[[1]](#footnote-1)[[2]](#footnote-2)

**本科毕业设计（论文）**

**( 2017届 )**

**题 目： 学术会议管理系统**

**分 院：** 工学分院

**专 业：** 计算机科学与技术

**学生姓名：** 王开 **学号：** 13196133

**指导教师：** 苗春雨 **职称：** 副教授

**合作导师：** **职称：**

**完成时间：** **2017 年 4 月 22 日**

**成 绩：**

**浙江师范大学行知学院本科毕业设计（论文）目录**

一、诚信承诺书

二、正文

三、浙江师范大学本科毕业设计（论文）任务书

四、浙江师范大学本科毕业设计（论文）开题报告

五、浙江师范大学本科毕业设计（论文）指导记录

六、浙江师范大学本科毕业设计（论文）中期检查表

七、浙江师范大学本科毕业设计（论文）结题答辩资格审查表

八、浙江师范大学本科毕业设计（论文）结题答辩记录

九、浙江师范大学本科毕业设计（论文）评审表

**浙江师范大学行知学院本科毕业设计（论文）**

**诚信承诺书**

**本人郑重承诺**：

我承诺所呈交的毕业设计（论文）是本人在指导教师的指导下，按照学院和分院的有关规定，独立研究完成的。本人在毕业设计（论文）写作过程中恪守学术道德和学术规范，设计（论文）中凡引用他人已经发表或未发表的成果、数据、观点等，均已注明并列出了有关文献的名称、作者、年份、刊物名称和出版文献的出版机构、出版地和版次等内容，除此之外均为本人的观点和研究成果。

如有违反，本人愿接受处罚并承担一切责任。

承诺人签名（手写）：

年 月 日

[**浙江师范大学行知学院本科毕业设计(论文)正文**](#_第_一_部)

**目 录**

摘要………………………………………………………………………………………1

关键词……………………………………………………………………………………1

英文摘要…………………………………………………………………………………1

英文关键词………………………………………………………………………………1

前言………………………………………………………………………………………1

一 背景介绍……………………………………………………………………………1

二 研究内容 …………………………………………………………………………2

1 论文管理的设计 ………………………………………………………………2

2 传统框架的局限性 ……………………………………………………………2

3 系统开发的敏捷性 ……………………………………………………………2

三 平台关键技术介绍

1. Python ……………………………………………………………………3
2. Django ……………………………………………………………………4
3. Wsgi ……………………………………………………………………4
4. Message queue……………………………………………………………4

四 系统需求分析 ……………………………………………………………………4

1. 系统总体目标 ……………………………………………………………5
2. 可行性分析 ………………………………………………………………5
3. 系统功能需求 ……………………………………………………………8

五 系统总体设计 ……………………………………………………………………8

六 系统详细设计与实现 ……………………………………………………………12

七 系统测试 …………………………………………………………………………14

八 总结与展望 ………………………………………………………………………15

参考文献 ………………………………………………………………………………16

致谢 ……………………………………………………………………………………17

**学术会议管理系统的设计与实现**

工学分院 计算机科学与技术专业 *学生姓名王开（13196133）*

指导老师： 苗春雨（副教授）

**摘要**：随着经济发展学术水平的提高，附在会议上的需求越来越多样化，细节化，学术会议作为一个分支，也从多方面迫切的需要一个更符合目前需求的管理系统。研究的目的是实现一个对于组织者和参会者来说一个好用可靠的学术会议管理系统。研究方法主要是对国内外已经有的管理系统进行调研和分析，保留了需要有了基础功能。根据持续的调研使用者的需求总结出了网上注册，论文上传，论文评审等功能。再理清各个功能的流程后，使用 Django 框架进行敏捷开发，快速迭代实现。

**关键词**：学术会议；敏捷开发；Django框架

**Design and Implementation of Academic Conference Management**

Wang Kai Director：Miao ChunYu

(College of Engineering, Computer Science and Technology, Zhejiang Normal University )

**Abstract**：With the improvement of the academic level of economic development, the demand at the meeting is becoming more and more diversified and detailed, and the academic conference as a branch also needs a management system which is more in line with the current demand. To achieve a useful for the organizers and participants for a reliable academic conference management system. The research method is mainly to carry on the research and the analysis to the domestic and foreign already has the management system, has retained the need to have the basic function according to the continuous research user's needs summed up the online registration, the paper uploads, the paper appraisal and so on the function. And then sort out the various functions of the process, the use of Django framework for agile development, rapid iterative implementation.

**Key Words**： academic conference；agile software development；django framework

**前言**

**一、背景介绍**

随着国内经济发展和学术水平的提高，中大型的各类学术会议举办的数量日益增多。随之而来的是会议的管理需求越来越多样化，细节化。参会者希望在会前会后得到准确及时和会议信息，举办者希望可以减少琐碎的需要人工一一确认的事务，比如会议场地的安排，邀请嘉宾名单，参会者名单，论文上传，论文评审，参会者住行安排等，十分复杂。在以往出现的会议系统中，还存在许多问题和不足：

1. 会议举办方需要通过各种通讯方式联系参会者确定是否参加和通知参加的注意事项，还需要记录各个参会者提出的要求，花费大量的时间。
2. 论文管理也是传统管理的一个大难题。论文无论是通过电子邮件还是通过普通邮寄的方式来收取，会议举办方都需要通过手工处理，需要整理拆封寄来的论文才能获得论文的相关信息，然后再评审，针对论文的作者的水平层次来进行整理分类。最后再对录用的论文进行编写。这往往会占用会议组织者很多时间，并且有可能会出现论文丢失，评审失误等错误。
3. 传统的会议系统还在使用 C/S 架构，即还需要下载实体的软件来运行程序，如果有10台计算机则需要安装10次该软件，每台电脑的系统环境等不同，安装的版本可能也不同。这样会耗费会议组织者大量的时间，有些组织这不得不考虑时间成本后放弃这种管理方式。旧的会议管理系统往往设计死板不可扩展和自定义，不是设计繁琐难以使用就是设计过于简单导致很多功能都缺失。有些则需要付费后才可以订制开发来满足不同学术会议的需要。
4. 现有的一些 B/S 架构的会议管理系统依旧在使用静态网页作为会议通知的平台。如果需要修改会议的开会时间等信息，就需要再次修改网页，发布。这就会导致不能及时的通知到参会者，往往需要结合电话和电子邮箱来通知参会者，论文管理等复杂功能也往往不能实现。

基于以上的问题，开发和采用一个敏捷、可复用、高效率的学术会议管理系统已经迫在眉睫。会议管理系统可以帮助会议组织这实现在线的参会邀请、通知，论文的收集、评审等过程。

**二、研究内容**

传统的学术会议开发系统中，存在着很多的不足，比如臃肿的开发框架，不敏捷的开发模式，非常慢的响应系统，归纳起来。需要改进的也是作为这次我的主要的研究内容的可以归纳成以下几点：

1. **论文管理的设计**

本文把论文管理分为论文提交，论文审核和论文评审。传统的论文投稿方式一般都是通过电子邮件和邮寄，而通过论文管理系统，参会者只需要打开网页，登录自己的账户后上传论文的电子版即可。参会者上传论文以后，组织者也不需要再像传统的方式一样去手工的分类整理论文，而是可以直接登录网站对论文进行初步的审核，审核通过以后开始分发给各位专家进行评审。评审一般是评审，论文的评审是由多个审稿员打出的，由多个审稿人评阅后，打出相应的等级再排名来确定是否邀请该人参会。评审过程中，论文作者的姓名，审稿人的姓名都是匿名的，以此保证评审结果的公正。

1. **传统框架的局限性**

目前主流的开发框架还是以 J2EE 类型的居多，一般都是使用 SpringMVC和 mybatis作为开发框架。使用这种传统的框架来开发这种需要频繁修改功能的时候，就会出现很多问题。比如最基础的需要修改一个字段，就需要修改DAO等一整套东西，十分的繁琐复杂， 系统的冗余度非常高，同样的内容会出现再很多地方，使得开发效率大大的降低。传统的 SpringMVC等框架可以在大型复杂的系统中大展生手，它拥有完备的事务管理，依赖注入控制翻转等功能十分完善，但是在需要面对灵活多变的情况的时候，有有点杀鸡用牛刀的感觉，往往使不上劲。而学术会议管理系统的需求就是比较灵活多变的那一类，如果使用传统框架开发就明显不太合适。

1. **系统开发的敏捷性**

**相对于产同软件的开发方式，现在敏捷开发越来越受互联网公司的欢迎。这是一种崭新的软件工程的软件开发流程，拥有快速应对需求变化的软件开发能力。它比较注重开发团队与业务专家团队彼此的紧密协作和面对面的沟通，需要平凡的开会交流并发布软件版本，更加注重软件开发中人的作用。传统的开发方式一般是瀑布模型式，需要严格遵循事先安排好的需求、分析、设计、编码、测试的步骤顺序进行。对比传统的瀑布流开发方法，敏捷方法则在几周或者几个月的时间内完成相对较小的模块，可以尽早开始测试和沟通需求，强调的是能将尽早将尽量小的可用的功能交付使用，并在整个项目周期中持续改善和增强。**

**平台关键技术介绍**

1. **Python**

python 是面向对象的脚本语言，是吉多·范罗苏姆再1989年的圣诞节期间\_开发。python 是完全面向对象的语言，“一切都是对象”，数字，字符串，模块，字符串都是对象，完全支持继承、派生、重载、多重继承。它语法简单，并且使用缩进来写代码块。Python 的设计哲学是一种方式完成一件事，“明确”，“优雅”，“简单”。python 广泛的应用于 Web 程序，GUI 开发，操作系统等。Django、Tornado、Flask 等 Web 框架的出现，大大的提高了程序员的开发效率来管理复杂的 Web 程序。

1. **Django**

Django 是一个开放源代码的 Web 应用框架，由 Python 写成。采用了 MVC 的软件设计模式，即模型 M，视图 V 和控制器 C。它最初是被开发来用于管理劳伦斯出版集团旗下的一些以新闻内容为主的网站的。并于2005年7月在 BSD 许可证下发布。这套框架是以比利时的吉普赛爵士吉他手 Django Reinhardt 来命名的。

1. **Wsgi**

wsgi(Web Server Gateway Interface）项目致力于为构建一个全栈式的托管服务。在WSGI规范下，web组件被分成三类：client, server和 middleware。WSGI apps(服从该规范的应用)能够被连接起来(be stacked)处理一个request，这也就引发了中间件这个概念，中间件同时实现c端和s端的接口，c看它是上游s，s看它是下游的c。WSGI的s端所做的工作仅仅是接收请求，传给application（做处理），然后将结果response给middleware或client.除此以外的工作都交给中间件或者application来做。多功能，高性能，占用资源少和可靠性是这个项目的优势（也是唯一遵循的规则）。

1. **Message queue**

消息队列是进程间通讯或者是统一进程不同线程间的通讯方式。消息队列是一种异步的通讯方式，也就是说输入信息的人和取数据的人不需要在同一时间操作，消息会保存再队列中，直到接受者取回它。消息队列的实现是保存信息在链表结构中，一个进程如果拥有权限的话可以向消息队列中写入或者取出消息。但是消息队列也有缺点，因为它本身是异步的，而 HTTP 协议一般都是同步的，再发出请求后必须等待服务器的回应，所以接受者往往就需要轮询，才可以确保快速的接受到最近的消息。

1. **本章小结**

本章对学术会议开发时用到的技术进行了总结。该系统开发是需要考虑到敏捷开发和扩展性的需求，故系统选择了 python 语言的 django 框架进行开发，也借此引出了 wsgi 和 message queue 等技术支撑，使得在开发技术层面上有足够的技术储备。

**系统需求分析**

1. **系统总体目标**

本次重新设计的学术会议管理系统应该有敏捷、可复用、高效率的特点，同时具有信息录入发布，开放，统一的服务平台。会议组织者可以方便的给参会者发布信息，可以快捷的处理论文。

根据以上描述，本系统大致分为论文管理和会议管理。会议管理中，需要实现会议信息的及时发布和通知。在论文管理中，大致分为系统管理员，论文提交者，评审专家三个角色，每个角色再页面中的操作权限是不同的，但是流程不收影响。系统管理员确定论文的主题，上传的截至时间和论文写作要求，并再论文提交后进行初步审核。论文提交者根据页面的提示来上传写完的论文，查看论文的审核和评审情况。审稿专家再不知道的作者的前提下进行论文的打分，并可以收到论文更新的提醒。

1. **可行性分析**

项目的可行性一般都是从时间、经济和技术三个方面进行分析：

1. **时间可行性**

项目的开发到项目的正式上线的使用，用了四个月左右的时间。如果会议有特殊的需求，根据新的需求进行产品的迭代也不会占用很长的开发时间可以迅速的满足新的会议管理的需求。

1. **经济可行性**

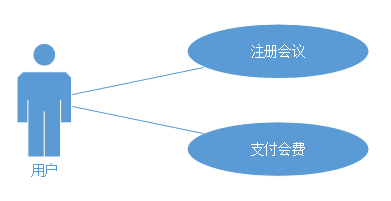
如果一个项目可以带来的收益大于付出的成本，那么它就是经济可行的。传统的会议管理常常需要大量的成本比如沟通成本，时间成本等。会议管理系统可以大大减少参会者和会议举办者的沟通成本，不需要电话邮件等传统形式的通知，论文管理可以加快审核评审的速度，大大提升了效率，减少了时间成本。所以不管从人力的成本还是从时间成本的角度计算，经济上都是可行的。

1. **技术可行性**

项目需要用到成熟的敏捷开发框架，而 Django 是目前业界公认的实现敏捷开发的最好用的框架之一。并且 Django 是基金会维护的开源框架之一，使用 Python语言，django 注重方法的重用和可插拔性，敏捷开发和 Don't Repeat Yourself 法则。所以用来开发灵活多变的会议管理系统是完全没有问题的技术上也是可行的。

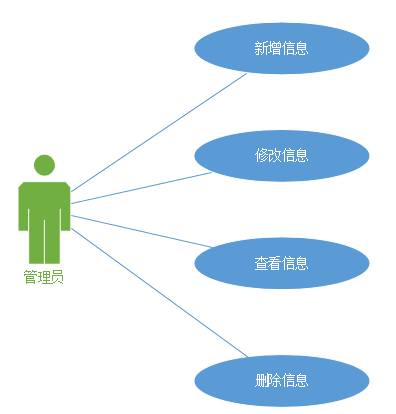
1. **系统功能需求**
2. **会议管理模块**
3. **用户注册**

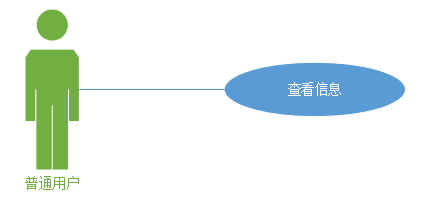
​ 参会者、评审者都需要进行注册以后才能登录系统，从而看到更多有关自己参加的会议的信息和论文审核的信息，该模块的用例图如下所示：



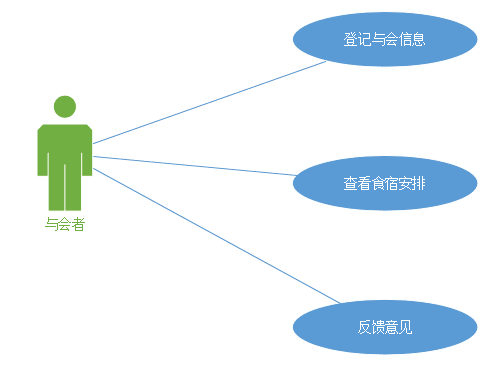
1. **会议信息发布**

该模块的使用者一般是系统的管理员，只有管理员才能发布或者修改删除信息，普通的参会者等用户只能登录后查看会议信息。会议信息一般可以自定义，包含会议地点，会议时间，论文要求等信息。信息发布模块的用例图如下所示：



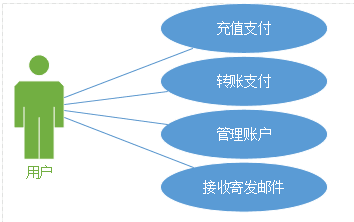


1. **食宿安排**

学术会议往往时间长，需要吃饭与住宿的地方，所以这是每一次线下会议避不开的功能，在传统的会议管理系统中都是到了现场才开始安排食宿，这样沟通起来会出现非常多的问题。在现在的系统中，用户可以提前登记往返时间和需要住宿的信息。会议举办者就可以很方便的根据会议参加者的统计信息来安排饭店和旅馆，参会者如果有特殊的要求可以发送给会议安排者。食宿安排的用例图如下所示：

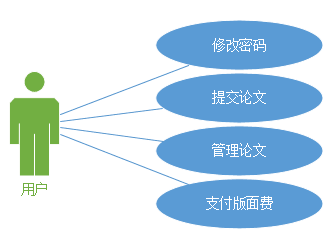
1. **在线支付**

在学术会议管理系统中，缴纳参会费，来往的交通和食宿费用加上论文的审核费用和版面的费用都需要通过网页在线支付。在此系统中一般使用支付宝和微信支付来进行，自动生成订单后付款即可。在付费成功以后可以通过快递或者电子发票来寄送发票。在线支付的用例图如下所示：

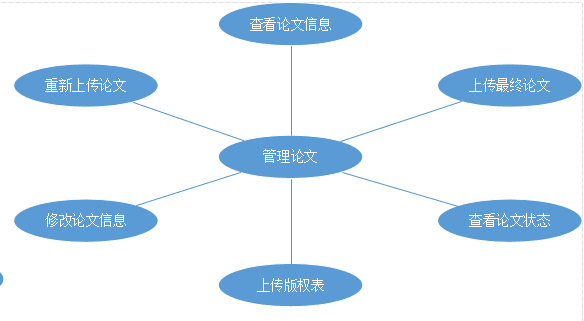


1. **论文管理模块**
2. **参会人模块**

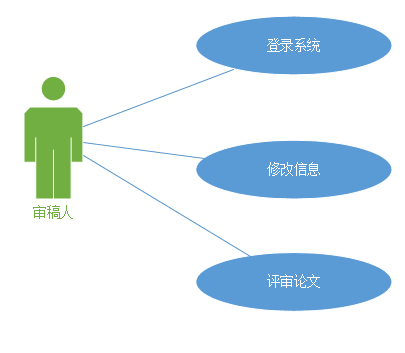
​ 参会者还未入选会议前一般只能访问论文管理系统，需要有注册功能，登录前如果忘记密码可以通过手机号码或者电子邮箱号码找回。登录系统后的主要功能需求还是上传论文，包括完整的论文信息和论文的 PDF 或者 Word 文件，如果没有审核通过可以继续修改论文信息和文件然后继续上传重新评审。参会人还可以在论文管理里查看自己的论文评审和审核状态，如果论文通过了评审，还需要支付论文的版面费用。参会人的用例图如图所示：



管理论文的功能也需要详细说明，在调查需求的时候发现，审核人员一般会对论文有一些指导批注，参会者需要根据批注等对论文进行修改并再次上传。论文的状态也包含审核通过，外审，修改，终审等状态。论文管理的用例图如下所示：

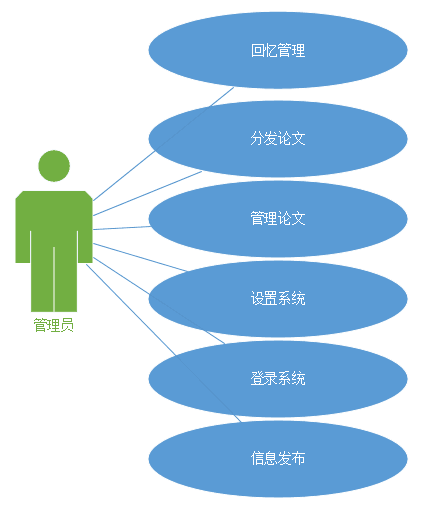


1. **审稿员模块**

 审稿员主要是登录学术管理系统后，对管理员分发的参会员上传的论文进行评审。审稿员需要再系统制定的时间范围内对分配的论文进行评审打分并给出修改意见，填入学术会议管理系统后上传。审稿员的用例图如下所示：

1. **管理员模块**

管理员的主要需求有两个：流程安排和上传论文的整理分配给审稿员，用例图如下所示：



流程安排主要包含一下需求：开启/关闭注册，论文上传的截止时间设置，论文提交要求编写发布，设置本次学术会议的主题，修改密码等功能。上传论文的管理中，管理员需要对上传的论文进行审核，查看论文信息是否填写正确，是否有差错，上传的 PDF 是否正确完整等，并且再审核之后根据主题不同分配给不同的专家进行评审。管理员也负责审稿人帐号的维护，需要填写审稿人负责的主题，并提供列表查看的功能。

1. **系统非功能性需求**

​ 学术会议管理系统的非功能性需求包含系统的可靠性，可拓展性（快速开发性），易用性：

1. **可靠性**

​ 可靠性一般以一年中宕机的时间百分比来计算，一般以几个9来定义，比如4个9代表99.99%的可靠性，及一年24\*365小时可以宕机0.8小时，学术会议系统对可靠性的要求一般再3个9即可，选择阿里云等主流的云服务器提供商一般都可以满足需求，这类云服务提供商一般也有健全的防攻击系统，可以抵御一般的 DDOS 等网络攻击。

1. **可拓展性**

学术会议的需求十分多变而且难以抽象出一个合适的模型出来，所以系统的可扩展性十分重要，一般可以先开发一个高内聚低耦合的模型，然后根据每个系统的不同需求进行二次开发来满足客户的需求。系统尽量做到不写死，系统的环境变量尽量做到都可以修改，以方便后期开发和维护。

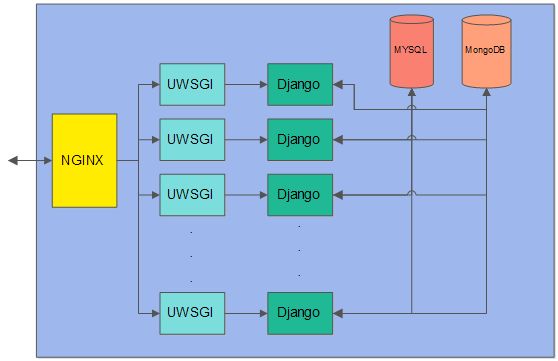
1. **易用性**

学术会议管理系统的用户一般都是从传统管理转型用户，是否能吸引用户用户的体验占了很大比例，所以需要有良好的交互来简化操作，方便用户快速的找到想用的功能，而不需要通过培训等方式去教会用户使用系统。

1. **本章小结**

​ 本章一开始对学术会议管理系统的总体需求进行了分析，然后从经济，技术，时间三方面验证了实现的可行性。接着对会议管理和论文管理的具体需求进行了需求分析，找出了最关键最需要解决的需求痛点，最后对系统的非功能性需求完成了需求分析。

**系统总体设计**



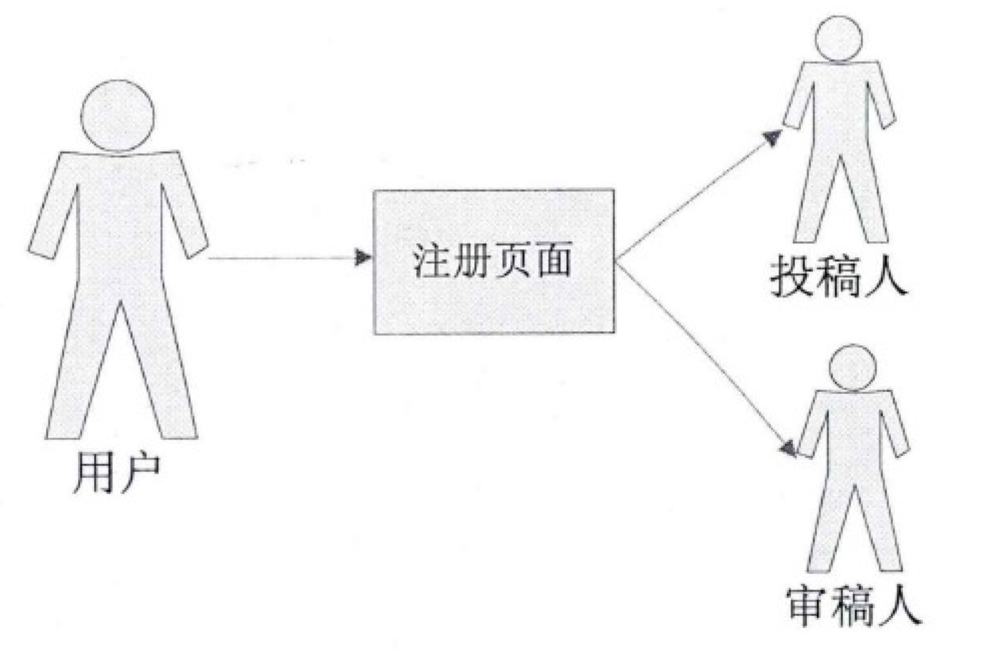
1. **网络架构设计**

​ 学术会议管理系统采用 nginx<-uwsgi<-Django app 的网络结构，通过 uwsgi 的 web 服务器发出socket ，nginx读取的方式来实现。uwsgi 相对于传统的 Gunicorn，拥有更高的性能，和 nginx 的结合度也更好。无论从服务器拓展性和性能方面，可以说uwsgi+nginx 是目前 django 框架部署的最佳实现。

1. **会议管理模块设计**

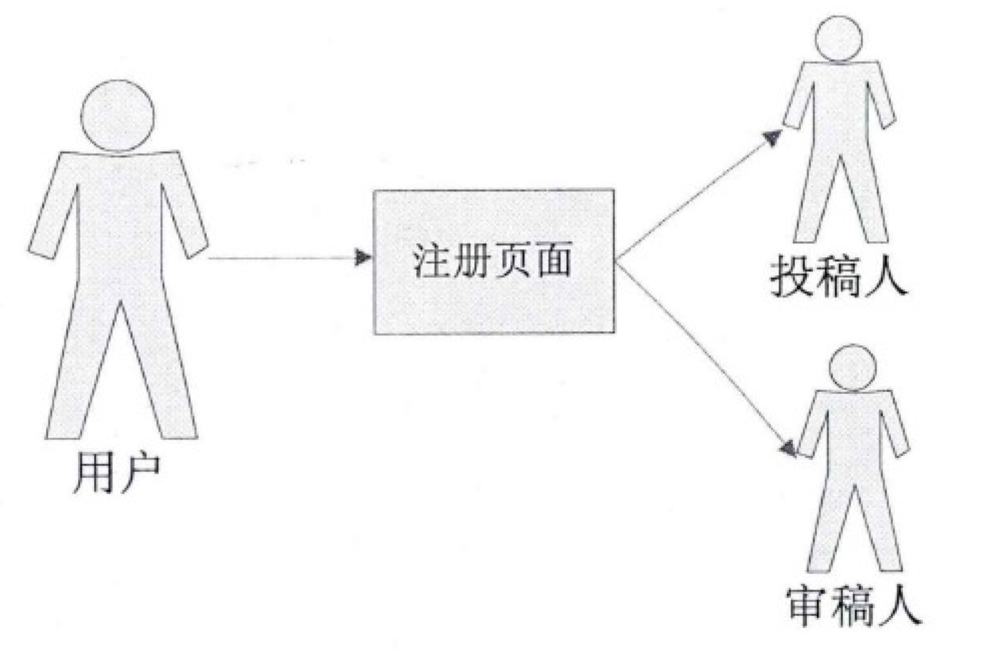
会议管理模块的主要功能是安排会议的准备工作和会议进行时的管理，更具流程的进行可以大致的分为以下这几个功能：用户注册，会议信息发布，会议参会者的食宿安排，各种功能的在线支付。以下是各种子功能的介绍：

1. **用户注册**

每个用户在使用系统前必须注册，注册的时候可以选择自己的角色，分为参会者和审稿员，注册完成后，进行登录，不同的角色进入不同的界面。用户注册的流程图如下所示：

1. **会议信息发布模块**

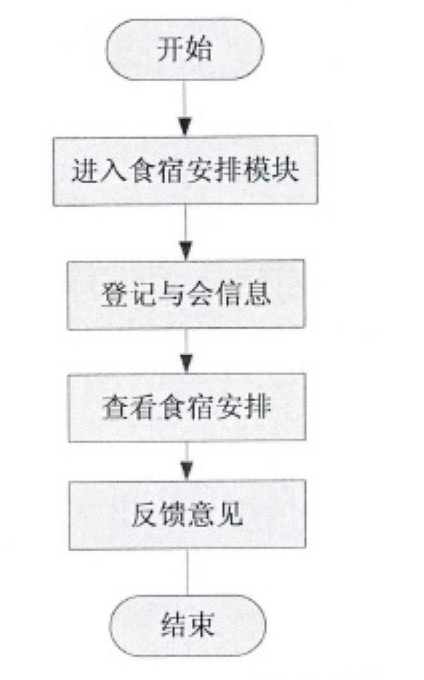
这个模块主要是帮助管理员发布会议信息，包括会议地点，会议主题，注册截至时间等等详细信息。具体的流程图如下：



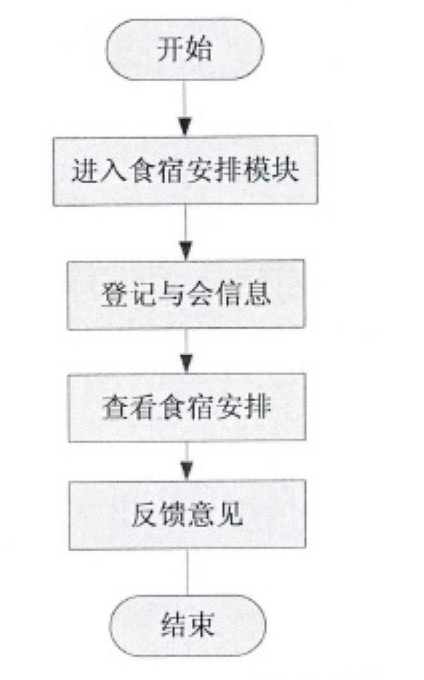
管理员登录系统以后，点击进入会议信息发布，系统中包含了一个会议的增加，删除，修改，过滤查询等功能。管理员输入各个字段以后，就可以将数据保存进入数据库，并返回列表可以点击发布。重新点击列表里已经存在的会议，就可以对会议里面的信息进行修改，保存之后还是会将数据存入数据库。保存和修改失败时，都会提醒相应的信息。点击删除后，先做出一个提示“是否删除该信息”，确认后就会删除，删除成功后跳转到会议列表，删除失败返回提示信息页面。

1. **食宿安排模块**

作为学术会议正常举行的必备条件，吃饭和住宿必须安排妥贴。这块内容也是参会者和会议安排者也就是系统管理员的角色都可以访问的但是功能不同。以下是参会者的流程图：



在最开始的时候，参会者先要填写自己的签到时间，离开时间，身份证号，并在学术会议管理系统内提交。填写完之后就等待系统管理员的通知，通知里会有详细的吃饭和住宿的地址，房号，何时签到入住合适最晚退房等信息。并且可以反馈意见给系统管理员进行调整。系统管理员的安排住宿吃饭的流程图如下：



系统管理员可以接受到参会者提交的信息，并根据信息来进行吃饭和住宿的安排。把安排发布给对应的参会者之前可以规定一个截至日期，在此之前可以提交反馈意见对食宿安排进行调整。

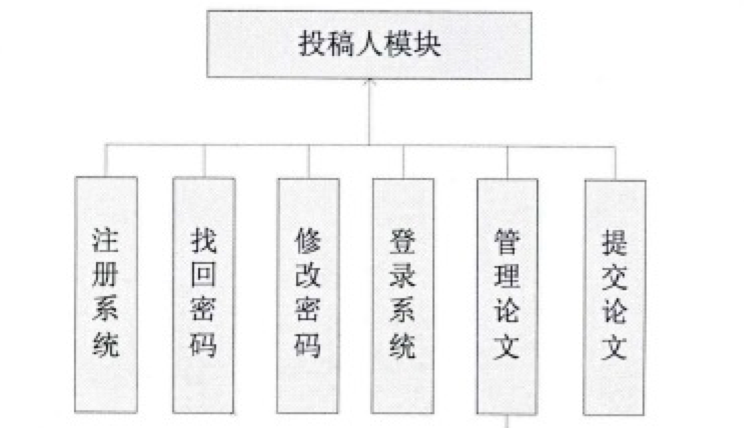
1. **论文管理模块合计**

​ 本模块根据使用人的角色不同，分为参会人（投稿人），审稿员，管理员三个功能：

1. **参会人模块**

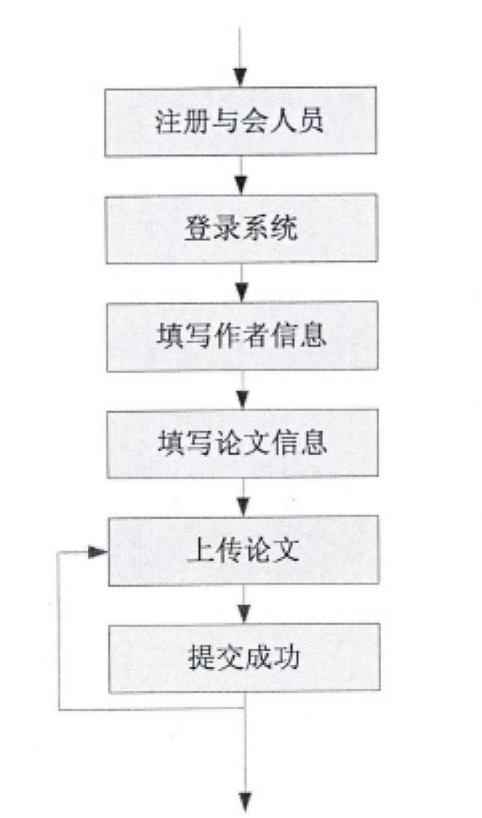
参会人登录系统后选择论文管理，模块图如图所示：

如上图所示，参会者在进入学术会议管理系统后可以修改个人的信息（修改名称，密码等），管理文论（查看论文状态，删除上传的论文等），上传论文。作为参会人，上传投稿的论文是最最重要的功能。上传论文不光是上传 PDF 或者 word文件，还需要要填写相关的论文信息，比如作者信息，论文的题目，论文摘要等信息，然后再上传作品的正文，再进行提交。 提交后，可以再论文管理中查看论文的审核状态，如果没有通过，可以重新打开上传页面对论文信息进行修改。流程图如下：



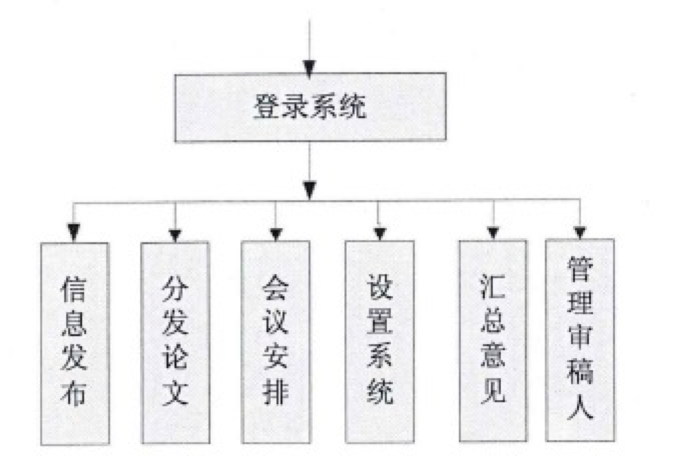
1. **审稿员模块**

审核员主要负责对管理员初步审核后的论文进行评审，登录后可以看到需要自己评审的论文，如果遇到论文不是自己负责的方向或者内容有错误可以拒绝评审，如果正确那就可以进行审稿并打出分数作为是否进入会议的标准，还需要写评语对打出的分数进行阐述以作为依据并提交到系统之中。审稿员具体的工作流程如下图所示：



1. **管理员模块**

管理员需要负责大部分的会议内容，需要对会议举办的信息进行通知，还需要对论文进行相关的管理。首先管理员需要发布会议的相关信息，比如会议的举办时间地点，主题等等。需要对上传的论文进行审核，查看各个信息是否填写的正确和完整，上传的正文文件是否对。审核完论文以后需要根据主题转发给相对应的审稿人对论文进行评审。确定参会者名单以后，管理员还需对提交上来的食宿信息进行安排，还有在线支付的发票寄送等等。以下是管理的流程图：



1. **数据库设计**

学术管理系统采用开源高性能的 mysql 关系型数据库。为了方便将设计的字段概念转换为现实世界中的模型，本文采用 E-R 模型（实体-联系模型）来进行描述。

1. **表结构设计**

学术会议管理系统共有17张表结构，篇幅原因仅列出几张例子：

个人信息表：

名称 字段 类型 备注

用户 id id Int 主键

姓名 user\_name varchar(20)

性别 sex varchar(10)

工作单位 company varchar(50)

职称 position varchar(20)

职务 duty varchar(15)

手机号码 mobile varchar(15)

密码 password varchar(20)

邮箱 mail varchar(50)

住宿宾馆 hotel varchar(20)

房间号 room varchar(20)

入住时间 in\_date datetime

退房时间 out\_date datetime

住宿费 hotel\_fee varchar(10)

证件号 idcard\_num varchar(20)

证件类型 idcard\_type varchar(20)

创建人 created\_by int

创建时间 created\_time ·datetime

会议信息表:

名称 字段 类型 备注

会议 id id Int 主键

会议名称 meetname varchar(20)

开始时间 start\_time datetime

结束时间 end\_time datetime

会议地址 address varchar(100)

会议日程 meet\_plan varchar(200)

创建时间 created\_time datetime

创建人 created\_by int

1. **本章小结**

本章一开始先对整体的网络架构进行了介绍，然后通过流程图加上语言解释的形式对会议信息和论文管理进行了如何设计流程的阐述，然后对数据库的设计通过展现表的形式和系统用例图的形式进行了说明。

**系统详细设计与实现**

1. **学会会议管理开发软件列表**

操作系统：centos7

前置负载均衡：nginx

web 服务器：uwsgi

数据库服务：mysql

Python 版本：3.5.0

开发工具：Pycharm

1. **会议管理模块**

角色分为普通用户和管理员，两个登录后的界面有所不同：

1. **会议信息**

普通用户登录系统后可以查看系统管理员发布的最新的关于会议的通知，列表包含通知的标题和通知的时间，如图所示：

首页展示前20条会议通知，可以点击更多来进入会议信息列表页面，提供翻页功能，点击标题可以查看详情。

1. **参会申请**

如果用户想参加通知的会议，可以进入参会申请页面，如图所示：

用户填写完必须的信息之后，点击提交就录入到系统之中。

1. **后台管理**

后台管理是系统管理员才能看到的模块，包含会议信息的发布，并且对参会者的住宿和吃饭进行安排，发布会议信息的界面如图所示：

会议管理者填完所需的信息之后就可以进行发布。

注册用户的参会申请后，会议管理者可以对用户进行住宿进行安排等。

1. **论文管理模块**

论文管理也如会议信息模块，需要分为参会者，审稿员和系统管理员。

1. **参会者**

参会者登录的界面如图所示：

界面的左边是菜单栏，可以帮助用户方便的看到导航信息，登录进入系统之后就可以使用各种功能。首先可以使用投稿功能，界面如图所示：



论文需要的一些信息比如作者名字，会自动填写，还需要填写其他作者，摘要，论文题目等信息，这些都是必须填的内容，如果没有填写点击提交会提示错误信息，同样，本系统中的表单提交前都需要进行验证，比如手机号码的格式，特殊字符的过滤等等。代码如下：

phone=forms.CharField(required=True,validators=[mobile\_validate,],error\_messages={'required':'请填写手机号'} )

def mobile\_validate(value):

mobile\_re.compile(r'^(13[0-9]|15[012356789]|17[678]|18[0-9]|14[57])[0-9]{8}$')

if not mobile\_re.match(value):

raise ValidationError('手机号码格式错误')

提交论文以后就可以在论文管理中看到自己上传的论文。接下来管理员会对论文进行评审，查看论文信息是否有填写错误，论文文件是否完整 ，如果审核通过，则会现实已通过审核，论文会分发给审稿员。

1. **审稿员**

审稿员登录学术会议管理系统后，可以看到给自己分配的需要打分的文章的列表，点击论文题目可以查看论文的详细内容，然后通过 pdf 的方式查看。

1. **系统管理员**

​ 系统管理元登录学术管理系统后，如果发现有新的没有审核过的论文，需要对论文进行初步审核，查看论文信息等是否有填错，如果没有问题，则转发给对应主题的审稿员。界面如图所示：

1. **本章小结**

本找详细说明了学术会议管理系统的实现，对系统的登录进行了介绍，对会议信息管理通过参会者和系统管理者两个角色分别进行了阐述，详细介绍了信息发布、参会注册等子模块。论文管理也分为参会者，审稿员和系统管理员，介绍了各自的实现界面。

**系统测试**

1. **系统测试环境**
2. **硬件环境**

处理器：Inter i7 4770HQ

内存：4\*4GB ddr3 1600mhz

硬盘：512GB SSD

网络：100M

1. **软件环境**

操作系统：centos 7

web服务器：uwsgi

数据库：mysql

测试工具：apache ab

测试参数：2000个并发 5000~10000个请求

1. **系统功能测试**
2. **测试用例**

操作名称 预期结果 实际结果

注册各个角色 可以注册 注册成功

登录各个角色 可以登录 登录成功

发布会议详细内容 可以发布 发布会议详细成功

与会注册 可以注册 注册成功

在线支付 可以支付 在线支付成功

查看参会者信息 能够查看 查看成功

删除参会者信息 可以删除 删除参会者信息成功

填写论文信息，上传稿件 可以填写论文信息 填写，上传成功

提交审稿意见 可以提交审稿意见 提交成功

汇总审稿意见 可以汇总 汇总成功

1. **系统性能测试**

系统性能测试是对网站性能的评价，总体来说性能测试主要衡量的是网站有及时响应度的能力。所以，性能测试是采用测试手段对软件的响应进行评价的一种方式，软件类型不同，侧重点也不同。使用 httperf 评估性能

现在市面上有许多的网络性能测试工具，主要包括Apache 的 ab、Flood和 httperf。由于更熟悉 httperf，故本次系统性能测试使用 httperf。

1. **测试结果**
2. **系统功能**

​ 选取测试的功能基本来自于上面的需求分析后的设计的功能，均可以操作，已经可以达到初步使用的目的。

1. **系统性能**

​ 在2000个并发 5000~10000个请求的压力下，平均响应时间可以控制在300ms 以内，CPU 占用率一直保持在80%以下，保证系统的运行的稳定性。

1. **本章小结**

本章对学术管理系统进行了全方面的测试，首先介绍了一下测试的硬件环境和软件环境，然后设计了测试用例对系统的功能进行了测试。最后测试了多用户并发方式进行性能的压力测试，跑出来的效果达到了要求。

**总结与展望**

​ 本系统是基于 Django 框架开发的，开发模式采用了时下最常用的敏捷开发模型，把学校中学习的软件工程的知识与实际的工程开发相结合，从分析需求开始先画出用例图，与一些必须要有的需求痛点。然后对这些已知的需求进行了详细的设计，而不是盲目的先开始开发，画出了详细的流程图，然后在具体实现的开发过程中，就如鱼得水，配合简单易用的 Django框架，迅速完成了简单的原型开发。本次开发的系统主要还是分为论文信息管理和会议信息管理，达到了敏捷开发，操作简单，安全高效的信息化管理目标。主要特点如下：

1. **总结**
2. **采用敏捷开发高效修改需求**

学术会议的需求总是变化的很快，基本上每个会议需要的东西都不太一样而且很难去抽象出一个简单的模型来涵盖所有的功能。所以本系统放弃了传统的瀑布流的开发模式，使用和需求人员沟通更紧的敏捷开发模式，采用了 Django 敏捷开发框架，可以针对不同的系统快速交付出满足需要的系统版本。

1. **系统用户交互好，操作简单**

传统的会议管理系统的 UI往往还停留在上一个时代，使不太会操作的大龄使用者难以操作。这次开发特意照顾了各个年龄端的使用者的操作习惯。界面设计的是左边是菜单栏，右边是显示详细的信息。数据的增删改查如果出现错误都有相应的提示。

1. **展望**

​ 由于时间紧凑，很多功能仅仅是满足了需求，还有许多更多的特性可以加入：

1. **在线编辑**

现在的系统功能只能通过上传附件的方式上传论文，理想的系统中，参会者可以通过在线利用 markdown 或者 latax 等语言来进行文档的编辑，除此之外还需要增加一个在线的 word 编辑器来方便论文编辑。审核者也可以直接在线对论文进行批注，还可以约论文作者进行在线的协同批改论文，可以大大的加快论文评审的效率。

1. **增加统计模块**

​ 学术会议管理系统被大量的使用后，系统中会积累很多论文数据，可以通过分词的方式产生很多大数据，这就需要一个统计模块对论文的大数据进行分析。可以对这些数据进行一个初步的认识和了解，然后再深入的去了解和挖掘。这就需要用到数据挖掘的框架和其他大数据常用的技术，可以在以后加入。

**参考文献**

[1]黄钊.基于Web的学术会议管理系统的设计与实现[D].西安:西安电子科技大学,2010.

[2]童益民.基于SSH框架技术的企业信息管理系统的设计与实现[J].中国髙新技术企业,2011 (11): 69-71.

[3]翁晓红,郭新海等.生物医学工程国际学术会议管理系统的设计与实现[J].生物医学工程学杂志,2009 (4) :108-111.

[4]王进J2EE框架深度历险[M].南京:东南大学出版社,2009.1—2,

[5]任振宇,汪成曦.基于SSH框架的企业信息管理系统设计研究[J].科技资讯,2012(1) :15-16.

[6]常中华.基于Web的SQL数据库管理系统的设计[J].电脑知识与技术:学术交流,2006(4) :5-6,19.#-

[7]钱雪忠.数据库原理及应用[M].北京:北京邮电大学出版社[M],2009.264~265.;

[8]孙薇.基于PHP的会议管理系统设计与实现[D].西安:西安电子科技大学,2011.

[9]钟涛.校园会议管理系统的设计与实现[D].西安:电子科技大学,2013.

[10]李刚.疯狂Java讲义[M].北京:电子工业出版社,2011.8.

**致谢**

四年本科的学习和生活即将结束，在此论文完成之际，首先想到的是要感谢那些帮助和关怀过我的老师和同学们。

我首先要感谢我的论文导师苗春雨老师。她在论文的选题、研究思路的确定、资料的收集到论文的撰写、修改以及最后的定稿这一过程中，都对我严格要求，悉心指导，并给予了切实的建议，让我受益匪浅。她渊博的知识、严谨的治学态度给我留下了深刻的印象，她在学术研究上的经验和热心助人的精神，也使我深受教益。

此外，我还要感谢计算机系的许多其他老师，在我四年的本科学习期间，正是他们辛勤的传授、不倦的教诲，才为我最终完成毕业论文打下坚实的基础。

最后感谢各位评委老师对我毕业论文的审评。

1. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)