# 

**队 伍：Team18**

**成 员：田润琦，田辰星，高泽，周易，陈澈**

**书评影评网站开发计划**

|  |  |  |
| --- | --- | --- |
| **项目成员信息** | | |
| 学号 | 姓名 | 本文档中主要承担的工作内容 |
| 13241075 | 田辰星 | 风险管理和需求分析及项目估算 |
| 13241065 | 田润琦 | 系统设计 |
| 13081157 | 高泽 | 进度表及人员分工 |
| 14231038 | 周易 | 测试计划 |
| 12231016 | 陈澈 | 用例模型细化 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本变更历史** | | | | |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1.0 | 2016-10-10 | 田润琦 | 田辰星 | 项目计划书初次版本 |
| 1.1 | 2016-10-16 | 田辰星 | 田辰星 | 在上个版本的基础上，添加了用例模型的细化分析，根据SDP的规范对内容及顺序进行了一定程度的修改 |
|  |  |  |  |  |
|  |  |  |  |  |

[1](#_Toc464397484)

[1. 范围 6](#_Toc464397485)

[1.1 标识 6](#_Toc464397486)

[1.2 项目概述 6](#_Toc464397487)

[1.3 文档概述 6](#_Toc464397488)

[1.4 术语和缩略词 6](#_Toc464397489)

[1.5 引用文档 6](#_Toc464397490)

[1.6 软件需求规格说明 7](#_Toc464397491)

[2. 项目任务概要 7](#_Toc464397492)

[2.1 工作内容 7](#_Toc464397493)

[2.2 主要工作人员 7](#_Toc464397494)

[2.3 交付产品 8](#_Toc464397495)

[2.3.1 程序 8](#_Toc464397496)

[2.3.2 文档 8](#_Toc464397497)

[2.3.3 非移交产品 8](#_Toc464397498)

[2.4 运行与开发环境 8](#_Toc464397499)

[2.4.1 运行环境 8](#_Toc464397500)

[2.4.2 开发环境 9](#_Toc464397501)

[2.5 项目的最后完成期限 9](#_Toc464397502)

[3. 风险管理 9](#_Toc464397503)

[3.1项目规模及工作量估算是否准确 9](#_Toc464397504)

[3.2项目需求是否充分了解 9](#_Toc464397505)

[3.3项目是否存在评审标准 9](#_Toc464397506)

[3.4项目的整体设计 10](#_Toc464397507)

[3.5人员以及时间的分配 10](#_Toc464397508)

[3.6开发环境 10](#_Toc464397509)

[4. 监督和控制机制 10](#_Toc464397510)

[5. 软件开发详细计划 11](#_Toc464397511)

[5.1建立软件开发环境 11](#_Toc464397512)

[5.2需求分析 11](#_Toc464397513)

[5.2.1游客用户需求 11](#_Toc464397514)

[5.2.2注册用户角度需求 11](#_Toc464397515)

[5.2.3管理员角度需求 12](#_Toc464397516)

[5.2.4非功能性需求描述 12](#_Toc464397517)

[5.2.5技术需求 12](#_Toc464397518)

[5.3系统设计 13](#_Toc464397519)

[5.3.1 MVC结构设计 13](#_Toc464397520)

[5.3.2功能性设计（系统用例模型） 13](#_Toc464397521)

[5.3.3非功能性设计 14](#_Toc464397522)

[a) 系统安全性 14](#_Toc464397523)

[b) 保密性保证 15](#_Toc464397524)

[c) 私密性保证 15](#_Toc464397525)

[5.4软件开发活动计划 15](#_Toc464397526)

[5.4.1模块功能开发计划 15](#_Toc464397527)

[5.4.2开发时间表 17](#_Toc464397528)

[5.5测试计划 18](#_Toc464397529)

[5.5.1测试内容 18](#_Toc464397530)

[5.5.2测试1（用户模块功能测试） 18](#_Toc464397531)

[5.5.3测试2（管理员模块功能测试） 19](#_Toc464397532)

[5.5.4测试3（系统安全性测试） 19](#_Toc464397533)

[5.5.5测试4（系统性能测试） 20](#_Toc464397534)

[6. 资源计划 21](#_Toc464397535)

[7. 项目估算 21](#_Toc464397536)

[7.1规模估算 21](#_Toc464397537)

[7.2工作量估算 21](#_Toc464397538)

[7.3成本估算 21](#_Toc464397539)

[8. 进度计划 22](#_Toc464397540)

# 范围

## 标识

文档标题：书评影评网站系统开发计划书

版本号：1.1

## 项目概述

在互联网极度普及的今天，互联网上影视，读物等内容繁杂多样，质量也良莠不齐，我们希望建立一个集影评，书评为一体的影视书籍评论网站，让用户可以在网站上自由评论，交流心得。通过对用户的点赞评价，为用户提供当前热门，高质量的影视书籍推荐，并且根据标签，为用户个性化的定制个人的影视书籍库。

该网站的功能参考了豆瓣网，注册用户可以发表评论，收藏评论，follow其他用户，分享推荐文章。任何登录本网站的用户，都可以在首页上获取热门影视书籍信息，阅读推荐评论。

通过本网站对于影视书籍信息的汇总，可以为用户节约大量在互联网上寻找喜爱的电影书籍的时间，并且我们的推荐个具有个性化的特点。本网站旨在建立一个以影视书籍为主题的评论平台，方便用户找到自己喜爱的书籍影视，与共同爱好者交流沟通。

## 1.3 文档概述

本文档是书评影评网站系统的开发计划文档，用于指导书评影评网站系统的开发过程以及进行大致的需求分析，规模估算，风险管理等工作。文档的主要内容包括项目的概述，需求分析，用例模型，系统结构设计，人员分工，工作进度安排，规模估算以及风险管理等。

## 术语和缩略词

暂无

## 1.5 引用文档

由于项目尚未开始，无法确定具体会引用哪些文档，目前计划引用的文档有：计算机软件文档编制规范，JavaAPI手册，SSH架构的说明文档，Lucene架构的说明文档。

## 1.6 软件需求规格说明

该评论网站可以面向校园网，在负载量不会太高的状况下可以面向互联网。该软件中运用了Lucene检索框架，可以承载更多的数据量，保证网站运行的流畅，为用户提供良好的体验。

# 项目任务概要

## 2.1 工作内容

软件开发过程中须进行的主要工作如下：

1. 系统的需求分析
2. 系统的结构设计
3. 系统开发所采用的语言及架构
4. 系统的测试计划
5. 系统的规模估算以及风险管理
6. 系统的实际开发
7. 系统的测试与交付

主要编制的文档有：

1）《软件开发计划书》

2）《软件测试说明》

3）《软件需求规格文档》

4）《软件结构设计说明》

5）《软件测试报告》

## 2.2 主要工作人员

参与本项目开发的主要人员信息如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 工作任务 | 主要负责人 | 角色 | 角色要求 |
| 项目文档整理，排版，分发，项目进度控制，风险管理 | 田润琦，田辰星 | 协调者 | 项目责任感及时与准确性，贯穿全部开发过程 |
| 需求分析及功能模块划分 | 高泽，周易，陈澈 | 系统功能统筹者 | 需求把握能力和工程素养 |
| 数据模型和系统整体架构设计及接口模块划分 | 周易 | 需求分析和技术实现的协调者 | 实现难度评估和技术指导，良好沟通能力 |
| 系统实现（前端） | 高泽 | 技术实现者 | 全面前端开发技术经验 |
| 系统实现（后端） | 田润琦 | 主要技术实现者 | 后端技术和系统实现 |
| 系统测试 | 田辰星，陈澈 | 系统实现评估者 | 监督和评估系统实现完成正确性和完成度，回馈技术实现 |

## 2.3 交付产品

### 2.3.1 程序

采用SSH架构，Lucene架构编写的javaWeb项目一份。

### 2.3.2 文档

* 《网站管理员使用手册》
* 《网站用户使用手册》
* 《系统需求规格说明书》

### 2.3.3 非移交产品

* 《软件开发计划书》
* 《软件测试说明》
* 《软件需求规格文档》
* 《软件结构设计说明》
* 《软件测试报告》

## 2.4 运行与开发环境

### 2.4.1 运行环境

本系统为书评影评网站系统，因此系统应当能够在任何支持网络接入和浏览器的设备上运行

### 2.4.2 开发环境

本网站采用java语言编写，开发者需要java的开发工具jdk。

使用开发工具进行开发，我们统一使用eclipseMars的javaEE的IDE进行网站开发。

使用github团队进行代码管理。

## 项目的最后完成期限

软件工程课程要求的截止日期前。

# 风险管理

## 3.1项目规模及工作量估算是否准确

存在的问题：由于缺乏相关系统的编写经验，以及在开发过程中，成员的技术问题可能导致的学习时间以及相应的初学者对系统构建的理解方面等问题，可能导致对项目规模以及工作量的估算过于乐观。

解决方案：成员之间相互学习促进，多多了解相关系统开发过程中的难题。同时将任务分配严格到人，各成员务必在规定时限内完成指定任务，如未完成，则接下来的时间内争取补上。

## 3.2项目需求是否充分了解

存在的问题：开发过程中，可能由于对项目需求的理解角度不一致，导致功能实现的过程中出现分歧，从而延缓了项目的进度。

解决方案：前期进行充分的交流，对各个模块以及相应的功能实现达成一致。

## 3.3项目是否存在评审标准

存在的问题：由于是第一次接触软件工程，因此对在实现了项目的某个功能以及整体实现了之后的评审标准并不是很熟悉，因此可能导致的一个严重问题就是，由于不知道什么是好什么是坏，开发出的软件在质量上就得不到应有的保障。

解决方案：在项目进行初期尽快制定相应的评审标准，并和需求方达成一致。如果开发过程中发现现有的评审标准有没有考虑到的地方，尽快进行定制或者修改。

## 3.4项目的整体设计

存在的问题：项目的整体的架构影响了开发出产品的质量。由于我们水平的限制，因此可能在项目初期进行整体的设计时，考虑问题过于片面，对某一功能的实现需要的技术进行了过于乐观的估计等。因此会导致项目的拖延甚至夭折。

解决方案：充分向老师请教，了解此类产品的设计思想和手段。在遇到技术性困难时，多多向大佬寻求帮助。同时在设计之初，进行详尽的分析，并且撰写项目需求说明，开发计划等文档。

## 3.5人员以及时间的分配

存在的问题：小组成员之间了解程度不够，因此在进行任务以及时间分配时，可能会存在分配的不合理性，从而使项目进度缓慢。

解决方案：小组成员在进行任务认领时充分考虑到自己的长处。同时开发过程中，团队可以定期进行交流，将开发过程中遇到的问题拿出来大家一块进行讨论。

## 3.6开发环境

存在的问题：开发工具与期望不符，计算机故障等等。

解决方案：项目进行之前充分了解开发项目所需要的各种硬软件的要求。充分了解所要用到的开发工具。计算机出现故障及时找维修人员进行排除。

# 监督和控制机制

暂无

# 软件开发详细计划

## 5.1建立软件开发环境

本网站采用java语言编写，开发者需要java的开发工具jdk。

使用开发工具进行开发，我们统一使用eclipseMars的javaEE的IDE进行网站开发。

使用github团队进行代码管理。

## 5.2需求分析

### 5.2.1游客用户需求

我们的网站定位是一个书评，影评的评论网站，游客一般通过搜索引擎获取到我们的网站，而且游客用户一般只会浏览我们的首页，因此我们需要把所有对游客用户有用的信息全部展现在首页。搜索到我们网站的用户自然就是希望我们可以为他推荐当前热门书籍，热门电影。我们需要在我们的首页，游客用户最容易看到的地方，为他提供书籍，电影的top50排行。我们的网站是一个评论网站，因此也会有用户希望阅读优质评论，因此，我们需要将筛选出的优质评论信息放在一个热门评论区，除此之外，我们可以为游客用户推荐我们网站的优质作者，这也可以吸引起游客用户的兴趣。当然，最重要的一点，我们需要把我们网站的注册，登录展现在游客用户的眼前，方便用户注册或者登录，这也是挖掘潜在用户的重要措施。

综上所述，我们需要展现给游客用户的信息有，热门书籍，热门影视排行，热门评论排行，优秀作者排行，注册，登录链接。

### 5.2.2注册用户角度需求

用户注册了我们的网站，自然是希望参与我们网站的其他用户的群体，一起探讨电影书籍的主题。因此我们要给注册用户提供更多的交互功能。

用户登录后，可以看到主界面，是一些影片或者书籍的推送，用户可以点击进行查看，并且进行加入收藏，分享，点赞，评论等操作。此外，主界面中还会根据一定的算法推送他人的状态信息，比如，某人最近的评论，收藏，还可以对其他用户进行关注。在主界面中还包含一个通往个人主页的链接，点击可以前往个人主页。个人主页中包含的信息主要是，个人收藏的影片，书籍信息，撰写个人对某个影片或者书籍的认识，个人资料的编辑，以及他人访问主页是的权限设置。

### 5.2.3管理员角度需求

系统管理员登陆之后可以进入到管理员界面。主要包括网站内容的管理以及用户的管理。用户管理方面，可以更改用户的权限，或者删除用户。网站内容管理方面，主要涉及网站内容的审核以及更新。

### 5.2.4非功能性需求描述

1. 系统安全性：由于使用系统牵涉到用户的个人信息，账户密码等，以及用户个人资料中的信息，因此要确保系统的安全性。
2. 系统易用性：系统面向的是广大普通用户，他们并不一定具备计算机相关专业的背景知识，因此在对系统进行设计时，要简单易用，提供良好的易于理解的交互接口，使用户能够尽快的熟悉相关的操作。
3. 系统可扩展性：在开发过程中，可以考虑可扩展性。因为随着发展，后续还可能进行其他方面的分享与交流，而且在此基础上，甚至还可以扩展一个用户之间的交易平台。因此需要系统具有一定的可扩展性，方便后续的调整和发展。
4. 系统的易恢复性：本系统中牵涉到大量的用户数据，这些数据基本存储在本地服务器中。

当遇到一些不可抗因素导致数据库损坏时，如没有事先采取一定的措施，将导致数据的丢失，严重影响到很长一段时间内的系统使用，以及造成非常不好的用户体验和相关的影响。因此，为了防止这种情况的发生，数据应当定时备份另一个或者多个地点的数据库中。在一个数据库收到损坏以后，可以及时从其他数据库中获取数据使系统继续正常运转下去。

### 5.2.5技术需求

本网站采用java编写，用到了以下的技术框架：

SSH框架：我们的网站采用MVC的设计结构，使用Spring框架，Struts2框架对于请求信息进行统一的分发。这里我们不准备采用hibernate框架，而是采用my batis框架，这样可以自行书写sql语句，灵活性更好。

数据库结构：我们计划针对博文，用户各建立一张总表，总表中用不同的id进行管理，这样针对不同的功能，我们只需要获取其特有的id，建立联系即可。

Lucene检索（待定）：lucene框架是一个建立检索的框架，当我们的数据量很大时就可以通过这个框架加快检索的效率，是否要加入这个框架目前待定。

## 5.3系统设计

## 5.3.1 MVC结构设计

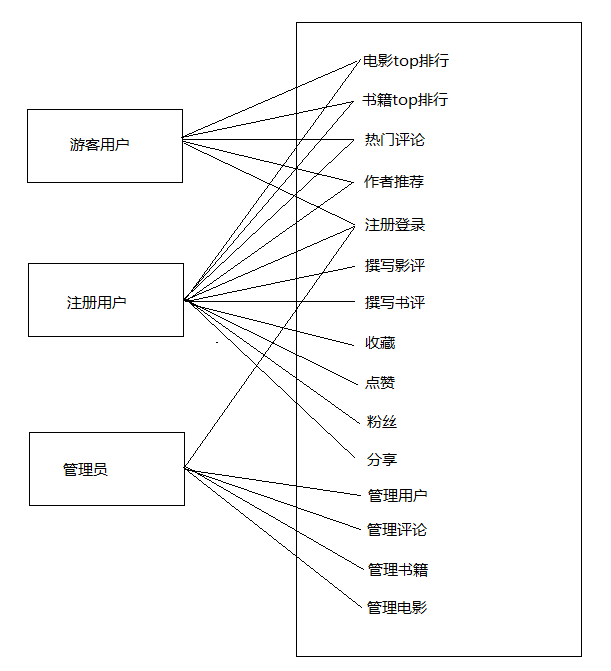
本网站计划采用Model-View-Controller的网站设计结构进行设计。Model为业务模型，View为用户界面，Controller为控制器。

下面说一下网站采用MVC模型的处理请求的流程：

View层（视图层）也就是我们的Html页面，接收到了信息和提交动作，通过表单提交或者Ajax提交，将信息提交到了后台，这时进入了我们的SSH框架。通过spring配置文件，将不同的请求讲给不同的Controller进行处理，Controller层主要负责业务逻辑的处理，如果需要查找或者获取数据，我们需要进入Model层，用到了Mybatis框架来访问数据库，之后访问到的数据传回到Controller继续进行处理，处理完成后传回到View层，完成显示。

## 5.3.2功能性设计（系统用例模型）

网站从三个主要用户的角度对于功能进行了划分，其中为游客用户，注册用户，管理员。注册用户可以进行用户所能进行的各种功能，而游客用户没有权限对网站的内容进行改动，即添加评论等操作。管理员用户需要对网站进行维护，因此拥有对于网站内容的增删改的权限，比如删除评论，修改用户等操作，具体划分附下图所示：



系统用例模型图 1

接下来主要对系统中的个用户用例进行描述：

1. 撰写影评书评

　　用例目标：当注册用户开始撰写一份影评书评时用例开始。它处理有关影评书评的发表问题，当注册用户结束编辑以后用例结束。

　　前提条件：注册用户登录进入系统

主路径：

1.当注册用户选择“发表书评影评”时，用例开始

2.注册用户编写影评书评内容

3.注册用户选择“发表”或者“放弃发表”

成功后件：如果注册用户选择发表，则增加一份书评影评，被保存在系统里，并返回给注册用户“发表成功”的通知

1. 电影书籍排行

　　用例目标：对电影书籍按照点赞收藏数量进行排行

　　前提条件：无

主路径：

1. 按一定权重讲点赞收藏相加得到“综合评分”
2. 系统按照”综合评分”高低来排序并以每天零点刷新显示在主页对应版块

成功后件：无

1. 注册登录

　　用例目标：用户注册登录

　　前提条件：用户进入登陆注册界面

主路径：

1. 系统显示登录界面
2. 若用户无账号，则选择注册，若有账号则跳过3-9步
3. 用户输入账号
4. 系统判定账号是否合法，若否，显示提示，返回第3步
5. 用户输入密码
6. 系统判定密码是否合法，若否，显示提示，返回第4步
7. 用户确认密码
8. 系统判定两次密码是否一致，若否，显示提示，返回第7步
9. 注册成功，系统保存账户密码，并返回提示信息
10. 用户输入账户密码
11. 系统验证身份信息，若密码正确，显示登陆成功，若否，显示提示，返回第10步

成功后件：保存用户账号密码，替换最后登录时间，进入系统

1. 点赞，收藏，关注

　　用例目标：注册用户为书评影评，以及书评影评下的评论点赞，或者关注作者

　　前提条件：注册用户登录进入系统

主路径：

1. 注册用户浏览书评影评以及评论，或者看到作者的某篇文章
2. 注册用户选择为文章，评论点赞，收藏文章，或者关注作者

成功后件：点赞数+1，收藏+1，注册用户可以在“我的收藏”下查看收藏，或者作者关注+1，注册用户可以在我的关注下查看作者

1. 作者推荐

　　用例目标：

　　前提条件：无

主路径：

1. 按照关注数对作者进行排序
2. 每天零点更新显示在主页的作者推荐上

成功后件：无

1. 热门评论

　　用例目标：

　　前提条件：无

主路径：

1. 按照点赞数对评论进行排序
2. 将点赞数前三的评论每天零点刷新置顶

成功后件：无

1. 分享

　　用例目标：

　　前提条件：注册用户进入系统

主路径：

1. 注册用户点击分享按钮
2. 系统弹出社交网站选项
3. 用户选择
4. 跳转至社交网站接口

成功后件：无

1. 管理用户

　　用例目标：管理员管理用户

　　前提条件：管理员进入系统

主路径：

1. 管理员选择封禁用户的发文权力，评论权利，以及封号
2. 系统通知用户，并返回管理员操作成功消息

成功后件：用户数据相应权限修改并保存

1. 管理书籍，电影，评论

　　用例目标：

　　前提条件：管理员进入系统

主路径：

1. 管理员封禁相应书评，影评，评论
2. 系统返回操作成功信息，并发送给文章作者/评论者提示信息

成功后件：书评影评评论数据修改并保存

## 5.3.3非功能性设计

### 系统安全性

作为游客用户，只能参观浏览网站信息，无法删除或修改其他用户的内容，这一点上保证了网站数据的安全性。

注册用户需要使用用户名，密码进行登录，我们采用的Mybatis属于预编译生成sql语句的一种框架，可以有效防止sql注入的漏洞，因此这一点上也相对安全的保护了注册用户的账号安全。

关于管理员，我们规定管理员只能从本地服务器上登录管理员界面修改信息，这也保护了网站数据的安全性。

### 保密性保证

为了防止用户的密码在数据库中被直接窃取，或者在数据传输的过程中被获取，我们可以采用java提供的md5加密，将用户的密码信息加密后再进行传输，保证用户账号的安全性。

### 私密性保证

任何用户浏览网站信息，只能看到其他用户愿意公开的影视书籍评论，分享的书籍电影等，无法查看其他用户私密的收藏，个人的账号信息等，这一点上保证了用户的私密性。

## 5.4软件开发活动计划

### 5.4.1模块功能开发计划

|  |  |  |
| --- | --- | --- |
| 模块（页面） | 描述 | 功能实现 |
| 首页(index) | 是大多数用户登录本网站看到的第一页，因此需要把最有用，最精华的信息展现在这一页。 | 书籍Top排行，电影top排行，评论top排行，热门作者推荐，登录注册模块 |
| 书籍Top排行页面 | 显示书籍的排行，并且可以为用户提供更加详尽的书籍信息，登录的用户可以使用收藏功能，书籍的分享功能。 | 书籍的top排行，书籍的链接，书籍的收藏，书籍的分享 |
| 电影Top排行页面 | 显示电影的排行，并且可以为用户提供更加详尽的电影信息，登录的用户可以使用收藏功能，电影的分享功能。 | 电影的top排行，电影的链接，电影的收藏，电影的分享 |
| 评论Top排行页面 | 显示评论的排行，用户可以查看热门评论，可以查看评论的作者，登陆的用户可以收藏，分享，点赞评论。 | 评论的top排行，评论的链接，评论的收藏，评论的分享，评论的点赞，作者链接 |
| 热门作者页面 | 显示热门的作者，可以根据粉丝的数量进行筛选，可以查看作者，登录的用户可以follow该作者。 | 热门作者列表（根据粉丝数量排行），作者链接，follow选项。 |
| 书籍页面 | 显示书籍的信息，提供书籍的收藏，分享功能，检索出该书籍的评论放在书籍信息下，评论可以链接到详细评论。 | 书籍信息，检索的书籍评论，书籍的收藏，书籍的分享，评论的链接 |
| 电影页面 | 显示电影的信息，提供电影的收藏，分享功能，检索出该电影的评论放在电影信息下，评论可以链接到详细评论。 | 电影信息，检索的电影评论，电影的收藏，电影的分享，评论的链接 |
| 登录页面 | 输入用户名，密码进行登录 | 用户名，密码，登录按键 |
| 注册页面 | 输入信息，完成注册 | 用户名，密码，个人信息，注册按键 |
| 个人主页 | 登录进入之后，链接到个人主页，在个人主页默认显示空间动态。 | 左侧建立treeview，有空间动态，个人动态，我的收藏，我的点赞，我的follow，上方功能区有用户设置，发表评论。 |
| 用户设置页面 | 可以修改用户信息，修改密码。 | 提供可以修改的用户信息，修改按键。 |
| 个人动态页面 | 显示个人的收藏，点赞，发表评论的状态，评论可以点击进入查看详细信息，个人动态页面的内容会显示在动态空间中。 | 显示个人收藏，点赞，发表评论，评论有链接。 |
| 我的收藏页面 | 查看我收藏的书籍电影，可以删除我的收藏。 | 显示我收藏的电影书籍，有删除按键。 |
| 我的点赞 | 查看我点赞的书籍电影，可以取消点赞。 | 显示我点赞的书籍电影，有取消点赞功能。 |
| 我的follow页面 | 查看我follow的用户的列表，可以点击进入该用户的个人动态页面。 | 用户列表，该用户主页的链接。 |
| 发表评论页面 | 可以在该页面发表书籍影视评论。 | 书籍影视类型选择，评论撰写区，确认按键。 |
| 管理员主页 | 管理员可以在该页面管理用户，评论电影，对于这些信息的查询都是基于模糊查询，管理员基于查询结果对数据库的内容进行管理。 | 四个区域：  管理用户，支持对用户的模糊查询，查询出用户后可以修改用户的信息，删除用户。  管理评论：支持对评论的模糊查询，查询出评论后可以删除评论。  管理书籍：支持对书籍的模糊查询，查询出书籍后可以删除评论。添加书籍按键。  管理电影：支持对电影的模糊查询，查询出电影后可以删除评论。添加电影按键。 |

### 5.4.2开发时间表

|  |  |  |
| --- | --- | --- |
| 时期 | 工作内容 | 工作成果 |
| 10% | 搭建网站开发环境，搭建好SSH框架，建立好数据库的表单。 | 可以使用SSH框架输出“HelloWorld”。 |
| 20% | 设计网页页面，将网页页面与Spring框架建立好分发的javabean，设计好后端的Controller，以及Controller实现的接口。 | 完成后端对于前端请求的分发。 |
| 40% | 书写Controller的业务逻辑，书写Mybatis对于数据库的操作，完成前端的业务请求。 | 可以基本实现网站功能。 |
| 20% | 进行测试，修改开发过程中的bug，完善网站的功能，完成网站的搭建 | 完全实现网站功能，测试后没有bug。 |
| 10% | 完成测试报告，撰写用户手册。 | 交付产品。 |

## 5.5测试计划

### 5.5.1测试内容

主要对于以下四个模块进行测试：

用户模块，管理员模块，系统安全性，系统性能。

### 5.5.2测试1（用户模块功能测试）

测试模块：用户功能模块

测试人：田润琦

|  |  |
| --- | --- |
| 功能 | 输入 |
| 用户登录 | 1.已经注册的用户名和正确的密码  2.已经注册的用户名和错误的密码  3.未注册的用户名 |
| 搜索书籍、电影 | 1. 对书籍、电影编码：   （1）存在的编码  （2）不存在的编码  2.对书籍、电影名：  （1）完全正确或部分完全正确的名称  （2）部分正确的名称  （3）难以搜索的名称 |
| 个人信息管理 | 1. 已登录用户的个人信息   （1）个人信息的查询  （2）个人信息的更新与验证   1. 未登录用户点击个人信息 |
| 评论编辑、修改与互动 | 1. 撰写评论 2. 查看自己或他人的评论 3. 点赞   在评论下方进行交流 |
| 个人动态 | 1. 对文章或用户进行分享 2. 发表评论或进行收藏   查看个人动态 |
| 用户间互动 | 1.多个用户  2.用户A关注用户B  3.用户A查看用户B的动态 |
| 系统反馈 | 1.用户输入反馈并提交 |

### 5.5.3测试2（管理员模块功能测试）

测试模块：管理员模块

测试人：田辰星

|  |  |
| --- | --- |
| 功能描述 | 计划测试用例 |
| 管理员登陆 | 1. 输入正确的管理员账号和密码 2. 输入正确的账号和错误的密码 3. 输入错误的管理员账号 |
| 管理员账号注销 | 注销管理员登陆 |
| 管理员管理界面测试 | 1. 管理员暂停使用某用户账号 2. 管理员对非法作品予以下架 3. 查看用户反馈并回复 |

### 5.5.4测试3（系统安全性测试）

测试系统的登录安全性，密码的加密性：

测试人：高泽

|  |  |
| --- | --- |
| **测试内容** | **输入** |
| 非法访问 | 1、未注册用户的用户名 |
| 非法访问 | 1、已注册用户的用户名  2、非该用户对应的密码 |
| 非法访问 | 1、已注册用户的用户名  2、该用户对应的密码 |
| 用户资料加密情况 | 1、用任一用户名登录系统之后，随便点击页面的内容，查看是否能获得别的用户的私人信息 |
| 用户资料加密情况 | 1、任一用户的密码忘记，点击找回密码，系统发送给用户修改密码的链接，查看用户收到的信息里面有无其他已注册用户的私人信息 |
| 用户资料加密情况 | 1. 搜索栏输入非法的一些任意字符，查看搜索结果是否涉及私人信息 |

### 5.5.5测试4（系统性能测试）

主要测试系统的启动时间、页面响应时间、并发访问服务器负载情况等。

测试人：周易

|  |  |
| --- | --- |
| **测试内容** | **输入** |
| 启动时间 | 多次启动后台服务器然后关闭再启动 |
| 页面响应时间 | 所有可能的操作  1.注册  2.登陆（成功或失败）  3.评论发表  4.搜索  5.点赞  6.关注  7.反馈  此外，其余产生数据交互的操作仍需记录 |
| 数据更新时间 | 1.用户A向用户B发送信息后，到用户B收到信息的时间  2.用户A的动态能够被用户B看到的时间 |
| 并发访问服务器负载 | 少于20人同时访问本系统 |
| 并发访问服务器负载 | 20—60人同时访问本系统 |
| 并发访问服务器负载 | 多于60人同时访问本系统 |

# 资源计划

暂无

# 项目估算

## 7.1规模估算

系统的规模估算采取按照代码量估算的方式进行，首先对系统各部分的代码量进行估算：

1. 系统使用者主要包括游客，普通用户以及管理员，为了对这些用户进行区分，在代码中需要进行相应的体现。预计代码行数200行；
2. 系统需要显示各种信息提供给用户，其中包括公共的主界面，用户的个人主页，管理员的管理界面。预计代码行数1000行；
3. 系统需要保存用户的个人信息，牵涉到数据库相关的操作。预计代码行数600行；
4. 系统需要允许管理员对用户权限进行设置，对显示内容进行设置等，管理员权限牵涉到系统的方方面面。预计代码行数800行；
5. 系统可以正常使用，有美观的界面，并且点击之后可以显示相关信息。预计代码行数1000行。

综上，最终预计代码行数为3600行。

## 7.2工作量估算

工作量估算与项目规模相关联。在估算出项目规模最终代码行数位3600，以及考虑到开发过程中需要的各种文档编写的情况下，团队中每人的个大致为900行代码以及一篇文档。

## 7.3成本估算

成本估算采用自下而上和参数法的结合模型,步骤如下:

对任务进行分解

估算每个任务的成本Ei

直接成本=E1+E2+……+ Ei+……+ En

间接成本=直接成本\*间接成本系数

项目总估算成本= 直接成本+间接成本

项目总报价=项目总估算成本+风险利润

风险利润=利润+风险基金+税

先估算规模Qi，然后估算成本Ei= Qi \*人力成本参数唯一估计值：PERT算法: Qi=(Max+4Avg+Min)/6

Qi=（5000+3600\*4+3000）/6=3733

由于是作业，因此人力成本参数唯一估计值定为1元/行，因此最终的成本估计为3733元。

# 进度计划

|  |  |  |  |
| --- | --- | --- | --- |
| 任务开始时间 | 任务结束时间 | 工作任务 | 主要负责人 |
| 2016.10.10 | 2016.10.17 | 完成项目开发计划文档 | 全体成员 |
| 2016.10.18 | 2016.10.25 | 完成项目需求规格说明书 | 田润琦 |
| 2016.10.26 | 2016.11.01 | 完成项目计划书 | 田辰星 |
| 2016.11.02 | 2016.11.09 | 完成项目测试计划书 | 高泽 |
| 2016.11.10 | 2016.11.17 | 完成系统数据模型设计和功能模块切分 | 周易 |
| 2016.11.18 | 2016.12.01 | 编码实现 | 全体成员 |
| 2016.12.02 | 2016.12.15 | 完成系统测试工作及测试文档及维护文档 | 全体成员 |