**软件设计文档**

**Software Design Documentation**

项目名称：社团管理系统 Union CMS

组长： 罗干

组员： 周小峰 刘东华 王辛瑀 苏海涛 马超

**软件的技术选型及理由：**

1. 开发语言：Python 2.7

Python是一种面向对象的解释型计算机程序设计语言，它语法清晰简洁，而且具有丰富和强大的库。Python语言代码简单易懂，可移植性强。使用python语言进行软件的开发可以使开发速度得到较快的保证，而且我们采用的框架是Django，这是一个开放源代码的Web应用框架，由Python写成，所以采用python为开发语言。

1. 客户端：Web客户端

Web客户端是指基于Web的，现在一般是动态页面技术的客户端软件。也就是我们通常说的网页程序。Web客户端降低了应用软件部署的难度，减少了更新操作。web系统的维护相对简单，出现任何可以快速定位。

1. 数据库：SQLite

SQLite是一个轻量级、跨平台的关系型数据库。它的数据存储非常简单高效。SQLite可以用作临时数据集，以对应用程序中的一些数据进行一些处理。在SQLite数据库中，数据查询非常简单。SQLite非常容易学习和使用。它不需要任何安装和配置。只需复制计算机中的SQLite库，就可以创建数据库了。由于我们开发的社团管理系统是一个简单的小型系统，鉴于SQLite的上述特点，使用SQLite作为数据库是一个不错的选择。

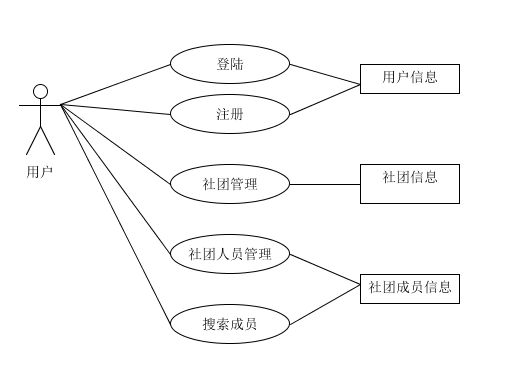
1. 基本技术框架：Django

Django是一个开放源代码的Web应用框架，由Python写成，采用了MVC的框架模式。使用Django的主要目的是能够简便、快速地开发数据库驱动的网站。它强调代码复用,多个组件可以很方便的以“插件”形式服务于整个框架，Django有许多功能强大的第三方插件，这使得Django具有很强的可扩展性。

**软件的架构设计：**

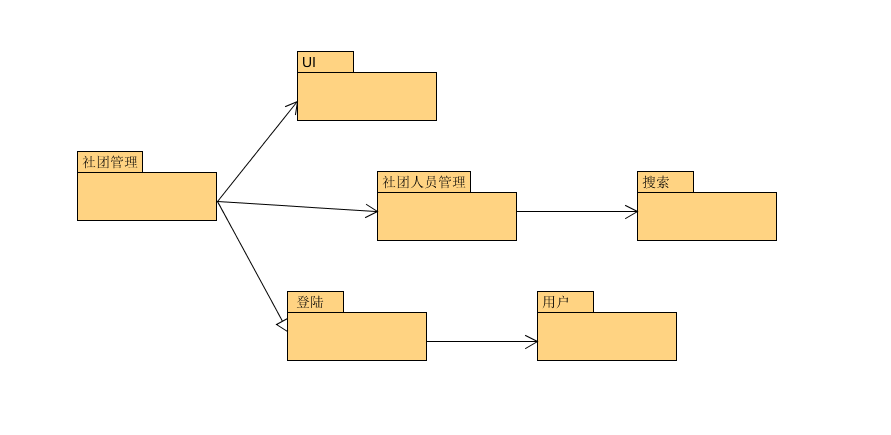
本组设计的是一个社团管理系统，主要功能有用户的登录与注册。再通过登陆后进入主页面，可以进行社团的创建与管理，在社团的内部也可以进行成员的添加，删除，编辑成员信息。还可以通过搜索进行成员的查找。因此我们采用了分层的架构，来时的网站的架构更为清晰，从而便于网站的设计、运行与维护。

之后通过用户需求分析我们进行了相应的架构设计，产生了用例图。



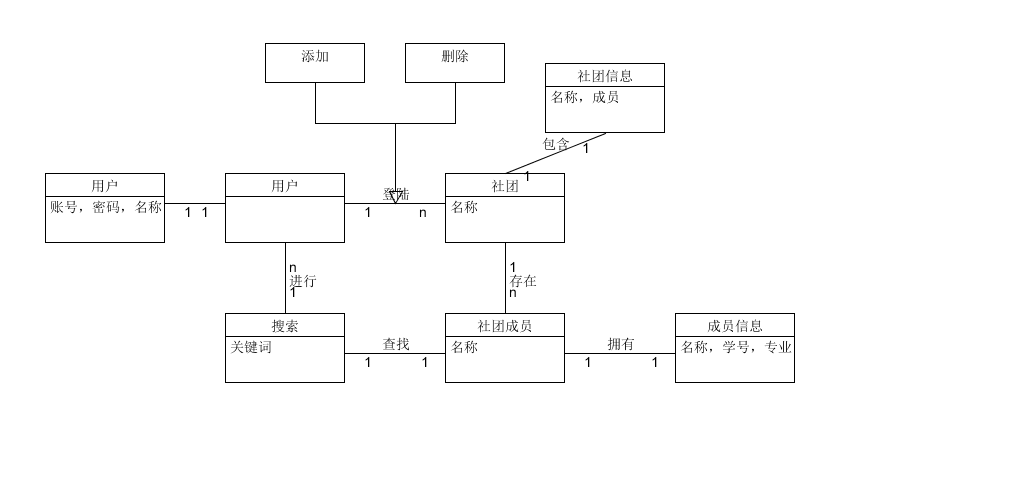
以此来表示所需要实现的

网站架构设计



从此图我们可以看出，用户通过登陆进入社团管理界面，之后再社团管理界面进行社团的编辑，之后可以在社团成员的管理界面进行社团人员的信息的编辑，还可以使用搜索功能来进行社团成员的查找。

之后是画出ER图的设计，从而进行相应设计。



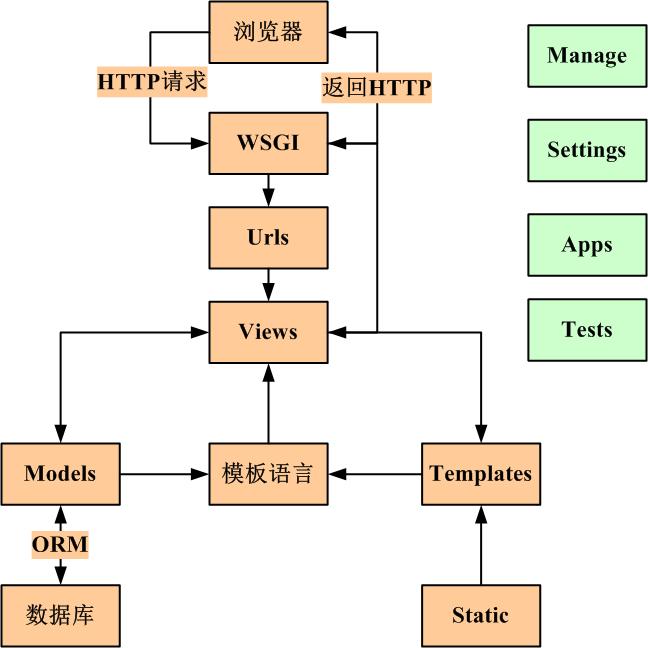
**项目的模块划分：**

|  |  |  |  |
| --- | --- | --- | --- |
| 模块 | 功能 | 负责人 | 说明 |
| Module 1 | 用户注册登陆模块 | 罗干，马超 | 管理人员注册、登陆 |
| Module 2 | 社团功能操作模块 | 周小峰，刘东华 | 已注册人员可以添加社团、删除社团、查看社团信息、编辑社团信息。（社团信息包括：成员姓名、学号、性别、专业） |
| Module 3 | 社团成员操作模块 | 周小峰，刘东华 | 包括社团成员增删操作，成员信息编辑操作。 |
| Module 4 | 关键词搜索模块 | 王辛瑀，苏海涛 | 根据成员学号、姓名、专业进行搜索（无法进行模糊搜索） |

**软件设计技术：**

**Object-Orented Programming（OOP，面向对象编程）:** 面向对象编程是种具有对象概念的程序编程范型，同时也是一种程序开发的抽象方针。它它将对象作为程序的基本单元，将程序和数据封装其中，以提高软件的重用性、灵活性和扩展性，对象里的程序可以访问及经常修改对象相关连的数据。

优势：数据和方法被封装在一起，易维护，更有利于应对可能出现的需求变更；通过继承减少代码冗余；易拓展，由于继承、封装、多态的特性，自然设计出高内聚、低耦合的系统结构，使得系统更灵活、更容易扩展，而且成本较低。



**Design Patterns: MTV:**

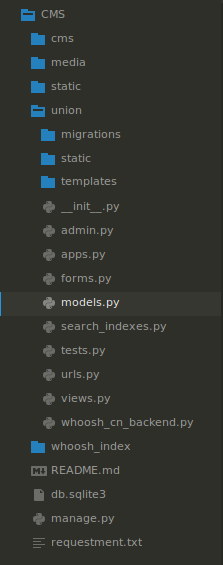
M(model)：定义数据库相关内容，放置在models.py中。

T（templates）：HTML被放置在templates中，称作模板。

V（views)：视图，定义业务逻辑相关代码。

在Django中，urls、orm、static、settings等起着重要的作用。一个典型的业务流程如右图所示：

Union-CMS项目的文件目录结构：



manage.py：Django为我们生成的管理这个项目的 Python 脚本文件。

db.sqlite3: 数据库。

Requestment.txt：包含安装所需依赖项的txt文件，可通过pip指令安装。

media目录：用户上传的图片等资源的保存位置，如社团Logo图片。

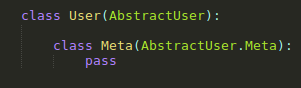
Static目录: 用于保存整个项目中的静态文件（例如html，css，js）。

migrations目录：用于存放数据迁移的文件。

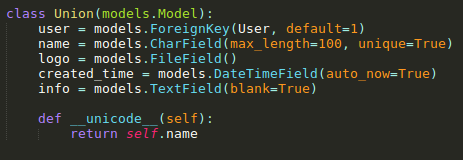
templates目录：用于存放html模板。

Union目录下重要代码简介：

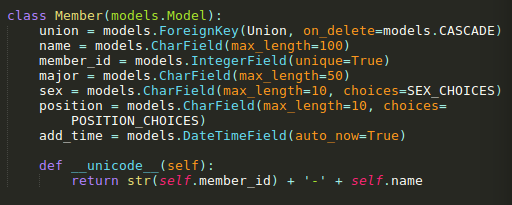
**models.py:** 主要用于定义整个应用的数据模型，相当于数据库中的sql语句：



定义社团管理员，继承AbstractUser



定义社团信息，user是这个社团的外码，采用多对一对应管理员，管理员可以管理多个社团。Name,社团名字；logo，社团logo图片;created-time社团创建时间；info社团信息简介。



定义社团成员信息，通过类比，position主要是成员的职位，主要有两种，president和member,会长和会员。

**\_\_init\_\_.py:**包含这个文件说明这个文件夹是一个应用；

**Admin.py：**用定义后台管理的文件。

**Forms.py：**这个主要用来定义表单的，应用中涉及到的表单都由这个文件定义。**Search\_indexes.py:**搜索功能。

**Tests.py：**用于写测试代码的文件。

**Urls.py：**定义的是URL映射，前段url通过通过这个文件映射的views.py中对应的函数，进行业务逻辑处理。

**Views.py：**进行业务逻辑处理的文件。

**Whoosh\_cn\_backend.py：**搜索功能涉及到的文件。