提供高质量的使用体验。

### 3.2 目标用户分析

#### 3.2.1 参加研究生（硕博生）入学考试的学生

对于参加研究生入学考试的学生而言，选择准备报考的目标高校是整个备考学习过程的第一步。影响学生选择目标高校的因素主要有以下两个，即高校的综合排名和所报考院系的师资力量及研究方向，而因素二往往在学生进行院校选择时起到了决定性的作用。《师生互选助手》中提供完整详细的导师信息，可为学生在进行院校选择时提供较强的方向指引。在信息查询过程中无需安装任何插件或手机应用便可使用小程序实现各个学校间师资信息查询的切换，这简化了学生频繁查询相关信息的步骤，即单个平台实现多校信息查询。一个高效的导师信息检索平台，会给学生用户在信息查询时带来极大的便利。

#### 3.2.2 获得入学资格的准研究生（硕博生）

准研究生是本款微信小程序（导师选择助手）的主要目标人群。查询导师信息并和心仪导师进行沟通交流是他们在入学前的必要环节，选择一个适合自己的导师将会对未来几年的个人学习及发展起到积极的推动作用。所以对于准研究生而言，能否获取到完整的导师信息并实现与导师间的高效沟通是至关重要的。微信小程序中除了覆盖导师的个人信息，还提供了导师评价功能(学生评价经过后台审核后，才能进行显示；为了对新生进行导师选择时起到正面引导，负面评价会被自动筛选；本评价系统的引导原则为，越受学生喜爱的导师得到的正面评价越多)。通过参考其他学生对该导师的评价可以使用户更加深入了解心仪导师的

其他情况。同时，系统内部提供了师生间交流的内部通讯系统，可以实现二者间文件的传输及信息的交流。上述功能模块，使得准研究生可高效获取较为准确的信息，并高效完成与心仪导师的初步交流。

### 3.2.3 高校的硕博研究生导师

在每年的研究生新生入学前期，导师们希望选择心仪的学生加入自己的团队，从而为实验室注入新的血液。但是对于老师而言，他们有着较为繁重的教学任务和科研任务，所以能够用来与前来联系的学生进行交流的有效时间并不多。如何利用导师零碎的时间去有效的和学生交流是相关应用需要解决的实际问题。提供较为高效的信息交流及信息发布平台对于高校教师是极为重要的。在本微信小程序中，导师只需用微信端进入小程序，实现快速的教师人脸认证便可完成相关注册，相关操作时间不超过一分钟。完成认证的导师可以使用内部通讯系统实现与学生的交流及文件收发，在使用过程中，无需安装任何外部插件或第三方应用。微信小程序的使用体验，远远高于相应的网站系统和手机应用，这对于繁忙的导师用户而言，无意极大的提高了师生交流的效率。

## 3.3 技术可行性分析

师生互选助手的构建主要包括前端、后台、数据爬取及数据库设计四个方面的技术模块。其中前端设计使用微信小程序前端文本标记语言，后台设计利用 Spring boot 框架及 Mybatis 框架，数据爬取操作基于 python 的 Request 框架及 BeautifulSoup3 框架，数据库的开发基于 Mysql，各个部分所涉及的技术细节我们将会在第 5 章节进行详细介绍。

## 第4章 产品介绍

本章摘要：本章对师生互选助手的功能界面进行了详细的介绍，主要包括用户登录与导师信息搜索引导界面、个人基本信息与用户身份修改界面及个人基本信息与身份修改界面等 10 多个界面信息的介绍。

### 4.1 产品界面与功能总览

师生互选助手是一款实用、简洁、高效的微信小程序产品，为高校导师、硕博备考考生及准硕博新生提供了一个相互了解与交流的平台。通过这个平台，学生们可以更加全面的获取导师信息，导师们也可以更为方便的与学生进行交流。师生互选助手致力于成为导师信息查询及师生互选交流的一体化平台，让学生寻找心仪导师、导师寻找心仪学生的双向过程变的简单高效，系统的结构图（如图 4.1）如下：

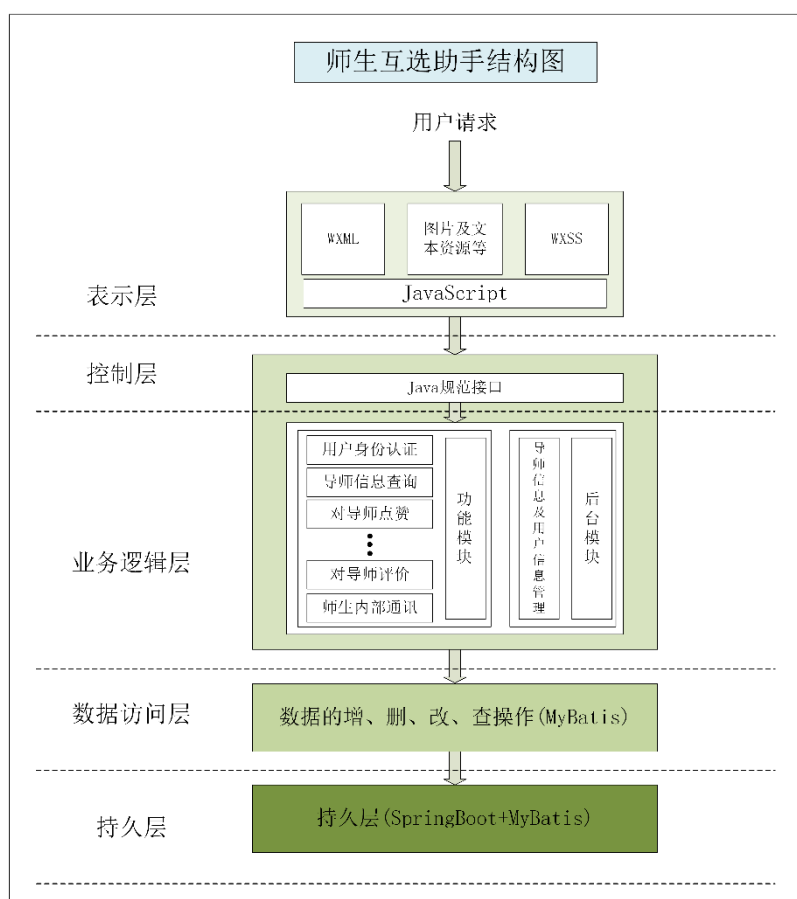


图 4.1 师生互选助手系统结构图

师生互选助手为老师和学生提供不同的功能入口。作为学生，目前可以搜索到目标院校的心仪导师信息。进入心仪导师的个人主页界面，系统提供了导师评价功能。你可以在评论区查看其它学生对该导师的评价，后台系统会筛选出相关的非负面评价在评价区展示。评价子系统的目的是为学生在导师选择时进行正面的引导，而非利用导师间的正负面评价反差对学生进行引导。在内部的通讯系统中学生可以向心仪的导师递交简历，发送私信。作为导师，您可以在评论区与学生们交流，你也可以查看学生们给您递交的简历和发送的私信，并可选

择性的回复。作为导师,如若想获得个人主页信息修改权限及使用内部通讯系统进行 1: N(一个导师对应多个学生)的交流,需在小程序内的人脸识别页面进行导师身份认证。导师评价系统的主要功能如下图 (如图 4.2):

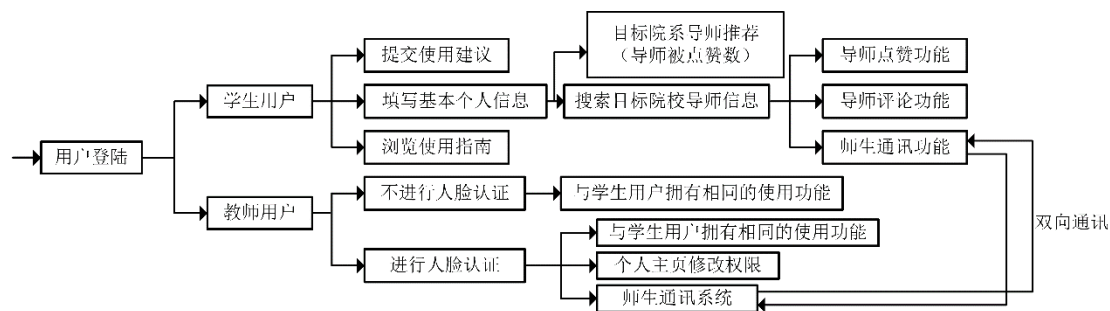


图 4.2 师生互选助手功能示意图

## 4.2 产品子界面与功能详细介绍

师生互选助手主要包含四个功能模块,具体为院校导师信息检索模块、导师身份认证模块、师生内部通讯模块及导师正面评价模块,其中包括小程序登陆界面、导师身份认证界面、小程序使用后用户体验提交界面、小程序功能介绍图示界面、导师检索界面、目标院系导师总览界面、心仪导师个人主页界面、导师评价和点赞界面及师生内部通讯界面等。在接下的子章节中,我们会对各子界面进行展示及功能介绍。

### 4.2.1 用户登录与导师信息搜索引导界面及功能

进入登陆界面 (如图 4.3 左子图),首先需要获取使用者的权限,接着进入身份确认界面。

如果使用者的身份是学生,那么进入程序后首先需要填写您的个人详细信息(假设使用者是第一次进入小程序情况下)。这样您才可以进入搜索引导界面 (如图 4.3 右子图) 及院校导师界面、进入导师的个人主页、查看导师详细资料、浏览评论区、给导师发送私信及上传简历。

如果您的身份是高校硕博导师,那么进入程序后首先需要进行人脸识别来核实您的身份。确认了您的身份后,您就可以进入个人主页,修改个人信息,在评论区与学生们互动,浏览学生们发送的私信以及学生递交的简历,并对前来联系的学生们选择性的进行回复。



图 4.3 用户登录与导师信息搜索引导界面及功能

#### 4.2.2 个人基本信息与身份修改界面及功能

用户登陆之后，导师评价系统需要您填写或修改您的个人资料。如果您的身份是学生，您需要填写自己的姓名、学校、专业院系等个人信息（如图 4.4 左子图）。如果您的身份是硕博导师的话，那么除了以上的基本信息，您还可以进入自己的主页修改个人信息（如图 4.4 右子图）。小程序内部会从您所在的院系、您的个人主页及您所在实验室主页三种途径将您的信息导入我们的小程序作为默认数据。我们欢迎通过身份认证的导师们在这个平台上修改自己的个人信息，让学生们了解到导师们最新的科研动向。



图 4.4 个人基本信息与身份修改界面及功能

### 4.2.3 使用帮助与意见反馈界面及功能

如果您在使用《师生互选助手》时对程序的功能或者操作流程有疑问之处，我们在主界面添加了使用指南（如图 4.5 左子图）。您可以点击进入查看小程序的详细操作流程，相信您在浏览过使用帮助之后可以更好的使用师生互选助手。

除此之外，如果您在使用师生互选助手的过程中有关于小程序的改进建议，请点击进入意见反馈界面（如图 4.5 右子图），填写详细的建议信息。无论是中肯的建议，还是专业的指导，或者只是想和我们取得联系，我们都十分欢迎。我们相信只有在老师和同学们的共同参与下，《师生互选助手》才能越做越好，才能真正成为老师及学生间交流联系的助手。



图 4.5 使用帮助与意见反馈界面及功能

### 4.2.4 导师信息搜索界面与功能

我们将当前收集到的相关院校导师信息存入了我们的数据库中，并开放了导师信息搜索功能。目前主要收录了湖南大学及中南大学的部分院系导师信息，后期我们会将全国高校的导师信息逐步导入我们的数据库。我们希望将来的导师个人数据会从下面几个主要途径获得：学校官网教师信息、导师个人学术主页或所在实验室主页、经过身份认证的导师本人自行添加修改及一对一联系。其中第四种途径是最为困难的，这也是我们希望数据专业化及全面化的重要途径。与个人信息缺失的导师进行一对一联系，以获取其信息并添加入我们的数据库中。学生在这个平台中能够高效找到心仪的导师，导师在这个平台上能够高效找到心仪的学生，便是师生互选助手发展建设的目标。

搜索界面（如图 4.6 左子图）中点击选择具体的学校、学院及专业后，我们会列出该专业所有的导师（如图 4.6 右子图），点击导师照片便可进入导师详细信息界面。



图 4.6 导师信息搜索界面及功能

#### 4.2.5 所选院系导师推荐界面

进入具体的院系之后，您可以看到这个专业所有的导师信息。如果您对这个专业相关导师的情况不是很了解，请不用担心，点击导师信息图标。点击图标后，我们会按照学生对本系的导师点赞数将导师进行排序，点赞越多的导师所在的序号越靠前。您可以以左右滑动的方式浏览各位导师信息，并且可以根据需要点击进入这些导师的个人主页进行更为细致的了解。如图 4.7 点击进入导师推荐界面，搜索框中确定你的目的院系，然后我们会将导师的信息展示在类似于滑动窗的控件里。利用导师获得的点赞数进行排序推荐，可为学生在选择导师时提供一些借鉴意义。



图 4.7 所选院系导师推荐界面



#### 4.2.6 所选导师个人信息界面与功能

进入导师的个人信息页面，我们为您提供了全面详细的导师信息，大致包括导师的研究方向、个人荣誉、联系邮箱、研究方向、已发论文、个人主页网址及已获项目等。这些信息足以让你对心仪导师有着一个基本的认识。本界面中的底部，我们添加了点赞、评论、内部通讯系统及导师信息分享四个功能接口（如图 4.8）。通过点赞数，我们可以基本上了解该导师在学生中的热度值，导师评论功能及内部通讯系统我们将在下面的小节中详细介绍。

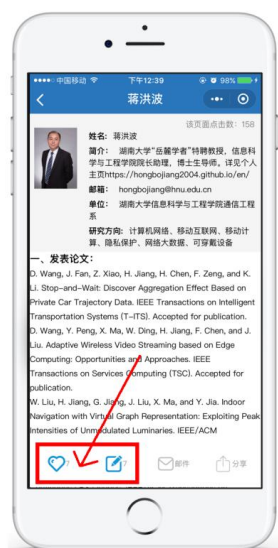


图 4.8 所选导师个人信息界面

#### 4.2.7 导师非负面评价界面与功能

师生互选助手小程序中，我们添加了导师评价子系统（如图 4.8）。学生可以对导师进行评价，但我们会对学生的评价进行筛选，以剔除负面及消极评价。剔除负面评价的原因在于，我们希望通过用对导师的正面评价为学生提供选择指引。越受学生欢迎的导师获得的评价越多，学生可依据评价系统的信息综合其他信息进行全方位考虑。如果您对这个导师有一定的了解的话，也欢迎您进行客观的评论，为其他同学在进行导师选择时提供参考。



图 4.9 导师非负面评价界面

#### 4.2.8 基于深度学习人脸对比认证界面与功能

微信小程序中，每一个微信号都有着唯一的标识码与其对应。为了导师的微信号与标识码实现绑定，我们使用了基于深度学习的人脸对比算法，这也是我们依据所掌握的导师信息进行标识码绑定的唯一可行办法。导师在登陆小程序的时候，可以选择进行人脸认证，以使用我们的内部通讯系统与学生进行交流。导师身份的认证，是实现内部通讯系统的基础。认证时，认证者需提交个人的基本信息（本人所在的学校、学院及姓名），系统根据提交的信息提取数据库中导师的真实照片，将验证者提交的前置摄像拍摄的照片（如图 4.10 左子图）与真实照片进行对比，便可获取认证结果（如图 4.10 右子图）。认证成功用户，我们将系统内用户的唯一标识码（我们设计的系统内部，也为用户设置了唯一标识码）换为该用户微信号所对应的标识码



图 4.10 基于深度学习人脸对比认证界面

#### 4.2.9 师生内部系统通信界面与功能

小程序内设计的师生通讯专用系统，具体的使用及介绍如下。师生双方可以通过该系统实现信息的交流及图片文件的发送。学生只需在心仪导师个人主页界面点击邮件图标，便可进入私信的使用界面（如图 4.11 左子图），填写完信息后点击发送即可。师生双方未读的消息数，会在个人主页界面的邮箱图标进行提示（如图 4.12 左子图），以防遗忘信息，导致双方的信息传送失败。同时可以通过点击删除按钮，清除已阅信息。

将学生的信息查询与师生之间的交流通讯，构建在同一个小程序中，无疑为师生双方带来了极大的便利。学生用户无需在不同学校的导师信息查询时，跳转于各个平台，节省了大量时间。教师用户获得了一个专用的招生平台，实现了与学生间的高效交流，而且无需下载任何插件及应用，该平台为其招生工作提供了极大便利。

考虑到老师在招生的过程中会给学生出一些题目来对学生的能力进行测试，因而在该通讯系统中，老师不仅可以查询到未读消息，还能查询所有历史聊天记录，这些消息将会快速为老师或同学回顾之前所交流的内容。



图 4.11 师生双方消息发送界面



图 4.12 未读消息提示及消息浏览界面

## 第 5 章 技术方案及细节

本章摘要：本章中主要介绍了师生互选助手功能实现中所涉及到的相关技术，其中主要包括前端技术、后台技术、数据库设计及数据爬取技术。

### 5.1 小程序中涉及到的前端技术

《师生互选助手》项目中，前端主要基于腾讯公司的微信小程序框架进行开发。本项目中主要包括身份认证、导师信息搜索、热门导师推荐及个人信息四大模块，在本章节中我们将对各子模块进行详细介绍。

#### 5.1.1 身份认证模块

在用户进入小程序时，需要对用户身份进行认证，老师身份与学生身份在系统中拥有不同的权限。用户认证流程如图 5.1 所示。

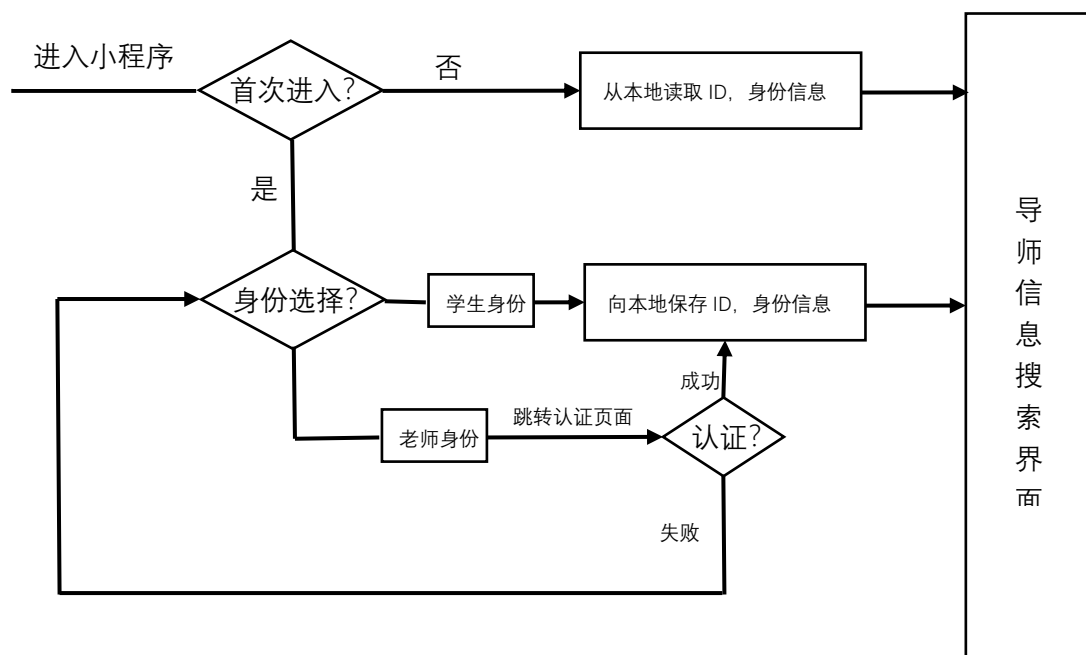


图 5.1 用户首次登陆身份认证流程

在身份选择模块中，主要涉及如下函数：

(1) `bindGetUserInfo ()` ;

该函数主要作用是从后台获取用户信息,并将获取到的信息保存至自己的数据库中，授权成功之后直接跳转到主页面，主要分为如下几个步骤：

- 🔧 发送 `res.code` 到后台换取 `openId`, `sessionKey`, `unionid`。
- 🔧 调用获取用户信息接口。
- 🔧 请求自己的服务器，解密用户信息获取 `unionId` 等加密信息。
- 🔧 解密成功后 获取自己服务器返回的结果，将获取到的信息保存到手机缓存或者全局变量中去，缓存有效期永久，除非用户删除。
- 🔧 插入登录用户的相关信息到数据库。

判断用户选择的类型跳转到不同的界面，老师→认证界面→主页面，学生→主页面。

(2) `sendEmail: function (id) ;`

该函数主要作用是向用户发送一条欢迎信息，目的是让用户了解如何使用该系统。

(3) `certifyTeacher: function (data);`

该函数主要完成导师身份认证功能，该函数向后台发送老师的 ID、姓名及单位信息，然后根据发送的照片以及数据库中存储的照片进行比对。比对结果以字符串形式返回，给用户可靠的提示与交互。

(4) `deletetempPic: function ();`

该函数主要的功能是向后台发送一条删除文件夹的指令，该文件夹是教师认证时所保存的两个本地照片文件。设计该函数的目的是为了后台照片数据太多而影响了认证的速度，另一方面也是为了保护用户隐私。

### 5.1.2 导师信息搜索模块

在导师信息搜索模块中，界面主要由轮播图部分、查询导师按钮部分和使用帮助与意见反馈三部分组成，该模块中所涉及的逻辑函数如下：

(1) `getStaticPhoto: function();`

该函数向服务器发送请求，获取主页面需要显示的照片，该照片是自主设计的漫画，用于介绍师生互选助手系统的功能。

(2) `searchByAffiliation: function (affiliation);`

该函数根据用户点击按钮选择的学校学院专业信息从数据库中加载这个系的所有老师的信息。

(3) `func_help: function ();`

该函数主要跳转到使用帮助页面，该页面主要介绍了该系统的使用介绍。

(4) `getStaticPhotoOfHelp: function ();`

该函数向服务器发送请求获取使用帮助页面需要显示的照片。

当用户选择查看一个老师的具体信息时候，就会进入 introduction 页面，在该页面中包含如下关键函数：

(5) `func_support: function(event);`

该函数绑定了点赞按钮，初次点赞数字加一，如果继续点击将使得点赞数减一。并且会根据用点赞数量情况显示不同颜色的 button，当点赞数为 0 的时候显示灰色的按钮，大于 0 的时候显示蓝色。

(6) `updateAccessCounts:function();`

该函数用于向数据库中更新老师页面的访问次数。

(7) `showTalks: function () {`

`this.loadTalks(); //用于加载数据`

this.animation.bottom("0rpx").height("100%").step();//设置动画内容为：使用绝对定位显示区域，高度变为 100%

```
    this.setData({
      showView: (!this.data.showView),//控制评论按钮图标消失，用于显示评论的界面
      talksAnimationData: this.animation.export()
    })
  }
}
```

该函数主要控制评论 dialog 出现与最消失，当用户点击点击按钮出现输入对话框，使得用户可以输入评论语句。

(8) func\_send: function (event);

该函数用于向后台提交评论结果。

(9) userInput: function (event);

该函数用于监测用户是否有输入文字，如果有的一旦有文字输入，则将发布按钮颜色变为蓝色，并且将文字内容记录到 message 变量中。

(10) fc\_deleteComment:function(event);

该函数为删除评论函数，若根据用户的 Id 来删除(这种情况下会出现一个 BUG：当你评论多次的时候，你会把所有的评论全部都删除。)更新之后根据用户的 ID 以及每条评论的评论时间组合起来达到唯一定位一条评论的效果。

(11) queryInfo: function (teacherId);

该函数按照老师的 ID 查询信息，目的是为了更新访问页面的次数。

(12) isPushedThumbs:function();

该函数用于判断用户是否点赞了。

(13) getThumbsCounts: function ();

该函数作用是获取点赞的数量。

(14) addAThumbs: function ();

该函数作用是插入一条点赞记录。

(15) deleteAThumbs: function ();

该函数作用是删除点赞记录。

(16) setEveryCommentsSupportCounts:function();

该函数用于遍历每一个同学评论对应的点赞数目。

(17) areYouSupportEveryComments:function();

该函数用于判断每条评论是否被点赞了。

(18) fuc\_giveCommentSupport:function(event);

该函数用于处理一条评论的点赞与取消点赞情况。如果没有点赞过那么你的点击会使得点赞数加 1, 如果你已经点赞过那么你会使得点赞的数量减 1。如果减 1 后总的点赞数为 0, 那么将会使得点赞图标的颜色换为灰色, 否则使用蓝色表示点赞数大于 0。

(19) `sendEmail: function (e);`

该函数处理用户点击了发送私信的按钮, 这时会把自己的 ID 老师的 ID 以及自己的昵称传送到发送私信的界面中去。所以用户只要在发送私信的页面中输入私信的内容以及建立的照片即可, 而不需要再输入目标对象的 ID, 也没必要知道因为那是一串 32 位的 openID。

在发送私信的页面 (`writeEmail`) 中, 主要涉及两个函数:

(20) `chooseImg: function ();`

该函数用于给用户添加照片信息, 可以是拍照也可以是直接从相册中选取。

(21) `sendEmail: function (data);`

该函数用于向后端发送一个请求, 请求内容为表单的形式, 表单中包含照片也包含发送邮件人的信息以及内容。但是当用户没有添加照片的时候, 则向后端发送一个不处理照片的普通 GET 请求。

### 5.1.3 热门导师推荐模块

在该模块中, 主要是展示热门导师的排序结果, 也就是访问次数多的老师将会优先推荐, 而被访问次数少的老师被推荐的可能性就会减少, 在该模块中主要涉及的函数及其作用如下:

(1) `func_detail:function(event);`

该函数用于处理用户点击了推荐导师的界面, 它会导航到这个老师的详细信息界面, 这与用户从主界面中搜索系之后再进入详细信息界面是一样的。

(2) `func_lisTopFive: function (affication) ;`

该函数根据用户选择的具体某个系之后展示这个系热门导师的排序结果。

(3) `searchTeachers: function (event) ;`

该函数用于根据用户在搜索栏中输入的信息进行搜索, 结果也会以热度排序展示, 用户可以具体目标院校的院系进行搜索, 也可以直接搜索某个专业进行查询。

### 5.1.4 个人信息模块

在个人信息界面用户可以看到自己的基本信息, 也可以看到未读私信内容数量, 当然也可以去修改自己的信息。在该页面中会根据自己授权的方式不同给与不同的权限, 如果你是一名学生你只有修改自己基本信息的权限。如果你是一名老师, 你将会拥有修改自己邮箱、发表论文及研究方向等相关信息的权限, 导师主页的信息也会被及时修改。在该模块中主要包含如下的函数:

(1) `getWxUserDetail: function ();`



该函数用于获取普通用户的具体信息，在 aboutme 页面中展示。

(2) updateUserInfo: function ();

该函数控制用户跳转到修改信息界面 (editInfo)，在修改信息界面中，会根据授权信息不同显示不同的展示内容，普通学生用于能修改的只是基本个人信息。而经过认证的教师用户可以修改的是自己展示给别人的信息，用户还可以在该页面中切换自己的身份信息。例如修改为老师身份，那么会跳转到 cetifyTeacher 界面中进行身份认证，认证通过即可切换；如果修改为学生身份，那么则会失去老师的身份以及相应的权限；在该页面中函数如下：

(3) handleTypeChange(e) ;

该函数用于监听用户身份转态，以及身份切换时对应的逻辑信息；

(4) handleSubmit();

该函数根据用户的身份信息，选择性的向服务器发送请求更改老师的个人信息或者学生的普通个人信息；

(5) myEmails:function();

该函数绑定用户点击私信按钮，将会跳转到自己收到所有私信的内容的缩略显示列表；

在 emailDetails 页面中主要函数如下：

(6) getAllEmails: function ();

该函数主要往后台发送请求，获取本 ID 对应的所有的私信内容。

(7) deleteEmail:function(event);

该函数监听用户点击删除私信按钮。根据用户点击的私信的 ID 删除这条私信，并且更新列表重新展示删除后的内容，同时向后端发送请求删除数据库中的这条私信记录。

(8) clearAll: function (event);

该函数用于清空该用户所有的私信，并且向后端发送请求删除该用户 ID 下的所有的私信记录。

(9) updateUncheckedEmails: function (fromId,told,sendTime);

该函数用于更新私信是否被查看的状态，它会在用户点击了查看私信细节之后被触发。

(10) eamilsDetailsInfo:function(event);

该函数用于绑定用户点击查看私信的具体内容，触发后会跳转到私信的具体信息页面 (eamilsDetailsInfo)，在该页面中用于可以看到其他用户发来的信息，自己也可以给发信息的用户回复信息。该页面中主要包含如下的函数：

(11) replyEmail: function (e);

该函数用于回复用户发来的私信，本质上是跳转到 writeEmail 页面中。

(12) deletetempPic: function ();

该函数用于向服务器发送删除临时文件夹的请求，该文件夹用于保存用于用户查看信息中的建立照片的；当用户退出查看之后就不需要了，下次查看信息时会继续创建。

(13) getUncheckedEmails:function();  
该函数用于向数据库中获取未读私信的数量。

(14) exit: function ();  
该函数用于清除个人信息，并退出登录。

## 5.2 小程序所涉及到的后端技术及数据库设计

《师生互选助手》项目后端主要采用 Springboot+MyBatis 框架+MySQL 数据库并采用了 MVC 设计模式进行开发；项目主要包含五个模块，对应于数据库中的五张表，下面将简单介绍 Springboot+MyBatis 框架，接下来对本项目 WEB 层各个功能进行介绍。

### 5.2.1 Spring boot 框架简介

Spring boot 是 Spring 家族中的一个全新的框架，它用来简化 Spring 应用程序的创建和开发过程，也可以说 Spring boot 能简化我们之前采用 SpringMVC+Spring+Mybatis 框架进行开发的过程。

在以往我们采用 SpringMVC+Spring+Mybatis 框架进行开发的时候，搭建和整合三大框架，我们需要做很多工作，比如配置 web.xml，配置 Spring，配置 Mybatis,并将它们整合在一起等，而 Spring boot 框架对此开发过程进行了革命性的颠覆，抛弃了繁琐的 xml 配置过程，采用大量的默认配置简化我们的开发过程。

采用 Spring boot 可以非常容易和快速的创建基于 Spring 框架的应用程序，它让编码变简单了，配置变简单了，部署变简单了，监控也变简单了。正因为 Spring boot 它化繁为简，让开发变得极其简单和快捷，所以在业界备受关注。Spring boot 在国内的关注趋势也日渐超过 Spring。

### 5.2.2 Spring boot 的特性

- ✚ 能够快速创建基于 Spring 的应用程序。(简化配置)
- ✚ 能够直接使用 java 的 main 方法启动内嵌的 Tomcat, Jetty 服务器运行 Spring boot 程序，不需要部署 war 包文件。
- ✚ 提供约定的 starter POM 来简化来简化 Maven 配置，让 Maven 配置变得简单。
- ✚ 根据项目的 maven 依赖配置，Spring boot 自动配置 Spring, SpringMVC 等其它开源框架。
- ✚ 提供程序的健康检查等功能。(检查内部的运行状态等)
- ✚ 基本可以完全不使用 xml 配置文件，采用注解配置。(或者默认约定的配置，代码中已经实现)

### 5.2.3 MyBatis 框架简介

MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。

MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或 注解 来配置和映射原生信息，将接口和 Java 的 POJO 映射成数据库的记录。

该框架简单小巧易于上手，方便浏览修改 SQL 语句，接触 SQL 与程序代码的耦合，提供映射标签，支持对象与数据库的 ORM 字段关系映射，提供 XML 标签，支持编写动态 SQL。MyBatis 是一个半自动化的持久化层框架。对于 JDBC 来说有如下的几个缺点，SQL 夹在 Java 代码块里，耦合度高导致硬编码内伤，维护不易且实际开发需求中 SQL 是有变化的，频繁修改的情况是多见的，Hibernate 和 JPA 长难复杂 SQL，对于 Hibernate 而言处理也不容易内部自动生产的 SQL，不容易做特殊优化基于全映射的全自动框架，大量字段的 POJO 进行部分映射比较困难，从而导致数据库性能下降。开发人员来说，核心 SQL 还是需要自己优化。SQL 和 Java 编码分开，功能边界清晰，一个专注业务，一个专注数据。

## 5.2.4 后端模块简介

微信用户模块，包含前端用于授权登录的后台逻辑，所涉及的主要功能如下：

- (1) @RequestMapping(value = "/getUserInfo", method = RequestMethod.GET)  
private Map<String, Object> getUserInfo(String uWxid)  
该函数用于控制 Service 从数据库中获取用户信息，返回到前端页面中。
- (2) @RequestMapping(value = "/getStaticPhoto", method = RequestMethod.GET)  
private Map<String, Object> getStaticPhoto()  
该函数用于处理微信前端（homeIndex）页面发送的请求图片指令。
- (3) @RequestMapping(value = "/getStaticPhotoOfHelp", method = RequestMethod.GET)  
private Map<String, Object> getStaticPhotoOfHelp()  
该函数用于处理微信前端（help）页面发送的请求图片指令。
- (4) @RequestMapping(value = "/modifyUserInfo", method = RequestMethod.POST)  
private Map<String, Object> modifyUserInfo(@RequestBody WxUserInfo wxUserInfo)  
该函数用于控制 Service 层修改数据库中的用户的个人信息。
- (5) @RequestMapping(value = "/insertWxUser", method = RequestMethod.POST)  
private Map<String, Object> addAreaByld(@RequestBody WxUserInfo wxUserInfo)  
该函数用于控制 Service 层往数据库中添加一条用于记录。
- (6) @RequestMapping(value = "/decodeUserInfo", method = RequestMethod.GET)  
@ResponseBody  
public Map<String, Object> decodeUserInfo(String encryptedData, String iv, String code)  
该函数会根据微信小程序前端发送过来的 encryptedData, iv, code 信息来解码用户的敏感数据；并将敏感数据封装到实体类中返回到前端去。

点赞记录模块，主要处理前端用于对老师或者对某一条评论的点赞情况，主要包含如下几部分功能：

- (1) @RequestMapping(value = "/isThumbsUp", method = RequestMethod.GET)  
private Map<String, Object> findRecords(String sourceId, String destId)

该函数根据源目 ID 从点赞表中查找记录，如果有的话，则说明你已经对这个老师点赞过了，则前端会加载已经点赞过的图标。

- (2) @RequestMapping(value = "/findRecordsCounts", method = RequestMethod.GET)  
private Map<String, Object> findRecordsCounts(String destId)  
该函数根据目的 ID 查找记录数目，有几条记录则说明老师被点赞数量是多少。
- (3) @RequestMapping(value = "/insertCounts", method = RequestMethod.POST)  
private Map<String, Object> insertCounts(@RequestBody ThumbsUpCounts thumbsUpCounts)  
该函数会往数据库中添加一条点赞记录。
- (4) @RequestMapping(value = "/deleteThumbsUpCounts",  
method = RequestMethod.GET)  
private Map<String, Object> deleteThumbsUpCounts(String sourceId, String destId)  
该函数会从数据库中删除一条点赞记录。
- (5) @RequestMapping(value = "/getCommentIdWithCounts", method =  
RequestMethod.GET)  
private Map<String, Object> getCommentIdWithCounts(String teacherUniqueId)  
该函数用于查找每条评论记录的点赞数目分别为多少。teacherUniqueId 用于区分是给老师点赞还是对这个老师下的评论点赞，假如 teacherUniqueId 不为空则代表是对这个老师下的评论点赞，如果 teacherUniqueId 为空则说明这条点赞记录是对应着老师个人的点赞。
- (6) @RequestMapping(value = "/insertCommentSupportCounts", method =  
RequestMethod.POST)  
private Map<String, Object> insertCommentSupportCounts(@RequestBody ThumbsUpCounts thumbsUpCounts)  
该函数会往数据库中添加一条点赞记录（对于评论）。
- (7) @RequestMapping(value = "/deleteCommentSupportCounts", method =  
RequestMethod.GET)  
private Map<String, Object> deleteCommentSupportCounts(String sourceId, String commentId)  
该函数会从数据库中删除一条点赞记录（对于评论）。
- (8) @RequestMapping (value="/isYourCommentSupported", method=  
RequestMethod.GET )  
private Map<String, Object> isYourCommentSupported(String sourceId, String teacherUniqueId)  
该函数查找某个人对每条评论记录的点赞数目分别为多少。如果数目大于 0，则说明他已经点赞了，结果返回 true，否则返回 false，表示他没有点赞。

老师信息模块,用于根据搜索关键字获取相关导师信息, 主要涉及的功能如下:

(1) @RequestMapping (value = "/findByAffiliation", method = RequestMethod.GET)

```
public Map<String, Object> findByAffiliation(String tAffiliation)
```

该函数向从数据库中根据老师所在单位 (tAffiliation) 字段获取所有老师的信息, 信息会根据访问量进行排序。

(2) @RequestMapping(value = "/findByTeacherId", method = RequestMethod.GET)

```
public Map<String, Object> findByTeacherId(String tId)
```

该函数根据老师的 ID 查询这个老师的详细信息。

(3) @RequestMapping(value = "/updateSupportCounts", method RequestMethod.POST)

```
public Map<String, Object> updateSupportCounts(@RequestBody Teacher teacher)
```

该函数会更新数据库中老师页面被访问的次数。

(4) @RequestMapping(value = "/modifyTeacherInfo", method = RequestMethod.POST)

```
public Map<String, Object> modifyTeacherInfo(@RequestBody Teacher teacher)
```

该函数用于修改数据库中老师的个人信息。

(5) @ResponseBody @RequestMapping(value = "/teacherCertification")

```
public Map<String, Object> teacherCertification(HttpServletRequest request, @RequestParam(value = "teahcerPhoto", required = false) MultipartFile[] files) throws Exception
```

该函数根据老师从前端传过来的一张自拍照, 以及根据前端传来单位信息和名字来从数据库中取出这个老师的公开证件照来使用百度 AI 人脸对比 API 进行对比, 结果分为五种:

- ✚ 未找到该老师的信息
- ✚ 前端照片为非活体照片
- ✚ 照片比对得分 (得分小于 80)
- ✚ 照片比对成功 (得分大于 80)
- ✚ 未知异常

私信记录模块, 主要用于获取用户的私信记录及删除相应的私信记录, 具体的功能如下:

(1) @RequestMapping(value = "/getAllEmails", method = RequestMethod.GET)

```
private Map<String, Object> getAllEmails(String told)
```

该函数根据 ID 从数据库中获取所有的私信返回给前端。

(2) @RequestMapping(value = "/deletetempPic", method = RequestMethod.GET)

```
private Map<String, Object> deletetempPic(String resumeUrl)
```

该函数根据前端传来的图片 URL 地址删除临时文件夹。

(3) @RequestMapping(value = "/emailsChecked", method = RequestMethod.POST)

```
private Map<String, Object> emailsChecked(@RequestBody EmailsContent emailsContent)
```

该函数用于获取一个用户所有私信查看状态。

(4) @RequestMapping(value="/removeEmail",method = RequestMethod.POST)  
private Map<String, Object> removeEmail(@RequestBody EmailsContent emailsContent)  
该函数根据 ID 删除数据库中一条私信的记录。

(5) @RequestMapping(value="/removeAllEmails",method = RequestMethod.GET)  
private Map<String, Object> removeAllEmails(String told)  
该函数根据用户 ID 删除该用户所有的私信记录。

(6) @RequestMapping(value="/uncheckedEmailsNum",method = RequestMethod.GET)  
private Map<String, Object> uncheckedEmailsNum(String told)  
该函数用于获取该用户所有未读私信的数量。

(7) @RequestMapping(value="/notifyInfo",method = RequestMethod.POST)  
private Map<String, Object> notifyInfo(@RequestBody EmailsContent emailsContent)  
该函数用于系统向用户发送一封通知的私信，该函数用于初次授权登陆时触发。

(8) @ResponseBody  
@RequestMapping(value="/writeEmail")  
public String writeEmail(HttpServletRequest request, @RequestParam(value =  
"resumePhoto", required = false) MultipartFile[] files) throws IOException  
该函数用于处理前端发送的一封带简历照片的请求，处理前端表单形式的请求。

评论记录模块，用于检索相关导师所获得的评价信息，主要功能如下：

(1) @RequestMapping(value = "/showComments", method = RequestMethod.GET)  
private Map<String, Object> showComments(String told)  
该函数用于查询某个 ID 下的所有的评论内容；

(2) @RequestMapping(value = "/addComment", method = RequestMethod.POST)  
private Map<String, Object> addComment(@RequestBody Comment comment)  
该函数用于向数据库中添加一条评论记录；

(3) @RequestMapping(value = "/removecomment", method = RequestMethod.POST)  
private Map<String, Object> removeComment(@RequestBody Comment comment)  
该函数用于删除数据库中的一条评论记录，根据用户的微信号+评论时间来定位一条评论记录。

### 5.3 小程序数据爬取涉及技术

师生互选助手中为了获得教师的基本信息，我们需要对相应教师所在院系的个人主页、实验室个人主页及其他相关网页的信息进行爬取。本次使用基于 python 的 request 库及 beautifulsoup4 库获取网页基本的信息，并对下载的 DOM 文档进行解析，获取到数据库表中相应的字段信息存入库中，爬虫的基本流程图如下：

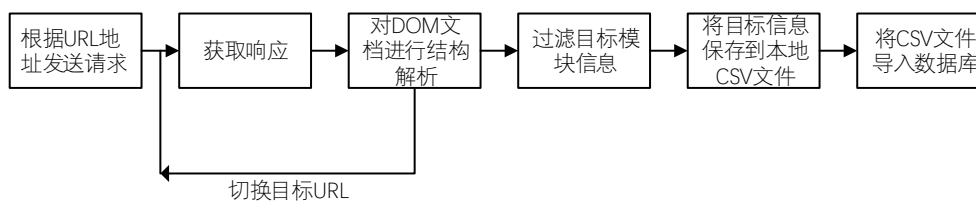


图 5.1 爬取数据流程图

为了保证导师的数据尽可能详细全面，在我们的系统中，我们选择从三个途径获取导师的基本信息，为用户提供优质的信息资源。在基本信息的基础上，我们与个人信息缺失的导师一对一联系，同时导师本人可对个人数据进行补充修改，这两个途径所获取的导师信息与原有的基本信息形成了互补。导师信息来源如图 5.2 所示：

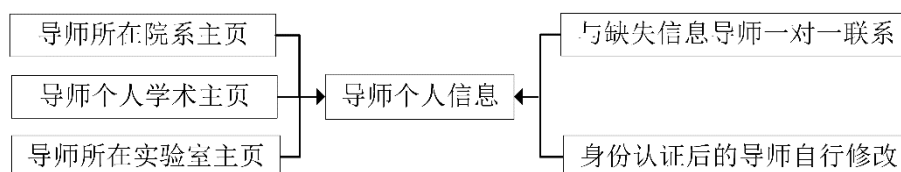


图 5.2 导师信息来源图

## 5.4 小程序数据库设计概述

师生互选助手中数据库一共包含五个表，分别为师生私信记录表（emails\_content）、学生用户信息表（user\_info）、教师用户信息表（teacher\_info）、点赞记录表（tb\_thumbsup）及评论记录表（user\_advice）。

师生私信记录表，用来保存师生双方通过内部通讯系统所产生的聊天记录。学生用户信息表用来保存学生的基本个人信息，其中包括学号、专业、姓名、导师等信息。教师用户信息表除了包含教师所属单位、姓名、个人照片等基本信息，通过身份认证的导师还保存了研究方向、项目列表及发表论文等基本信息。点赞记录表，用于记录相应导师被点赞的相关信息。评论记录表用来相应导师被评论的信息、时间及评论人 ID 等。为了能够详细并清晰的介绍师生互选助手的数据存储结构，详情请见表 5.1。

表 5.1 数据表名及其对应的内容信息

数据表名	信息构成
emails_content	发件人 ID、发件人昵称、发件时间、私信内容、查看状态及附件等信息
user_info	学生用户的学号、性别、昵称、学校、学院、专业及 openID 等信息
teacher_info	个人主页被访问次数、项目列表、姓名、单位、研究方向及 openID 等信息
tb_thumbsup	被点赞 ID、评论 ID、点赞人 ID、点赞时间等信息
user_advice	评论时间、评论人 ID、被评论人 ID、评论人昵称及评论内容信息

通过数据库的 ER 图（见图 5.2），我们可以快速的解读数据表之间的关联关系，从图中可得到表内包含的全部结构信息。



## 5.5 小程序数据库详细设计

### 表 5.2 师生私信记录表详细信息

### 表 5.3 点赞记录表详细信息



thumbsUpTime	bigint	20	否	是	点赞时间
--------------	--------	----	---	---	------

表 5.4 教师用户信息表详细信息

字段名称	数据类型	字段大小	是否为主键	允许空	字段说明
t_id	varchar	32	是	否	老师唯一标识码
t_name	varchar	20	否	否	老师姓名
t_affiliation	varchar	100	否	是	老师所属单位
t_email	varchar	100	否	否	老师邮箱
t_introduction	text	0	否	是	老师性别、职称及教育背景
t_researchInterest	varchar	500	否	否	老师研究方向
t_publishedPaper	text	0	否	是	老师发表论文
t_projects	text	0	否	是	老师研究项目
t_photo	varchar	255	否	是	老师照片
t_thumpUpCounts	int	11	否	是	该导师主页访问次数
t_detailLink	varchar	255	否	是	老师其他个人主页地址

表 5.5 学生用户信息表详细信息

字段名称	数据类型	字段大小	是否为主键	允许空	字段说明
u_wxid	varchar	30	是	否	用户微信号码
u_photo	varchar	255	否	否	用户微信头像
u_nickname	varchar	20	否	否	用户微信昵称
u_gender	varchar	5	否	是	用户的性别
u_city	varchar	15	否	是	用户所在城市
u_province	varchar	15	否	是	用户所在省
u_country	varchar	15	否	是	用户所在国家
u_school	varchar	30	否	是	用户所在学校
u_college	varchar	30	否	是	用户所在学院
u_department	varchar	30	否	是	用户所在系
u_studentId	varchar	20	否	是	学号

u_teacherName	varchar	20	否	是	导师姓名
---------------	---------	----	---	---	------

表 5.6 评论记录表详细信息

字段名称	数据类型	字段大小	是否为主键	允许空	字段说明
user_id	varchar	32	否	是	评论者 ID
advice_type	varchar	20	否	否	评论类型
advice_content	varchar	255	否	是	评论内容
advice_time	varchar	20	否	是	提交评论时间
advice_photo	mediumblob	0	否	是	评论截图

## 第 6 章 系统运维方案

### 6.1 运行维护目标

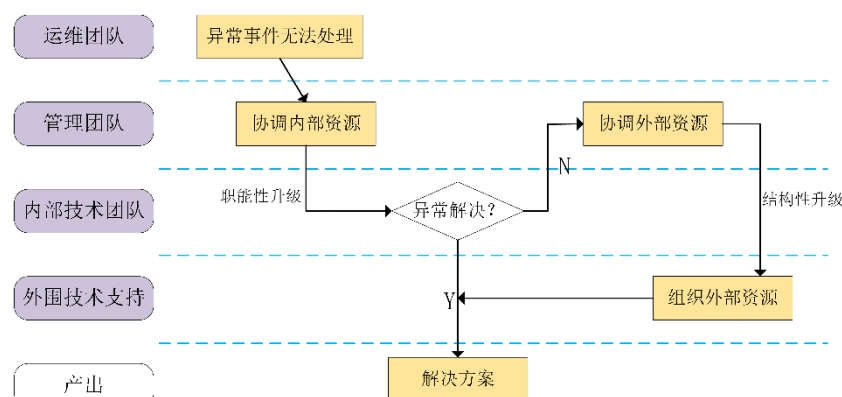
《师生互选助手》主要是为学生用户提供导师信息查询检索服务及为师生互选初期提供高效通讯服务。我们希望通过运行维护，小程序能够 7×24 的高效为用户提供服务。在用户进行检索时，能够快速及时的返回检索数据。在师生进行交流时，内部通讯系统能够完成数据的实时传输。

### 6.2 运行维护内容

- ✚ 由于提供导师详细全面的数据是小程序的重要部分，所以维护人员要定期对导师数据进行更新检查。
- ✚ 不断的对现有模块进行优化，主要包括界面交互性优化。
- ✚ 按照当前的相关领域发展趋势，为小程序添加新的模块，以提升小程序的实用性。
- ✚ 添加功能模块，以吸引潜在用户，并保留现有活跃用户。
- ✚ 不断完善用户的隐私保护机制，提高用户数据的安全性。
- ✚ 将用户的并发访问纳入考虑范畴，提升并发应对机制的能力。
- ✚ 及时对程序运行出现的问题进行报备处理。

### 6.3 程序异常事件处理流程

如果某一事件不能在规定的事件内由一线技术支持小组解决，那么就需要将问题提交给更有经验的人员进行处理。事件处理状态升级主要包括两类：职能性升级，需要具有更多时间、专业技能或访问权限的人员来参与时间的解决；结构性升级，当今授权的级别不能保证事件及时处理时，就需要更高级别的机构参与进来。但是在软件维护过程中，要尽可能保证问题在运维团队内解决，避免事态升级。程序异常事件的处理流程如下：



6.1 小程序异常事件处理流程图

## 第 7 章 总结及展望

### 7.1 功能拓展展望

目前，第一个版本的师生互选助手满足了用户所需的基本功能，可为硕博生备考者及准硕博生提供准确的导师信息，同时为师生间的交流提供了专业化平台。为保留现有的使用群体并扩大小程序的潜在使用群体，我们会对下一版本进行相应的功能拓展，具体的计划如下：

- 拟添加功能 1，实验室招生信息公布模块。已经过认证的导师可通过该区域发布硕博招生信息，主要包括所在团队的招生人数，招生要求及实验室相关的福利政策。该功能可帮助导师尽可能扩散本年度的招生计划择优进行学生选拔，同时学生群体可获取更多的导师信息。
- 拟添加功能 2，学生用户学习讨论模块。为不同的学校院系建立讨论区，学生们可以在该区域交流学习心得及学习经验，也可以在该区域共享相关的学习资料。在该模块中包括考研，读博和出国深造三大主题，为相关的学生提供信息及资源共享的平台。
- 拟修改功能 1，学生评价模块。当前的学生评价是由学生自己编写内容，然后经过后台审核才能进行评论内容展示。下一版本的师生互选助手，我们希望将导师评价设置为几个固定标签，如项目多、性格好、论文多及对学生好等。学生只需要选择几种固定的非负面评价标签，便可完成对该导师的评价。该模块的重新设计，既规范了评价的内容范围，同时可为学生用户提供导师选择时的良性引导。

### 7.2 《师生互选助手》推广方式

- 线下在各大高校进行宣传，关注小程序并转发朋友圈，送小礼品。
- 线上在各大论坛、贴吧及高校 BBS 上写软文进行广告宣传。

### 7.3 小程序总结与展望

在本次小程序的前期创意构思及后期的总体架构搭建过程中，我们的团队对每一步的细节都进行了多次的讨论。立足于研究生备考群体及准研究生所面临的，找导师完整个人信息困难和师生初步互选交流缺乏平台两大问题，我们完成了导师互选助手的设计。提供高价值的信息资源和便捷式的交流平台无疑会给目标用户带来良好的体验。在保证当前用户量的同时，我们会选择合适的主题模块进行小程序的功能拓展，去不断的发掘潜在的用户。

我们致力于将本款产品打造为功能完善，交互友善的一体化硕博信息获取及信息交流平台。希望借助微信小程序这种新颖的方式，将我们的设计理念及现有产品得以推广，希望更多的学生通过我们的平台找到心仪的导师，希望更多的团队及导师通过我们的平台招收到优质的学生。