目录

[第一章 引言 2](#_Toc39522905)

[1.1 课程背景和意义 2](#_Toc39522906)

[1.2 课题任务 2](#_Toc39522907)

[1.2.1 课题内容 2](#_Toc39522908)

[1.2.2 本人承担任务及方案选择 3](#_Toc39522909)

[1.3 论文结构 4](#_Toc39522910)

[第二章 系统的需求分析 4](#_Toc39522911)

[2.1 系统用户角色分析 4](#_Toc39522912)

[2.1.1 普通用户分析 4](#_Toc39522913)

[2.1.2 管理员用户分析 4](#_Toc39522914)

[2.1.3 超级管理员用户分析 4](#_Toc39522915)

[2.2 系统功能需求分析 5](#_Toc39522916)

[2.3 系统非功能需求分析 7](#_Toc39522917)

[2.4 本章小结 7](#_Toc39522918)

[第三章 系统的总体设计 8](#_Toc39522919)

[3.1 系统的结构设计 8](#_Toc39522920)

[3.2 系统功能模块设计 9](#_Toc39522921)

[3.3 系统的数据库设计 12](#_Toc39522922)

[3.3.1 系统数据库的E-R模型 12](#_Toc39522923)

[3.3.2 系统数据库表 13](#_Toc39522924)

[3.4 系统界面设计 16](#_Toc39522925)

[3.5 本章小结 18](#_Toc39522926)

[第四章 系统详细设计与实现 19](#_Toc39522927)

[4.1 用户管理模块的详细设计和实现 19](#_Toc39522928)

# 第一章 引言

## 课程背景和意义

高校招生，是指国家高等教育院校按照教育部高校招生分类进行招收新生的工作。而其中的全国统招这一招生类别，更是大多高等教育院校的主要新生来源。高校的生源质量直接影响着学校的发展和办学质量，更进一步，其更是影响着国家的人才储备。正因为其重要性，高校的招生工作备受高校、考生、社会的关注。

招生咨询工作是高校高招工作中不可缺失的环节，也是一次考生和家长能够直接与高校对话的机会。对于高校而言和负责招生咨询的工作人员而言，招生咨询工作是一个宣传自己高校的机会；是了解当年该省分数分布、预估分数线的重要手段之一；是帮助学校吸引优秀人才的重要方式。对于考生家长而言，这是他们在填报志愿之前，可以直接与来自各个高校各个专业的老师直接对话的机会。考生和家长都十分渴望能借助招生咨询的机会，听取老师关于报考和专业选择等方面的建议。为作为高校老师，也很希望能够通过已有的信息，给予学生力所能及的帮助。结合学生成绩，分数线等信息，综合全面给予考生有针对性的建议和宣传。

然而，现阶段招生咨询工作缺少电子手段的支撑，导致招生老师不得不使用较为传统，效率较低的方式进行信息的收集和分析。这不仅消耗了老师大量的精力，还会导致信息滞后、不准确，宣传口径出现偏差等情况。进而导致不能在第一时间给予考生最准确的建议，使得合适的考生资源流失。很大程度上限制了招生咨询工作的准确度和效率。

本文的目标是开发一套包括数据收集，数据分析，数据展示，资料展示的招生咨询系统。通过电子手段对数据进行储存、分析和处理，可以克服很多人工处理的弊端，例如：手写的纸质信息容易丢失，不宜整理、上报数据过程繁琐、分析数据困难等，具有实用价值。利用考生在系统中填报的信息，以及系统内置算法，可以在几秒钟内迅速获取当年分数与排名关系，录取分数线等招生咨询工作中的重要信息，辅助招生咨询工作组成员进行宣传口径的设计。让老师在数据收集和处理方面更加轻松，使得老师能将更多的精力放在招生宣传工作上，进而给考生和家长们留下良好的印象，更有针对性的完成招生咨询工作。

## 课题任务

### 1.2.1 课题内容

本课题的整体目标是开发一套涵盖手机，PC设备的招生咨询系统，使招生咨询工作中所有的参与对象能够通过系统得到有效地组织，沟通和支撑。

本文是基于Python语言设计的在PC端使用的招生咨询系统的一个子系统，系统的功能模块设计如下：

1）用户管理模块：

a、注册功能：用户需要输入用户名，密码，姓名进行账户注册。

b、登陆功能：用户需要输入用户名密码登陆系统。

c、用户信息完善功能：让用户在登陆之后填写自己的电话，咨询省份，咨询地点等信息。

d、 权限管理：系统提供普通用户，管理员，超级管理员三种权限，权限向下兼容。某些界面或功能需要特定的权限才能使用。

2）信息收集模块：

a、考生信息收集功能：在招生咨询现场可以收集考生的基本个人信息，包括姓名、考号、分数、位次、电话、毕业中学、文理科、拟报专业、志愿次序信息，而后进行验证，验证通过后根据登陆用户的信息及系统时间自动生成录入人、录入省份、录入地点、录入日期、录入时间信息并进行存储。

b、专业历年录取排名收集：各省的管理员用户（招生咨询组组长）需要录入本校各专业在各省历年录取的排名，系统将以此作为依据，预测今年的录取分数，以提供给招生咨询组组长作为确定宣传口径的依据

3）数据处理模块

a、分数排名关系预估：通过考生录入的分数与自己的排名，筛选出同分数中分数最高和分数最低的排名，进行显示。

b、录取分数线预测：根据预测的排名分数关系以及招生咨询组组长录入的历年排名数据，寻找历年排名在今年所对应的分数，给出今年的分数预估情况。

4）展示模块

a、数据展示：所有数据均有对应的页面进行展示，页面为表格形式，所有条目可以正序或逆序排序，用户可以在展示页面对数据进行修改。其中考生信息提供导出excel报表的功能。

b、招生电子资料展示：展示电子版的招生宣传材料。

### 1.2.2 本人承担任务及方案选择

本人承担的任务是招生咨询系统中的python子系统的设计与实现。在展示方式上，本人选择了使用web网页形式进行展示，理由是首先基于python的web开发拥有多种较为成熟的开发框架，在开发逻辑上更为清晰；其次，web应用提供了互联网这个天然的传输手段，而且不需要使用的用户单独安装任何附加的应用，从实用性角度，相较封装客户端更为便捷。

在Web开发框架的选择中，本人选择了Django开发框架。Django开发框架遵循MVC设计模式，由于强大的第三方插件支持，其具有很强的扩展性。最重要的是，相较于强调简约的Flask轻量级开发框架，使用Django开发逻辑更加清晰。

了解并熟悉招生咨询工作的流程和细节，明确了传统信息收集和分析方法弊端，并着重进行改进。收集阅读有关招生咨询以及python web应用开发等方面的相关资料和文献，学习Django开发框架。完成系统的需求分析，概要设计，详细设计，编码实现，单元测试，系统测试等工作，并最终完成论文的撰写。

## 论文结构

# 第二章 系统的需求分析

## 2.1 系统用户角色分析

本系统需要实现一个招生咨询系统，对应现实的招生咨询工作中的三类使用者设计系统用户角色，即学校各省招生宣传组成员、各省招生宣传组组长、招办负责人。这三类角色分别对应本系统中的普通用户、管理员和超级管理员。其权限从低到高并且向下兼容。具体用例图如2-1所示。其详细功能分析如下：

### 2.1.1 普通用户分析

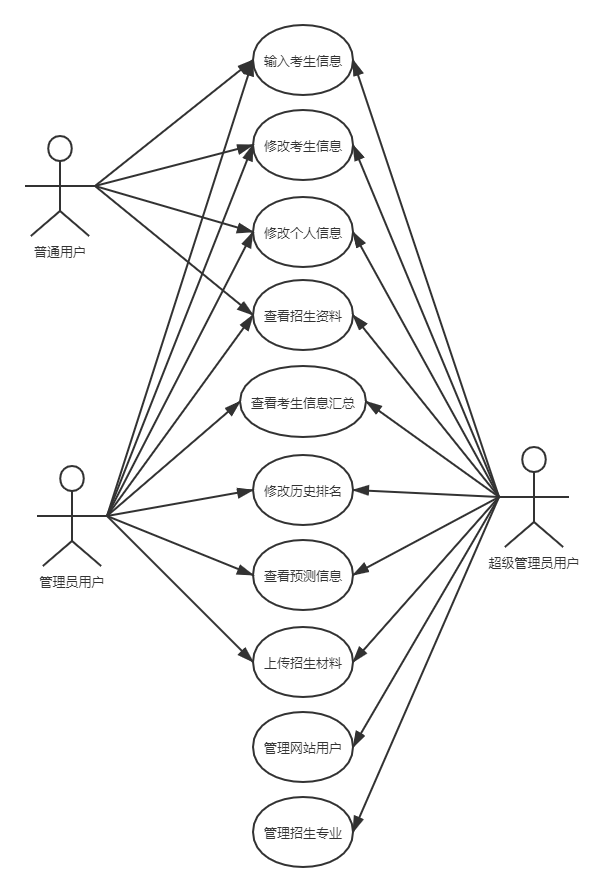
普通用户是指学校的招生宣传组组员，其主要负责在招生咨询会的现场收集考生资料，并且回答考生及考生家长的各类问题。普通用户可以通过注册和登陆获取访问招生咨询系统的权限。系统授予普通用户收集考生数据，查找考生数据，修改考生数据，查看招生材料，修改招生咨询省份和地点的权限。

### 2.1.2 管理员用户分析

管理员用户是指学校派至各省的招生宣传组组长，其主要负责确定招生咨询口径。管理员权限需要在注册后由超级管理员开启。管理员除了拥有普通用户所有权限外，还有查看所有考生信息，修改历史录取排名，查看系统预测信息，上传招生资料的权限。

### 2.1.3 超级管理员用户分析

超级管理员是指学校招生办的负责老师，其主要责任是管理涉及全网站的数据。超级管理员由后台生成，同时超级管理员无法修改其他超级管理员的信息。超级管理员除了拥有管理员所有权限之外，还具有管理网站用户，管理招生专业信息的权限。



#### 图2-1 系统用例图

## 2.2 系统功能需求分析

招生咨询系统的功能主要分为五个部分：用户管理、信息收集、信息管理、分析预测、信息展示。

用户管理的功能是让用户注册账号或使用已有的账号进行登陆，以及填写或更改个人信息。以及系统中一些界面的权限限制功能。

信息收集的功能按照数据进行分类，有考生数据，专业历史排名数据两类。采集考生数据需要老师在招生咨询现场引导考生在系统中填报自己的一些基本信息；采集专业历史排名数据需要招生组组长在招生咨询工作开展之前，在系统中填报所有在本省招生专业的历史排名信息。数据采集后，系统会将考生信息数据进行核查，将数据存放到数据库中。

信息管理功能需要用户对数据库中的数据能够进行增、删、改、查这四个基本操作。

分析预测功能可以将数据库中收集到的数据进行分析。系统主要需要分析预测出两个信息：一是根据考生填报的排名信息分析出分数排名之间关系，二是综合历史信息和分数排名关系，分析预测今年各个专业的录取分数线。

信息展示的功能需要系统能够通过web界面将信息展示给用户。其中由于本系统展示数据大多为表格类型数据，为方便查看，展示数据中包括排序功能。所收集的考生信息数据提供下载功能，允许用户将数据下载到本地进行处理。

系统功能需求表见表2-1所示

#### 表2-1 系统功能需求表

|  |  |  |
| --- | --- | --- |
| 需求 | 需求的功能 | 功能描述 |

|  |  |  |
| --- | --- | --- |
| 用户管理 | 注册登录 | 使用户在本系统内注册个人账户或使用已经注册的账户进行登陆。账户主要用于分辨用户的权限等级。 |
| 完善个人信息 | 用户登陆账户后，需要填写招生咨询省份、地点、电话等个人信息。 |
| 修改个人信息 | 用户可以随时通过修改个人信息的功能，对已经填写的信息进行修改。 |
| 页面权限限制 | 某些页面具有访问权限，拒绝没有权限的用户访问这些页面。 |
| 信息收集 | 获取数据 | 通过提交表单的方式获取系统所需的数据，并将数据保存至数据库中。 |
| 校验数据 | 考生信息数据将通过系统的查重，对于考号姓名均一致的数据系统将不再记录。 |
| 信息管理 | 增加数据 | 允许有权限的用户向对应的数据中加入新的数据。 |
| 删除数据 | 允许有权限的用户向对应的数据中加入新的数据。 |
| 修改数据 | 允许有对应数据库操作权限的用户对数据库中信息进行修改。 |
| 查找数据 | 允许有对应数据库操作权限的用户对数据库中信息进行查找。 |
| 分析预测 | 分数排名对应关系分析 | 通过系统采集的考生信息数据，找出同分情况下排名最高和最低的两个数据，并将其存入数据库 |
| 录取分数线预测分析 | 通过系统预估的排名与分数对应关系，以及采集到的历史录取排名数据，找出在本年分数排名最接近历史排名的分数区间，将该区间存入数据库。 |
| 信息展示 | 数据可视化 | 通过web页面向用户展示数据库中数据 |
| 数据排序 | 在展示数据页面，针对表格类型的数据，根据用户要求进行任意条目的正序或逆序排序。 |
| 下载报表 | 允许用户将采集到的考生信息数据下载excel格式的文件到本地，在本地进行查看和处理。 |

## 2.3 系统非功能需求分析

本系统计划部署在服务器中，用户通过浏览器访问实现在PC端使用。本系统使用B/S框架，即将系统的核心实现部分，如服务器程序和数据库均在后台进行运行，用户使用浏览器同数据库进行数据交互。作为招生咨询系统，普通用户的界面会展示给所有前来咨询的考生和家长。因此在界面美观方面，本系统的所有界面均使用一套样式，保证跳转界面不会显得很突兀。同时在普通管理员的首页上，除了常用功能链接外，还有加入了登陆用户的联系方式以及学校照片，方便考生及家长联系询问以及了解学校。可扩展性方面，本系统的核心逻辑使用Python语言编写，如需要系统升级维护等工作，只需要对服务端进行修改，而不需要每个用户进行额外操作。因此在这方面，本系统的升级维护成本十分低廉。

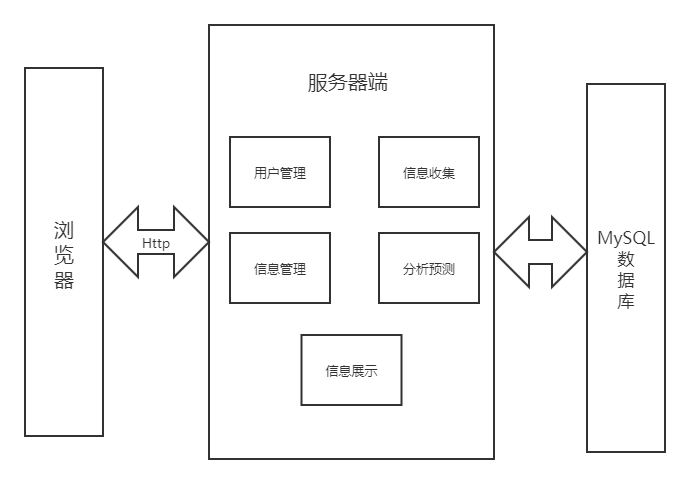
## 2.4 本章小结

本章进行了系统功能分析，分别从用户角度，系统功能需求和系统肺非功能需求角度进行了研究。为进行系统整体设计做好了准备。

# 第三章 系统的总体设计

## 3.1 系统的结构设计

由上一章中的非功能需求分析可知，本系统遵循B/S框架结构，系统的核心程序部署在服务器端上，而客户通过PC机的浏览器通过合同谈判请求获取服务器的信息，而后显示在浏览器上。数据库与服务器端建立连接，实现服务器与数据库的交互功能。系统结构设计图如3-1所示。



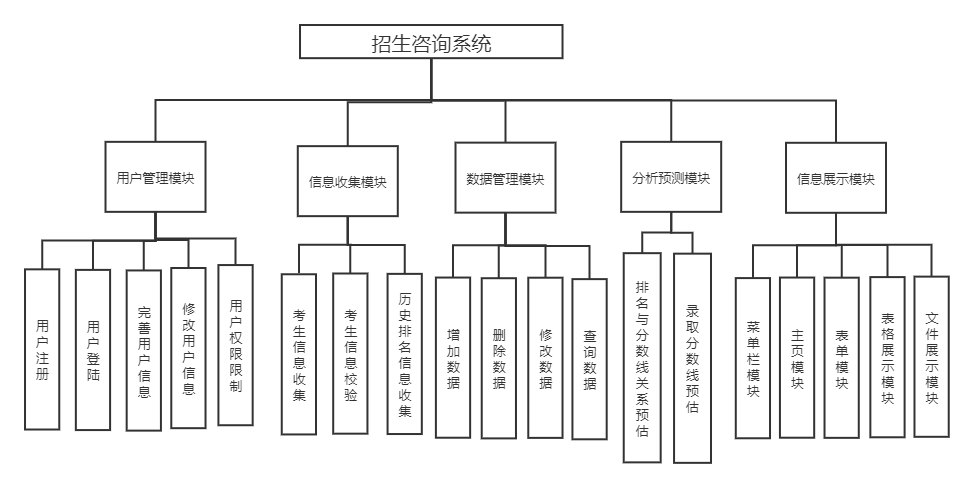
#### 图3-1 系统结构设计图

用户使用本系统时，只需要在浏览器中输入特定的地址，浏览器会通过http请求方式向服务器端发送请求数据，收到服务器端的回应后，浏览器会将完整的页面呈现给用户。

服务器端包含系统的五个功能模块，即：用户管理模块、信息收集模块、信息管理模块、分析预测模块和信息展示模块。用户管理模块在注册完成注册后将用户信息存入数据库；在用户登陆，跳转页面时请求数据库中数据进行判断，并将判断结果返回给用户。信息收集模块获取用户通过浏览器提交的表单信息，并将数据存储到数据库。信息展示模块收到请求时，会通过数据分析模块获取经过分析的数据信息，而后返回给用户。信息管理模块获取用户对数据库操作的请求，并对数据库中数据进行对应的修改并将结果返回给用户。

## 3.2 系统功能模块设计

根据之前系统功能需求分析的内容，将系统功能分为用户管理、信息收集、信息管理、分析预测和信息展示这五个功能模块。每个功能模块下又细分多个子模块，系统功能模块设计图如图3-2所示。



#### 图3-2 系统功能模块设计图

用户管理模块主要实现用户登陆注册，详细信息完善修改和跳转时权限验证功能。注册时，用户输入用户名密码姓名等信息，后台检测用户名是否被占用，检测通过后将用户信息存入数据库；登陆时后台获取前端用户所填写的用户名和密码，并与数据库中信息进行比对，比对成功后允许用户访问系统的其他功能；用户首次进入系统时，系统自动弹出完善个人信息界面，要求用户填写个人手机号，咨询省份及地点等信息；用户也可以通过修改个人信息功能随时对这些信息进行修改；当用户进行网页跳转时，权限管理功能会询问数据库判断用户是否有目标页面的访问权，如果有则返回目标页面信息，如果没有则返回404页面。

信息收集模块主要实现考生信息收集和校验，以及历史排名信息收集。用户在浏览器的表单中填写考生的基本信息并传入服务器端，服务器收到信息后检查表单各个数据是否合法，而后通过校验模块检查改考生是否第一次填写。如果是则存入数据库返回成功，如果不是则返回失败并抛弃数据。历史信息收集模块是向管理员用户询问各个专业的历史录取排名，管理员用户填写后，服务器端将数据存入数据库存储。

数据管理模块主要实现对数据库的进行增删改查的操作。实现人为对数据库数据进行管理。

分析预测模块实现系统对数据进行分析的逻辑。当用户访问预测信息页面是，该模块会向对应数据库请求数据，通过服务端的算法计算出预测结果，并将预测结果返回给用户。分析预测模块主要实现两种信息的分析预测：分数与排名关系的预测以及录取分数线预测。

信息展示模块主要实现向用户战士的交互界面，包括菜单，主页表单和数据库信息展示。其中数据库信息展示分为表格展示和文件展示模块两种。其目的是实现友好的交互界面，让用户操作起来更加简便。

本系统的功能模块设计描述表如表3-1所示。

#### 表3-1 系统功能模块设计描述表

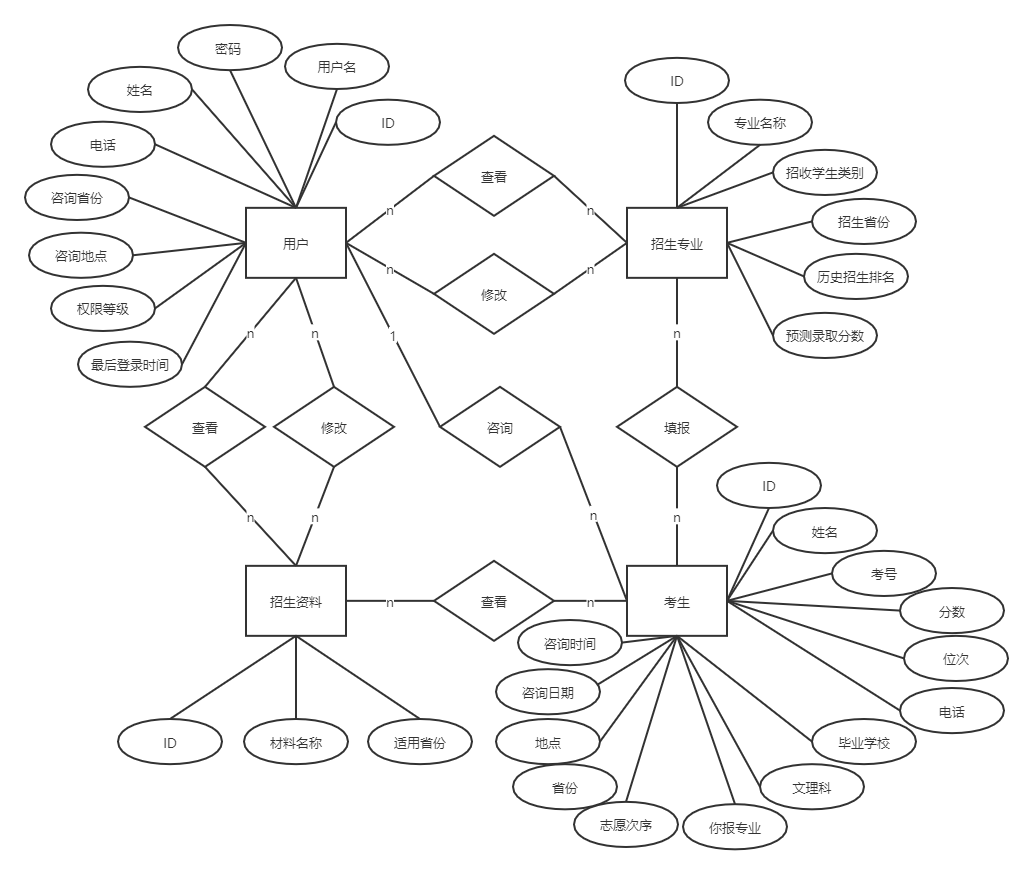
|  |  |  |
| --- | --- | --- |
| 模块 | 功能名称 | 功能描述 |

|  |  |  |
| --- | --- | --- |
| 用户管理模块 | 用户注册 | 用户在本系统中注册一个新的普通用户账号，系统检测账户名是否被占用。 |
| 用户登录 | 用户使用已经注册过的账号进行登陆，系统检测账号密码是否匹配。 |
| 完善用户信息 | 要求初次登陆系统的用户填写电话，咨询场地，咨询省份等更细致的个人信息，并保存进数据库。 |
| 修改用户信息 | 用户可以修改自己已经填写过的个人信息。系统会将修改后的信息存入数据库。 |
| 用户权限限制 | 用户进行页面跳转时，服务器端通过向数据库请求用户权限信息，判断用户是否有权限跳转至目标页面。 |
| 信息收集模块 | 考生信息收集 | 用户可以通过让考生填写表单的形式收集考生的各类基本信息。表单传到服务端后，服务端检测表单各项是否符合规范，而后向数据库请求用户信息，并将咨询地点、省份、时间等信息自动加入条目，保存至数据库中。 |
| 考生信息校验 | 保存考生信息时，系统需要检查数据库中是否有重复信息，如有则返回保存失败提醒。 |
| 历史排名信息收集 | 管理员用户向系统输入各个省份历史排名信息。系统将数据存入数据库。 |
| 数据管理模块 | 增加数据 | 用户可以通过此模块向任意数据库中人为增添数据。 |
| 删除数据 | 用户可以通过此模块删除数据库中数据。 |
| 修改数据 | 用户可以通过此模块修改数据库中数据 |
| 查询数据 | 用户可以通过此模块查询数据库中数据 |
| 分析预测模块 | 排名与分数线关系预估 | 当用户请求响应页面时，系统向数据库请求所有考生填报的信息，通过算法处理后预估出排名和分数线关系，存入数据库并展示。 |
| 录取分数线预估 | 当用户请求响应页面时，系统向数据库请求排名与分数关系预估和历史排名信息，通过算法处理后预估各个专业的分数线，并存入数据库并展示。 |
| 信息展示模块 | 菜单栏展示 | 菜单栏位于网页上方，按照用户级别分为普通用户菜单栏，和管理员用户菜单栏。菜单栏包括所有向用户开放的功能连接，方便用户随时跳转到需要的功能界面。 |
| 主页展示 | 主页根据用户级别分为普通用户主页和管理员用户主页。普通用户主页包括考生信息填报功能连接、招生材料展示连接、学校风光以及咨询老师姓名和联系方式；管理员用户主页包括管理员所拥有的全部功能。 |
| 表单展示 | 所有用户向服务端提交的信息均由表单的形式提交。系统根据用户进入的页面显示相应的表单信息让用户进行填写。 |
| 表格数据展示 | 所有数据库表的展示均为表格方式展示，服务器端筛选获取相应数据后以表格形式返回给用户。同时所有表单可以按照任意列的信息，进行排序功能。 |
| 文件数据展示 | 所有招生材料以pdf形式进行展示，服务端从数据库中提取pdf名称信息，拼接pdf存储的相对地址返回给用户，让用户的浏览器进行对应的显示。 |

## 3.3 系统的数据库设计

### 3.3.1 系统数据库的E-R模型

系统数据库设计采用实体-联系模型。本系统实体包括：系统用户、考生，招生专业，招生材料四个。系统用户实体类包括以下属性：唯一的ID、用户名、姓名、密码、电话、咨询省份、咨询地点、权限等级、最后登录时间。考生实体类包括：唯一的ID、姓名、考号、分数、位次、电话、毕业中学、文理科、拟报专业、志愿次序、咨询省份、咨询地点、咨询日期、咨询时间。招生专业实体类包括：唯一的ID、专业名称、招收学生类别（文科/理科）、招生省份、历史招生排名、预测录取分。招生材料实体类包括：唯一ID、材料名称、适用省份。

系统的数据库E-R图如图3-3所示。

#### 图3-3 系统数据库E-R图

### 3.3.2 系统数据库表

本项目使用MySQL数据库进行数据存储。所创建的数据库表较多，其中两类数据库表在这里不做介绍。首先，为了尽量减少存储的冗余数据项以节省空间，在数据表设计中包含了例如：省份信息（属性包括：ID，省份名称）、文理科信息（属性包括：ID，文科/理科）等内容相似，结构简单，属性少的表。其次，是Django自动创建的大部分表，例如：数据库迁移记录表、会话记录表等。这些表信息主要用于记录使用该数据库的日志信息，展示于Django自带的管理界面。

以下简单介绍本人为实现系统功能创建的较为重要的数据库表：

user用户信息表：该表有Django框架自动创建，属性未作修改，因此存在一些不使用的属性，在此不做陈述。该表的属性包括：ID（主键）、username 用户名、password 密码、first\_name 姓名（考虑到分为姓、名存储意义不大，因此使用该属性存储姓名，last\_name为空）、last\_login 最后登录时间、is\_superuser 是否为超级管理员、is\_staff 是否为管理员、is\_active 是否准入、date\_joined 注册日期。该表的设计见表3-2所示。

#### 表3-2 user表的设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性中文名称 | 属性英文名称 | 存储数据类型 | 长度 | 是否为空 | 备注 |
| 序号 | ID | int | 10 | 否 | 主键 |
| 用户名 | username | varchar | 150 | 否 |  |
| 密码 | password | varchar | 128 | 否 | 存储散列值 |
| 姓名 | first\_name | varchar | 30 | 否 |  |
| 最后登陆时间 | last\_login | datetime | 6 | 否 |  |
| 是否为超级管理员 | is\_superuser | tinyint | 1 | 否 |  | |
| 是否为管理员 | is\_staff | tinyint | 1 | 否 |  |
| 是否准入 | is\_active | tinyint | 1 | 否 |  |
| 注册日期 | date\_joined | datetime | 6 | 否 |  |

Userprofile用户信息扩展表：由于Django自带的用户信息表不能完全满足系统需求，本人采用一对一链接的方式扩充用户个人信息表。该表的属性包括：user\_ID（主键，与user表链接的外键）、telephone 用户电话、province 咨询省份（与省份信息表链接的外键）、place 咨询地点。该表的设计见表3-3所示

#### 表3-3 userprofile表的设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性中文名称 | 属性英文名称 | 存储数据类型 | 长度 | 是否为空 | 备注 |
| 用户序0号 | user\_ID | int | 10 | 否 | 主键、user表外键 |
| 电话 | telephone | varchar | 20 | 否 |  |
| 咨询省份 | province | int | 10 | 否 | province表外键 |
| 咨询地点 | place | varchar | 50 | 否 |  |

Stuinfo考生信息表：本数据库表是记录考生基本信息表，包括以下属性：ID（主键）、name 姓名、testnum 考号、score 分数、rank 排名、tele 联系方式、high\_school 毕业高中、application\_rank 志愿顺序、place 咨询地点、tip 备注、date 日期、time 时间、sciorart\_id（文科或理科表外键） 文科/理科、is\_international 是否报考国际学院、major1\_id（专业信息表外键） 第一志愿专业、major2\_id（专业信息表外键，可为空） 第二志愿专业、major3\_id（专业信息表外键，可为空） 第三志愿专业、province\_id（省份信息表外键） 咨询省份、staff\_id（用户user表外键） 录入人员。该表的设计见表3-4所示：

#### 表3-4 stuinfo表的设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性中文名称 | 属性英文名称 | 存储数据类型 | 长度 | 是否为空 | 备注 |
| 序号 | ID | int | 10 | 否 | 主键 |
| 姓名 | name | varchar | 20 | 否 |  |
| 考号 | testnum | varchar | 14 | 否 |  |
| 分数 | score | smallint | 5 | 否 |  |
| 排名 | rank | smallint | 5 | 否 |  |
| 电话 | tele | varchar | 20 | 否 |  |
| 毕业高中 | high\_school | varchar | 30 | 否 |  |
| 志愿顺序 | application\_rank | smallint | 5 | 否 |  |
| 咨询地点 | place | varchar | 30 | 否 |  |
| 备注1 | tip | varchar | 200 | 是 |  |
| 咨询日期 | date | date | 10 | 否 |  |
| 咨询时间 | time | time | 13 | 否 |  |
| 文/理科 | sciorart\_id | int | 10 | 否 | 文理科表的外键 |
| 是否报考国际学院 | is\_international | tinyint | 1 | 否 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 第一意向专业 | major\_id1 | int | 10 | 否 | 专业信息表外键 |
| 第二意向专业 | major\_id2 | int | 10 | 是 | 专业信息表外键 |
| 第三意向专业 | major\_id3 | int | 10 | 是 | 专业信息表外键 |
| 咨询省份 | province\_id | int | 10 | 否 | 省份信息表外键 |
| 咨询老师 | staff\_id | int | 10 | 否 | 用户信息表外键 |

rankpredict 排名分数关系预测表：本数据库表存储各省当年分数排名预测信息，包含以下属性：ID（主键）、score 分数、highrank 该分最高排名、lowrank 该分最低排名、province\_id（省份信息表外键） 省份、sciorart\_id（文理科信息表外键） 文理科。该表的设计见表3-5所示：

#### 表3-5 rankpredict表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性中文名称 | 属性英文名称 | 存储数据类型 | 长度 | 是否为空 | 备注 |
| 序号 | ID | int | 10 | 否 | 主键 |
| 分数 | score | smallint | 5 | 否 |  |
| 最高排名 | highrank | smallint | 5 | 是 |  |
| 最低排名 | lowrank | smallint | 5 | 是 |  |
| 省份 | province\_id | int | 10 | 否 | 省份信息表外键 |
| 文科/理科 | sciorart\_id | int | 10 | 否 | 文理科表外键 |

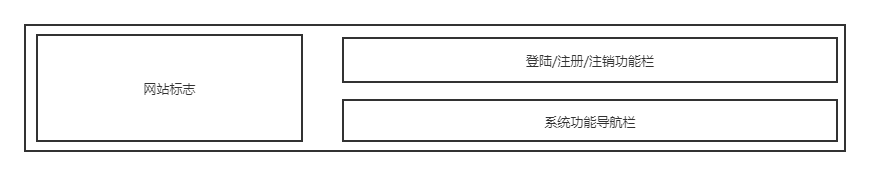
consultmaterial 招生咨询文件记录表：为了防止上传的文件命名问题导致寻找文件地址时可能出现错误，系统在上传文件后会为文件按照当前时间命名，同时通过本数据库表保存文件真实名称与存储名称之间关系，本表包括以下属性：ID（主键）、name 文件名、date 上传日期、time 上传时间、user\_id（用户信息表外键） 上传用户、file\_name 存储名、province\_name（考虑到文件可能不止作用于唯一省份，例如招生简章这种全国适用文件，这里增加了“全国”省份选项，因此该属性不是省份信息的外键） 文件作用省份。该表的设计见表3-6所示：

#### 3-6 consultmaterial表设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性中文名称 | 属性英文名称 | 存储数据类型 | 长度 | 是否为空 | 备注 |
| 序号 | ID | int | 10 | 否 | 主键 |
| 文件名 | name | varchar | 100 | 否 |  |
| 上传日期 | date | date | 10 | 否 |  |
| 上传时间 | time | time | 13 | 否 |  |
| 上传用户 | user\_id | int | 10 | 否 | 用户信息表外键 |
| 文件存储名 | file\_name | varchar | 25 | 否 |  |
| 作用省份 | province\_name | varchar | 5 | 否 |  |

## 3.4 系统界面设计

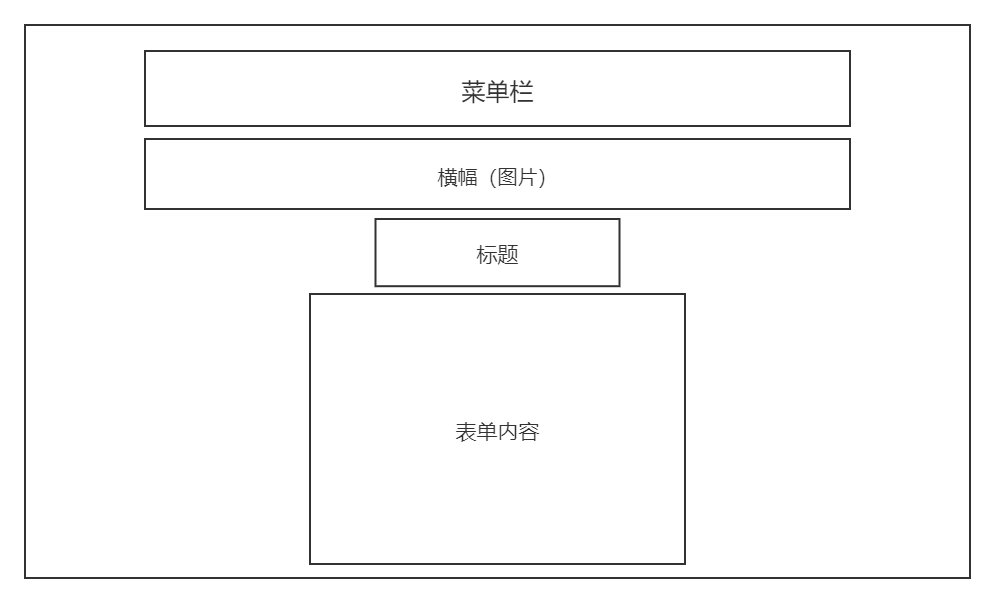
本系统主要有三个主要界面：主界面、表单展示页面、数据展示页面。

除数据展示页面外，所有页面共用一个样式的菜单栏。菜单栏的导航区域为不同权限的用户显示不同的功能。左边的标志可以方便用户随时回到首页，右上角注册/登陆/注销链接会根据用户的登录状态动态切换。菜单栏设计原型见图3-4。

#### 图3-4 菜单栏设计示意图

主界面根据登录用户权限不同显示不同的系统功能，示意图以普通用户主界面功能为例。普通用户主界面还有学校风光，老师联系方式板块，方便前来咨询的考生和家长了解学校。主界面的设计原型如图3-5所示。

#### 图3-5 主页设计示意图

表单展示页面用于所有需要数据库保存或调用数据库信息的操作，例如登陆、注册、考生信息收集等功能。表单页面包括菜单栏、标题栏和表单三部分。标题栏显示改表单的题目和作用，表单栏显示表单的具体内容，包括项目名称、填写框和填写提示。表单展示设计示意图如图3-6所示。

#### 图3-6 表单页面设计示意图

数据展示页面包括标题区域、数据表展示区域、备注区域和链接区域。标题区域展示了该数据表的标题。备注区域展示了该数据表的一些阅读注意事项，不一定所有数据表的展示都包括这块区域。链接区域包括查看省份选择、返回上一页、下载excel选项。数据展示页面设计示意图如图3-7所示。

#### 图3-7 数据表展示页面设计示意图

## 3.5 本章小结

本章在前一章需求分析的基础上，以实现需求为目的对系统的架构、功能模块、数据库和界面进行了分析。分析采用自顶向下的方法，功能模块下细分子功能模块，数据库先进行E-R图设计后确定数据表属性。为后续分析实现本系统做好了充足的准备。

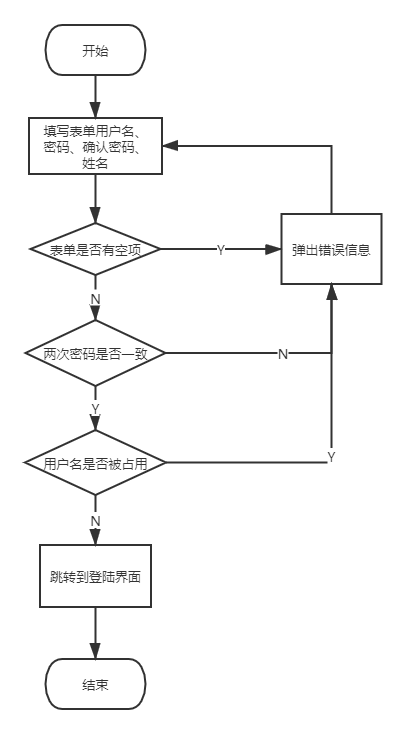
# 第四章 系统详细设计与实现

Django框架为了保持低耦合度，使用了与MVC模式大体相同MTV模式，即模型（model）、模板（Template）、视图（View）。本系统编写的业务核心逻辑代码大多在各自应用（application）文件夹下的views.py文件中，但一个view.py文件会包含多个核心业务逻辑。因此本章指出的逻辑代码位置会精确到函数名称。

## 4.1 用户管理模块的详细设计和实现

用户管理模块主要分为用户注册、用户登录、完善用户信息、修改用户信息和权限限制五个子功能模块。这五个模块前端依赖于HTML表单、服务器端使用python进行逻辑处理。其中注册登录表单和相关数据库表为Django框架自带，其余表单为自行设计。

### 4.1.1 用户注册子模块详细设计



#### 图4-1 用户注册流程图

用户注册流程图如图4-1所示。用户注册相关逻辑代码位于users文件夹中views.py中的register函数。首先判断用户是否在使用post方式提交数据，如果不是，则服务器让前端显示注册用的表单；如果是，则用临时变量form接收表单信息。之后进行表单认证，表单认证根据对应数据库要求，确认用户所填写各类信息符合存储要求，其次运行在forms.py中自定义认证方式，即认证两次输入密码是否一致。如果不一致则会抛出无效表单的错误并弹出提示。表单认证通过后，系统尝试到数据库中寻找该用户名的数据，如果找不到则为其创建条目并返回登陆界面；反之返回用户名被占用的提示。

### 4.1.2 登陆模块

本系统所使用的登陆模块逻辑是Django框架自带的登陆视图LoginView，该源代码可以在Django安装目录下contrib/auth/views.py中找到，这里简述一下大致的逻辑。首先当用户请求登陆页面是，首先用户通过get方式请求页面时，系统会首先调用dispatch()方法。之后用户通过post方式提交表单后，系统会再次调用dispatch，通过post()方法处理form数据。如果数据正确则调用form\_valid()，反之调用form\_invalid()。在form\_valid方法的最后会将用户重定向至登陆成功后的显示页面，在本系统中就是系统的主页。

### 4.1.3 完善个人信息模块

#### 图4-2 完善个人信息流程图

完善个人信息流程图如图4-2所示。当用户首次登陆系统时（即上次登录时间为空），系统会自动将用户重定向到完善个人信息页面。用户需要在页面上填写表单的对应项目，提交之后系统会检查表单是否有空项、表单各项是否符合数据库表存储格式和规定。检查无误后系统会在相应的数据库表中创建一条记录，其中定义为非空的项目将会随机生成数据，而后通过用户填写的表单修改这些数据。完成用户个人信息的完善功能。