**“学院路”校园购物平台**

**软件开发计划**

**Version：1.0**

|  |  |  |
| --- | --- | --- |
| **项目组成员信息** | | |
| **小组名称** | **北航软件开发小组05组** | |
| **学号** | **姓名** | **本文档中主要承担的工作内容** |
| 14061043 | 冯岩 | 文档主写人（40%） |
| 13051259 | 江涛 | 文档审核人（15%） |
| 13071079 | 卢嵩 | 文档审核人（15%） |
| 14061048 | 张阳 | 文档审核人（15%） |
| 14061036 | 韩旭 | 文档审核人（15%） |
|  |  |  |
|  |  |  |

2016-09

目录

[1 引言 1](#_Toc463789352)

[1.1软件标识 1](#_Toc463789353)

[1.2系统概述 1](#_Toc463789354)

[1.2.1项目背景 1](#_Toc463789355)

[1.2.2 软件功能及特性 1](#_Toc463789356)

[1.2.3 当前与计划运行现场 1](#_Toc463789357)

[1.3文档概述 2](#_Toc463789358)

[1.4与其他文档的关系 2](#_Toc463789359)

[1.5 基线 2](#_Toc463789360)

[1.5.1 软件需求规格 2](#_Toc463789361)

[2 引用文件 3](#_Toc463789362)

[3 产品交付 3](#_Toc463789363)

[3.1程序 3](#_Toc463789364)

[3.2文档 3](#_Toc463789365)

[3.3服务 4](#_Toc463789366)

[3.4非移交产品 4](#_Toc463789367)

[3.5验收标准 5](#_Toc463789368)

[3.6最终交付期限 5](#_Toc463789369)

[4 所需工作概述 5](#_Toc463789370)

[4.1 软件需求与约束 5](#_Toc463789371)

[4.2项目文档编制的需求与约束 6](#_Toc463789372)

[4.3在系统生命周期中的地位 6](#_Toc463789373)

[4.4开发环境需求与约束 6](#_Toc463789374)

[4.4.1 硬件环境 6](#_Toc463789375)

[4.4.2软件环境 6](#_Toc463789376)

[4.5项目进度安排及资源的需求和约束 7](#_Toc463789377)

[4.5.1需求分析 7](#_Toc463789378)

[4.5.2 系统设计 7](#_Toc463789379)

[4.5.3 编码及测试阶段 7](#_Toc463789380)

[4.5.4 文档、产品部署 7](#_Toc463789381)

[4.5.5 项目总结 7](#_Toc463789382)

[4.6其他需求 7](#_Toc463789383)

[5软件开发计划 7](#_Toc463789384)

[5.1软件开发过程 7](#_Toc463789385)

[5.2软件开发总体计划 8](#_Toc463789386)

[5.2.1软件开发方法 8](#_Toc463789387)

[5.2.2软件产品标准 8](#_Toc463789388)

[5.2.3可重用的软件产品 8](#_Toc463789389)

[5.2.3.1吸纳可重用的软件产品 8](#_Toc463789390)

[5.2.3.2开发可重用的软件产品 9](#_Toc463789391)

[5.2.4处理关键性需求 9](#_Toc463789392)

[5.2.5计算机硬件资源利用 10](#_Toc463789393)

[5.2.6记录原理 10](#_Toc463789394)

[5.2.7需方评审途径 10](#_Toc463789395)

[6实施详细软件开发活动的计划 11](#_Toc463789396)

[6.1 项目计划和监督 11](#_Toc463789397)

[6.1.1 软件开发计划 11](#_Toc463789398)

[软件开发计划表 12](#_Toc463789399)

[6.1.2 CSCI测试计划 12](#_Toc463789400)

[CSCI测试计划表 12](#_Toc463789401)

[6.1.3 系统测试计划 13](#_Toc463789402)

[系统测试计划表 13](#_Toc463789403)

[6.1.4 软件安装计划 13](#_Toc463789404)

[6.1.5 软件移交计划 13](#_Toc463789405)

[6.1.6 跟踪和更新计划 14](#_Toc463789406)

[6.2 建立软件开发环境 15](#_Toc463789407)

[6.2.1 软件工程环境 15](#_Toc463789408)

[6.2.2 软件测试环境 15](#_Toc463789409)

[6.2.3 软件开发库 15](#_Toc463789410)

[本Web系统开发使用CodeIgniter Web Framework库。 15](#_Toc463789411)

[6.2.4 软件开发文档 15](#_Toc463789412)

[6.2.5 非交付软件 15](#_Toc463789413)

[6.3 系统需求分析 15](#_Toc463789414)

[6.3.1 用户输入分析 15](#_Toc463789415)

[6.3.2 运行概念 15](#_Toc463789416)

[6.3.3 系统需求 16](#_Toc463789417)

[6.4 系统设计 16](#_Toc463789418)

[6.4.1 系统级设计决策 16](#_Toc463789419)

[6.4.2 系统级体系设计 16](#_Toc463789420)

[6.5 软件需求分析 17](#_Toc463789421)

[6.6 软件设计 17](#_Toc463789422)

[6.6.1 CSCI级设计决策 17](#_Toc463789423)

[6.6.2 CSCI级体系结构设计决策 17](#_Toc463789424)

[6.6.3 CSCI详细设计 19](#_Toc463789425)

[6.7 软件实现和配置项设置 19](#_Toc463789426)

[6.7.1 软件实现 19](#_Toc463789427)

[6.7.2 配置项测试准备 20](#_Toc463789428)

[6.7.3 配置项测试执行 20](#_Toc463789429)

[6.7.4 修改和再测试 20](#_Toc463789430)

[6.7.5 配置项测试结果分析与记录 20](#_Toc463789431)

[6.8 软件使用准备 20](#_Toc463789432)

[6.8.1 可执行软件的准备 20](#_Toc463789433)

[6.8.2 用户现场的版本说明的准备 20](#_Toc463789434)

[6.8.3 用户手册的准备 20](#_Toc463789435)

[6.9 软件质量保证 20](#_Toc463789436)

[6.9.1 软件质量保证评估 20](#_Toc463789437)

[6.9.2 软件质量保证记录 21](#_Toc463789438)

[6.9.3 软件质量保证的独立性 21](#_Toc463789439)

[6.10 联合评审 21](#_Toc463789440)

[6.10.1 联合技术评审 21](#_Toc463789441)

[6.10.2 联合管理评审 21](#_Toc463789442)

[6.11 文档编制 22](#_Toc463789443)

[7进度表和活动网络图 22](#_Toc463789444)

[8 项目组织和资源 23](#_Toc463789445)

[8.1项目组织 23](#_Toc463789446)

[8.2项目资源 24](#_Toc463789447)

[8.2.1人力资源情况 24](#_Toc463789448)

[人力资源情况表 24](#_Toc463789449)

[8.2.2 开发人员所需设备 24](#_Toc463789450)

[8.2.3 其他所需资源 25](#_Toc463789451)

[9培训 25](#_Toc463789452)

[9.1技能要求 25](#_Toc463789453)

[9.2培训计划 25](#_Toc463789454)

[10项目估算 25](#_Toc463789455)

[10.1规模估算 25](#_Toc463789456)

[10.2工作量估算 25](#_Toc463789457)

[工作量估算表 25](#_Toc463789458)

[10.3成本估算 26](#_Toc463789459)

[成本估算表 26](#_Toc463789460)

[10.4关键计算机资源估算 27](#_Toc463789461)

[11风险管理 27](#_Toc463789462)

[风险管理表 27](#_Toc463789463)

[12支持条件 28](#_Toc463789464)

[12.1计算机系统支持 28](#_Toc463789465)

[12.1.1 硬件环境 28](#_Toc463789466)

[12.1.2软件环境 28](#_Toc463789467)

[12.2需要需方承担的工作和提供的条件 29](#_Toc463789468)

[12.3需要分包商承担的工作和提供的条件 29](#_Toc463789469)

版本变更历史

需要为每个文档指定审核人，并做认真审核。请组长协调安排。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| V1.1 | 2016.10.3 | 冯岩 |  | 最初版本 |
| V1.2 | 2016.10.6 | 冯岩 |  | 修订版本 |
| V1.3 | 2016.10.8 | 冯岩 |  | 修订版本 |
| V1.4 | 2016.10.9 | 冯岩 | 江涛 | 最终版本 |
|  |  |  |  |  |
|  |  |  |  |  |

# 1 引言

## 1.1软件标识

文档标识号：A2016-10-10-00-14

文档标题：软件设计说明

项目/产品中文全称：“学院圈”校园购物平台

项目/产品英文全称："College Road" campus shopping platform

项目/产品英文简称：LMS\_1.0

项目/产品编码：201610100014

## 1.2系统概述

本项目由北京航空航天大学软件工程2016秋季课程Team05小组负责开发。

### 1.2.1项目背景

随着经济的发展和网络的进步，电子商务也逐渐成熟起来，网购因快捷、便利逐渐被人喜爱与接受，已成为人们生活的一部分。中国互联网络信息中心统计显示，截至2015年6月底中国网络购物网民规模达到4.71亿人，网络购物使用率提升至45.9%，这两大因素成为助推网络购物市场规模继续扩张的主要驱动力。

同时，当今大学生由于生活习惯、消费观念以及对于新技术的推崇等原因，已经逐渐成为网购平台的主要用户。调查显示当今超九成大学生有过网购经历。然而，当下却罕见专门针对大学生需求的在线购物平台。在此背景下，本项目应运而生。

### 1.2.2 软件功能及特性

本系统目的主要是为用户提供在线的商品信息检索和网上购物，通过吸引用户浏览商品，诱导用户达成网上购物，从而实现网站盈利。

本平台具有强大灵活的网上购物功能、会员功能和网站内容管理功能，可满足各种不同网上商店的需要。具备一般商店系统所具有的商品分类、商品检索、商品搜索、购物车管理、订单提交与取消、在线支付与申请退款、会员系统等功能。

我们牢牢把握目标用户为在校大学生的特点，运用现代流行理念和元素，成熟的网络开发技术体现网站现代不失传统，大众不失前卫的特点，并做好各个栏目的导航，突出网站的重要职能和重要内容，使得网站能够同今日高密度信息与市场营销行业完美结合。

### 1.2.3 当前与计划运行现场

本软件的目标用户为在校大学生。首先计划在北京航空航天大学校内进行推广和试运营，根据用户反馈情况进行升级和服务器扩容后可以向更多高校进行推广。

## 1.3文档概述

在软件的生产过程中，总是伴随着大量的信息要记录、要使用。因此，[软件文档](https://www.baidu.com/s?wd=%E8%BD%AF%E4%BB%B6%E6%96%87%E6%A1%A3&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3mHwbuj99rj9WnH61rHKh0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHm1PHfdrjm4" \t "http://zhidao.baidu.com/_blank)在产品的开发生产过程中起着重要的作用。   
1.提高软件开发过程的能见度。把开发过程中发生的事件以某种可阅读的形式记录在文档中。管理人员可把这些记载下来的材料作为检查软件开发进度和开发质量的依据，实现对软件开发的工程管理。   
2.提高开发效率。[软件文档](https://www.baidu.com/s?wd=%E8%BD%AF%E4%BB%B6%E6%96%87%E6%A1%A3&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3mHwbuj99rj9WnH61rHKh0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHm1PHfdrjm4" \t "http://zhidao.baidu.com/_blank)的编制，使得开发人员对各个阶段的工作都进行周密思考、全盘权衡、从而减少返工。并且可在开发早期发现错误和不一致性，便于及时加以纠正。   
3.作为开发人员在一定阶段的工作成果和结束标志。   
4.记录开发过程中的有关信息，便于协调以后的软件、开发、使用和维护。   
5.提供对软件的运行、维护和培训的有关信息，便于管理人员、开发人员、操作人员、用户之间的协作、交流和了解。使软件开发活动更科学、更有成效。   
6.便于潜在用户了解软件的功能、性能等各项指标，为他们选购符合自己需要的软件提供依据。   
 在从项目开发阶段到最后软件的正式发布期间，做好项目的保密工作，小组成员对所有项目所有相关文档进行加密，做好备份工作。

## 1.4与其他文档的关系

本文档论述项目组软件开发计划，主要涉及项目背景与市场需求进度与活动安排、软件功能、所需资源与技能、风险管理、成本估算等内容，是整个项目的计划书与基础。

之后的软件需求与规格文档是对所开发软件的功能、性能、用户界面及运行环境等做出详细的说明，它是对本文档涉及的项目背景与需求一项的具体阐述；软件结构设计文档则对于系统进行更为详细的分析与设计，是对本文档中软件功能的落实；软件测试说明文档主要对于具体实现完成后的测试工作进行规划，而软件测试是本文档工作计划中开发工作的最后一环。

2 引用文件  
[1] 杨叶中。 《图书馆管理系统开发计划》。2013年9月。  
[2] 朱蔷。《基于WEB的北航二手书交易平台》。2013年9月。

[3] 王裔鸣。《餐厅服务系统》。 2013年9月。

# 3 产品交付

## 3.1程序

项目名称为“学院路”购物平台，编程语言方面，主要使用PHP，JAVASCRIPT，HTML，数据库类型MYSQL。

主要功能：

用户功能：注册新用户，登录系统，浏览公告，发表留言，添加修改和删除购物车的信息，提交订单。

浏览者功能：查看网站主页，商品信息查询，浏览公告信息

购物系统管理后台：管理员注册系统，管理员登录系统，用户管理系统，订单管理系统，商品管理系统，公告管理系统。

## 3.2文档

1. 软件开发计划：为软件项目实施方案制订出具体计划，应该包括各部分工作的负责人员、开发的进度、开发经费的预算、所需的硬件及软件资源等。
2. 软件需求规格说明：对所开发软件的功能、性能、用户界面及运行环境等做出详细的说明。它是在用户与开发人员双方对软件需求取得共同理解并达成协议的条件下编写的，也是实施开发工作的基础。该说明书应给出数据逻辑和数据采集的各项要求，为生成和维护系统数据文件做好准备。
3. 软件结构设计说明：该说明书是概要实际阶段的工作成果，它应说明功能分配、模块划分、程序的总体结构、输入输出以及接口设计、运行设计、数据结构设计和出错处理设计等，为详细设计提供基础。
4. 软件测试说明：为做好集成测试和验收测试，需为如何组织测试制订实施计划。计划应包括测试的内容、进度、条件、人员、测试用例的选取原则、测试结果允许的偏差范围等。
5. 软件用户手册：软件用户手册是用户了解软件的重要途径之一。具体应包括：使用的开发语言与工具；软件适用的操作系统。用户使用时需要做的配置，或者需要安装的组件。

## 3.3服务

向用户提供的服务有：

1. 系统的配置，安装，保修，及其他的运行需求；

2. 提供系统使用培训，使受训人员能够熟练使用系统，提高工作效率。

3. 客户可以在工作时间向产品客服人员提出问题并获得解答。

4. 获取系统使用中的问题，提供补丁程序

## 3.4非移交产品

1. 可行性分析报告：说明该软件开发项目的实现在技术上、经济上和社会
2. 因素上的可行性，评述为了合理地达到开发目标可供选择的各种可能实施方案，说明并论证所选定实施方案的理由。
3. 开发进度月报：该月报系软件人员按月向管理部门提交的项目进展情况报告，报告应包括进度计划与实际执行情况的比较、阶段成果、遇到的问题和解决的办法以及下个月的打算等。
4. 项目开发总结报告：软件项目开发完成以后，应与项目实施计划对照，总结实际执行的情况，如进度、成果、资源利用、成本和投入的人力，此外，还需对开发工作做出评价，总结出经验和教训。
5. 软件问题报告：指出软件问题的登记情况，如日期、发现人、状态、问题所属模块等，为软件修改提供准备文档。
6. 软件修改报告：软件产品投入运行以后，发现了需对其进行修正、更改等问题，应将存在的问题、修改的考虑以及修改的影响做出详细的描述，提交审批。
7. 源程序：软件开发过程中的全部代码以及注释。

## 3.5验收标准

最后在交付客户之前进行小组内评审，保证产品的正确性、可维护性、可扩展性、可靠性：

1. 代码编写符合HB6465标准，与文档说明保持一致，代码书写风格统一，采用标准规范；
2. 整体产品符合需求设计、项目规划、技术规范、软件架构；
3. 有完备的软件需求说明书、产品使用说明书、维护测试文档、周期记录文档；
4. 文档格式符合HB6465标准，功能符合与客户的合同要求，清晰易读，没有语病与歧义。
5. 无下列错误：由于软件缺陷造成丢失数据，不符合设计要求，响应时间太长无法接受等。
6. 服务硬件达到文档说明的要求，人员考核合格，定期上门维护

## 3.6最终交付期限

从2016年10月10日开始至2016年12月1日，完成对整个系统的可行性报告分析、需求分析说明书、开发计划说明书、系统设计书、项目测试项目总结等工作，并交付用户。交付日期为2016年12月10日，延期交付日为12月20日。

# 4 所需工作概述

## 4.1 软件需求与约束

具备一般商店系统所具有的商品分类、检索、搜索、购物车、订单、在线支付、会员系统。顾客可以通过IE或者其他浏览器浏览商品信息；可以提交订单；具有购物车功能；客户能通过商品名称来检索商品；客户可以在网上注册，成为会员；顾客能查看当前订单和以往订单；系统管理员能查找客户信息。

## 4.2项目文档编制的需求与约束

项目文档应符合软件工程的技术规范和要求，至少应包含软件开发计划、需求规格说明、结构设计说明和软件测试说明。视开发情况添加数据要求和数据库顶层设计说明、测试报告等。

## 4.3在系统生命周期中的地位

本文档主要涵盖了系统生命周期中的问题定义、可行性研究、需求分析三大阶段，并最终形成可交付的文档。

## 4.4开发环境需求与约束

### 4.4.1 硬件环境

1. 服务器：Pentium III 500以上或更高， 内存4G以上； 硬盘至少80G以上；

2. 网络适配器：10MB/100MB自适应； 一台 UPS(选配)

3. 工作站：Pentium 4以上微机；内存1G；硬盘至少80以上；网络适配器10MＢ/100MＢ自适应；

### 4.4.2软件环境

1. 服务器端软件：WinXP/Win2003/Vista/Win7/Win8/Win10或Linux系列

2. 服务器环境：Apache，php

3. 数据库管理软件：MySQL

4. 开发工具：eclipse、sublime

5. 客户服务端软件：Firefox、Chrome等浏览器

## 4.5项目进度安排及资源的需求和约束

### 4.5.1需求分析

需求分析是整个设计中重要的一环，当可行性分析完成，项目立项，确定开发角色后，从10月10日开始至10月21日，有关的设计开发人员与相关业务人员共同对业务流程、管理方式进行分析，确定用户需求，对软件必须完成的功能进行定义，在此基础上完成数据定义，建立数据字典。

### 4.5.2 系统设计

从10月22日开始至11月1日，完成对整个系统的分析设计，对概念模型，存储模式、完整性控制存取权限等进行了定义，对系统功能各模块进行了详细设计，定义了数据库总体结构，编码命名规范。

### 4.5.3 编码及测试阶段

从11月2日开始至12月1日，完成程序设计和系统测试，完成数据库建立及程序的编制调试，为了避免错误累计，采用测试驱动开发的基本模式，对每个模块都安排专人进行单独测试，系统联调及系统测试，对系统处理逻辑、例外处理能力、容错能力等大规模的测试，对发现的问题进行彻底纠正

### 4.5.4 文档、产品部署

从12月1日开始至12月5日，完成用户培训工作，编写各类文档，系统投入运行阶段。

### 4.5.5 项目总结

项目结束后用一周的时间，对项目研发部署等开发过程中的问题、经验教训总结备案，以利于项目经验的积累和开发进程的缩短。

## 4.6其他需求

本系统为独立开发，不需要外单位提供条件。

# 5软件开发计划

## 5.1软件开发过程

软件开发采用瀑布式模型，整体流程包括需求分析、可行性分析、架构设计、编码、测试、交付、验收、运行维护。

## 5.2软件开发总体计划

### 5.2.1软件开发方法

软件开发使用面向对象开发方法，整个系统采用MVC开发框架。

### 5.2.2软件产品标准

本软件的开发本着满足用户需求的原则，使用尽量较少的代码满足整体设计要求。在编码阶段要求开发人员对所有的关键代码和接口做统一标准的注释，所有对象的选定和抽象要经过所有人的商讨与同意。在测试阶段，分为模块化测试、界面UI测试和性能测试。包括登陆、注销测试，在线支付测试（涉及第三方接口），管理员界面测试等。界面UI测试是为了检测界面是否美观，整齐，便利，用户对界面是否满意。性能测试则主要针对服务器的容量和压力进行测试，避免上线后出现服务器崩溃等情况。

### 5.2.3可重用的软件产品

### 5.2.3.1吸纳可重用的软件产品

采用面向服务架构（SOA）的方法对于已有软件进行重用，将相对独立的功能或服务组装成一个内聚的应用。在吸纳过程中主要考虑的因素有：

可重用材料的正规分类及用途；

连接可重用材料的标准接口的定义；

所有可重用材料的用户信息以及使用说明；

所有可重用材料相关的测试用例以及测试脚本；

所有可重用材料的错误报告；

可重用材料的来源确认；

所有可重用材料的变更记录；

所有可重用材料的种类；

所有可重用材料的分布记录（以备召回之需）；

对非免费的可重用材料实行收费；

确保可重用材料的版权和知识产权不受侵犯；

### 5.2.3.2开发可重用的软件产品

采用基于构件的软件工程开发方法，领域工程与软件工程同时进行。领域工程创建应用领域的模型，标识，构造，分类和传播一组可重用的软件；基于构件的开发完成使用可重用构件构造新的软件工程。为此，项目采用构件库对于可重用构建进行存储和管理，以为项目提供构建的存储、管理、检索、库的浏览与维护功能，其关键在于高效而准确地发现所需的可重用构件，相关的主要技术是分类方法和检索方法。

### 5.2.4处理关键性需求

安全性保证：为保证效益和效率，系统的安全必须从需求的定义阶段开始，直至最后的接收阶段。这种方法要求全体设计人员在整个过程中能够作为一个团队通力合作。在实施和测试等阶段，使用遵循策略的自动化工具能够支持可重复的测试，并且随着测试过程的标准化可以使开发周期更快。

保密性保证：本软件涉及大量用户数据和个人隐私，对整个数据库的信息安全提出了更高的要求。设计数据库时需考虑到安全性原则，考虑在前端加入加密函数。后端数据库也应实时备份，避免因紧急断电或自然灾害引发的数据丢失。

私密性保证：在开发与维护的过程中采用硬件防火墙，软件防火墙的措施对服务器中的内容以及用户数据进行保护，防止收到攻击而导致数据丢失。

其他关键性需求保证：软件质量保证通过对软件产品和活动进行评审和审计来验证软件是合乎标准的。软件质量保证组在项目开始时就一起参与建立计划、标准和过程。这些将使软件项目满足机构方针的如下目标：

目标 1: 软件质量保证工作是有计划进行的。

目标 2: 客观地验证软件项目产品和工作是否遵循恰当的标准、步骤和需求。

目标 3: 将软件质量保证工作及结果通知给相关组别和个人。

目标 4: 高级管理层接触到在项目内部不能解决的不符合类问题。

目标 5: 软件质量需要全面的测试工作来保证。

### 5.2.5计算机硬件资源利用

采用虚拟化技术，实现资源隔离的功能。使用软件方法重新定义划分硬件资源，可以实现IT资源的动态分配，灵活调度，跨域共享，提高硬件资源利用率。虚拟出的IT资源与底层硬件相隔离，所以虚拟化技术能够实现IT资源的自动化分配、调度、共享、迁移、监控。虚拟化可以让硬件资源按需分配。其主要具有以下四方面的功能：

1. 分区：在单一物理服务器上同时运行多个虚拟机
2. 隔离：在同一服务器上的虚拟机之间相互隔离
3. 封装：整个虚拟机都保存在文件中，而且可以通过移动和复制这些文件的方式来移动和复制这些虚拟机
4. 相对于硬件独立：无需修改即可在任何服务器上运行虚拟机

### 5.2.6记录原理

系统的记录原理工作必须从需求的定义阶段开始，直至最后的接收阶段。这种方法要求全体设计人员在整个过程中能够作为一个团队通力合作。

### 5.2.7需方评审途径

1. 分层次评审

用户的需求一般而言可以分成如下的层次：

目标性需求：定义了整个系统需要达到的目标；

功能性需求：定义了整个系统必须完成的任务；

操作性需求：定义了完成每个任务的具体的人机交互；

目标性需求是企业的高层管理人员所关注的，功能性需求是企业的中层管理人员所关注的，操作性需求是企业的具体操作人员所关注的。对不同层次的需求，其描述形式是有区别的，参与评审的人员也是不同的。

2. 正式评审与非正式评审结合

正式评审是指通过开评审会的形式，组织多个专家，将需求涉及到的人员集合在一起，并定义好参与评审人员的角色和职责，对需求进行正规的会议评审。而非正式的评审并没有这种严格的组织形式，一般也不需要将人员集合在一起评审，而是通过电子邮件、文件汇签甚至是[网络](http://net.it168.com/" \t "http://blog.csdn.net/lejuo/article/details/_blank" \o "网络)聊天等多种形式对需求进行评审。两种形式各有利弊，但往往非正式的评审比正式的评审效率更高，更容易发现问题。因此在评审时，应该更灵活地利用这两种方式。

3. 分阶段评审

应该在需求形成的过程中进行分阶段的评审，而不是在需求最终形成后再进行评审。分阶段评审可以将原本需要进行的大规模评审拆分成各个小规模的评审，降低了需求返工的风险，提高了评审的质量。比如可以在形成目标性需求后进行一次评审，在形成系统的初次概要需求后进行一次评审，当对概要需求细分成几个部分，对每个部分进行各个评审，最终再对整体的需求进行评审。

4. 精心挑选评审员

需求评审可能涉及的人员包括：需方的高层管理人员、中层管理人员、具体操作人员、IT主管、采购主管；供方的市场人员、需求分析人员、设计人员、测试人员、质量保证人员、实施人员、项目经理以及第三方的领域专家等等。

5. 对评审员进行培训

在很多情况下，评审员是领域专家而不是进行评审活动的专家，他们没有掌握进行评审的方法、技巧、过程等，因此需要对评审员进行，同样对于主持评审的管理者也需要进行培训，以便于参与评审的人员能够紧紧围绕评审的目标来进行，能够控制评审活动的节奏，提高评审效率。

6. 充分利用需求评审检查单

需求检查单是很好的评审工具，需求检查单可以分成2类：需求形式的检查单和需求内容的检查单。需求形式的检查可以由QA人员负责，主要是针对需求文挡的格式是否符合质量标准来提出的，需求内容的检查是由评审员负责的，主要是检查需求内容是否达到了系统目标、是否有遗漏、是否有错误等等，这是需求评审的重点。检查单可以帮助评审员系统全面地发现需求中的问题，检查单也是随着工程财富的积累逐渐丰富和优化的。

# 6实施详细软件开发活动的计划

## 6.1 项目计划和监督

### 6.1.1 软件开发计划

项目预计于2016年10月10日开始，2016年12月5日完成所有工作，交付客户。项目预计持续54个工作日。

### 软件开发计划表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 活动阶段 | 开始日期 | 完成日期 | 所需时间（天） | 所需资源 |
| 分析调研 | 2016.10.10 | 2016.10.15 | 6 | 资金支持、问卷设计 |
| 设计实现 | 2016.10.16 | 2016.11.10 | 24 | 软硬件开发环境、人员技术基础 |
| 初步测试 | 2016.11.11 | 2016.11.20 | 10 | 待测数据库与服务器 |
| 上线宣传 | 2016.11.21 | 2016.11.27 | 7 | 资金支持，宣传文案 |
| 用户测试 | 2016.11.28 | 2016.12.5 | 8 | 用户培训、实际运行 |
| 版本更新 | 2016.12.3 | 2016.12.8 | 5 | 无 |
| 项目总结 | 2016.12.8 | 2015.12.15 | 8 | 无 |

表 6- 1 软件开发计划表

### 6.1.2 CSCI测试计划

软件配置项（CSCI）内容包括：功能测试、性能测试、外部接口测试、余量测试和边界测试，必要时，应包括人机交互界面测试、强度测试、可靠性测试、安全性测试、恢复性测试、安装性测试、互操作性测试、敏感性测试等内容。

### CSCI测试计划表

|  |  |  |  |
| --- | --- | --- | --- |
| 任务 | 开始日期 | 完成日期 | 所需时间（天） |
| 功能测试 | 2016.11.26 | 2016.11.27 | 2 |
| 性能测试 | 2016.11.28 | 2016.11.29 | 1 |
| 外部接口测试 | 2016.11.29 | 2016.11.29 | 0.5 |
| 余量和边界测试 | 2016.11.29 | 2016.11.30 | 1.5 |
| 人机交互界面测试 | 2016.11.30 | 2016.12.1 | 2 |

表 6 - 2 CSCI测试计划表

### 6.1.3 系统测试计划

### 系统测试计划表

|  |  |  |
| --- | --- | --- |
| 测试阶段 | 测试类型 | 测试目标 |
| 单元测试 | 接口测试 | 确保接口调用正确性 |
| 单元测试 | 功能测试 | 确保单元模块功能正常 |
| 集成测试 | 功能测试 | 确保模块间交互正常 |
| 系统测试 | 用户界面测试 | 确保浏览可正确反映业务的功能和需求 |
| 系统测试 | 功能测试 | 确保系统功能正常 |

表 6 - 3 软件测试计划表

### 6.1.4 软件安装计划

软件将采用集成安装包的方式为用户安装，对于部分遇到困难的用户，采取上门指导培训的方式，直到用户掌握使用技能。

### 6.1.5 软件移交计划

软件将于12月5日移交到用户手中，具体移交物包括：

1. 程序：

项目名称为“学院路”购物平台，编程语言方面，主要使用PHP，JAVASCRIPT，HTML，数据库类型MYSQL。主要功能：

用户功能：注册新用户，登录系统，浏览公告，发表留言，添加修改和删除购物车的信息，提交订单。

浏览者功能：查看网站主页，商品信息查询，浏览公告信息

购物系统管理后台：管理员注册系统，管理员登录系统，用户管理系统，订单管理系统，商品管理系统，公告管理系统。

2文档：

软件维护说明：主要包括软件系统说明、程序模块说明、操作环境、支持软件的说明、维护过程的说明，便于软件的维护。

软件操作说明：本说明为用户提供软件各种运行情况的有关知识，特别是操作方面的具体细节

开发过程中产生的其他文档不包含在移交物之列中。

### 6.1.6 跟踪和更新计划

1. 计划活动跟踪

计划活动就是有明确完成时间的任务，是项目计划的主体和项目的主要活动。项目跟踪和控制的首要任务就是确保计划好的任务按期完成，具体措施是建立一包含计划完成时间和实际完成时间的任务表。颗粒度小的任务只有两个状态：完成或者未完成。跨度长的项目我们采用任务完成百分比来跟踪。项目组计划使用Project软件进行计划活动跟踪。

1. 缺陷跟踪

软件项目在制品（如文档、代码等）交付前经过测试都会产生大量缺陷，这些缺陷需要管理，并及时修复。这需要建立项目的缺陷跟踪系统，直到需要修复的缺陷全部关闭，产品才能交付。

1. 事务跟踪

项目过程中，总会产生一些计划外的事情，这些事务可能需要大量的协调或者沟通，不解决就会拖延项目的进度，甚至演化为项目的风险。

未完成上述目标项目组需完成：

1 项目计划跟踪报告（每天或者每周）

2 缺陷报告

3事务报告

4项目周报

5里程碑报告

共5种项目状态信息报告。

前三种报告都可以在通过工具中获得。后两种需要评审或者沟通获得：项目周报主要是以周为单位评估项目的状态，如任务完成情况，出现的问题等；项目里程碑报告主要是在里程碑处，如需求规格交付；原型交付，概要设计交付，集成测试交付等。召开项目评审会议评估实际的进度、工作量、资源消耗与计划的差异，评估风险等，以及采取的行动。

## 6.2 建立软件开发环境

### 6.2.1 软件工程环境

系统开发时需配置：

Windows操作系统下的Apache Web服务器

Mysql 数据库管理系统

PHP动态网页开发语言

Git版本控制

### 6.2.2 软件测试环境

测试时需配置与开发同样的运行环境，以及专用测试工具：

开源[测试管理工具](http://baike.baidu.com/view/1648241.htm" \t "http://baike.baidu.com/view/_blank)：Bugfree、Bugzilla

开源功能[自动化测试工具](http://baike.baidu.com/view/5093984.htm" \t "http://baike.baidu.com/view/_blank)：Quality Center

### 6.2.3 软件开发库

### 本Web系统开发使用[CodeIgniter Web Framework](https://www.codeigniter.com/)库。

### 6.2.4 软件开发文档

软件开发文档通过Microsoft Office Word 2016完成，并通过git进行版本控制。

### 6.2.5 非交付软件

除3.4中所列的非交付产品外，开发过程中所产生的源代码及其注释均为非交付软件。

## 6.3 系统需求分析

### 6.3.1 用户输入分析

1. 顾客通过IE或者其他浏览器浏览商品信息
2. 用户通过商品名称检索商品
3. 用户选取商品加入购物车
4. 用户通过注册成为会员
5. 用户查看当前订单与以往订单
6. 系统管理员查找客户信息

### 6.3.2 运行概念

Web系统主要是用于对网站前台的信息管理，如文字、图片、影音、和其他日常使用文件的发布、更新、删除等操作，同时也包括会员信息、订单信息、访客信息的统计和管理。简单来说就是对网站数据库和文件的快速操作，以使得前台内容能够得到及时更新和调整。

### 6.3.3 系统需求

系统运行所需环境：

服务器：Pentium III 500以上或更高， 内存4G以上； 硬盘至少80G以上；

网络适配器：10MB/100MB自适应； 一台 UPS(选配)

工作站：Pentium 4以上微机；内存1G；硬盘至少80以上；网络适配器10MＢ/100MＢ自适应；

## 6.4 系统设计

### 6.4.1 系统级设计决策

1. 系统初始化：系统初始化时，需要在服务器上开启Apache服务器，MySql数据库服务，之后用户即可通过URL对网站进行访问。

2. 运行控制：系统可采用自动控制方式运行，管理员只需要在特殊情况下进行少量的控制。数据库管理员可以对数据库后台操作当前注册的用户权限进行手动提权或降权。网站管理员可以通过对网站后台操作暂时关闭对于一些页面的访问。

3. 运行结束：当网站访问量过大，即将达到服务器承受上限时，系统会自动识别并且关闭服务器，从而结束运行。当系统需要更新或升级时，服务器管理员也可以手动关闭服务器与MySql数据库服务。

### 6.4.2 系统级体系设计

用户输入

前端界面

浏览器交互

数据库

后端逻辑

系统管理员

操作

图 6 - 1 系统体系结构设计图

## 6.5 软件需求分析

功能需求：顾客可以通过IE或者其他浏览器浏览商品信息；可以提交订单；具有购物车功能；客户能通过商品名称来检索商品；客户可以在网上注册，成为会员；顾客能查看当前订单和以往订单；系统管理员能查找客户信息。

性能需求：系统界面设计友好，操作方便、灵活；具有高可靠性和容错能力，不允许系统在工作时间停机，不允许系统丢失商品信息，不允许在运行期间出现故障和错误；具有安全检查机制，非法用户不能使用系统，不能偷看系统信息，不能修改商品，防止随意删改；具有保密机制，各种使用者只能看到允许查看的系统信息。

设计约束：系统采用MVC模型，实现表现层和控制层的分离，提高可重用性、可移植性。

安全性约束：系统采用门户服务器通过统一的目录服务管理，保证用户认证机制的一致性；用户授权机制通过角色的定义管理实现，通过定义某些角色能进行的操作权限，和定义用户拥有的角色，限定用户的操作权限，实现对用户的授权；防止对信息的错误修改、敏感数据的浏览等。

## 6.6 软件设计

### 6.6.1 CSCI级设计决策

a.CSCI应接受用户注册、订单、退订请求并产生的相应的输出、修改数据库，包括与其他系统、HWCI, CSCI和用户的接口。

b.CSCI在接受订单请求时，一定要保证响应的有效性及速度，这里可以选择KNN最近邻算法。当接受用户注册请求时，需要保证在数据库中没有重复元组，需数据库支持相应控制。不允许用户退订未提交的订单。不允许用户重复注册。

c.用户密码采取加密盐算法存取，以增加用户信息的安全性。使用https协议进行数据传输，以提高信息的私密性。同时，在多处物理地址备份数据库，以避免地震、火灾等意外情况的发生。

d.为提高系统的灵活性，系统采用面向对象技术进行开发，以在最大程度上提高系统的可重用性。系统采取CI框架开发，使得整个体系更加易用。

### 6.6.2 CSCI级体系结构设计决策

1. 程序(模块)划分

系统采用MVC框架开发，因此从体系角度来看共有三个模块

Model（模型）是应用程序中用于处理应用程序数据逻辑的部分。  
　　通常模型对象负责在数据库中存取数据。

View（视图）是应用程序中处理数据显示的部分。  
　　通常视图是依据模型数据创建的。

Controller（控制器）是应用程序中处理用户交互的部分。  
　　通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

2. 程序(模块)层次结构关系

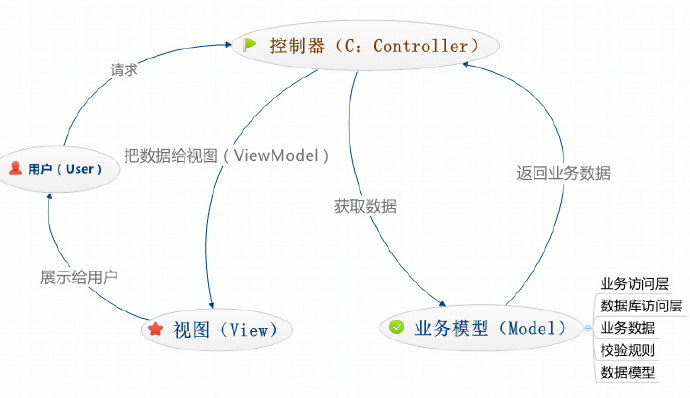


图 6 - 2 模块层次关系图

3.执行概念

作为架构模式时，View的职责就是负责展示数据，而Controller则负责获取View传递来的数据，然后调用业务逻辑层处理完成的数据传递给View进行展示。而Model则处理业务逻辑，并把结果返回给Controller。从传统三层架构上来看，View和Controller都属于UI层，而Model则横跨BLL与DAL层。

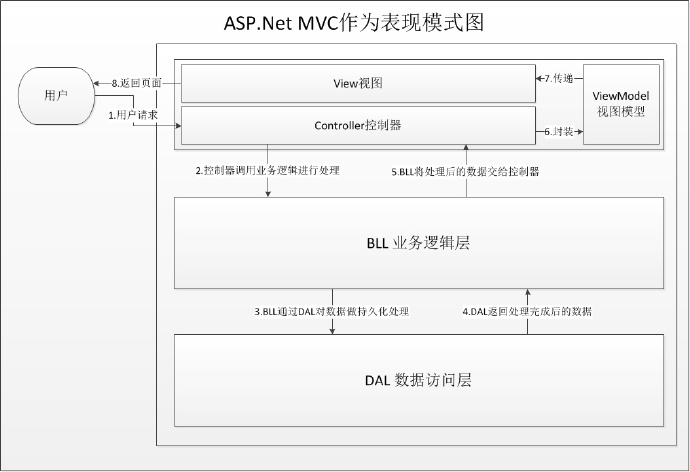


图 6 - 3 执行概念示意图

### 6.6.3 CSCI详细设计

a.设计约束：系统采用MVC模型，实现表现层和控制层的分离，提高可重用性、可移植性。

b. 安全性约束：系统采用门户服务器通过统一的目录服务管理，保证用户认证机制的一致性；用户授权机制通过角色的定义管理实现，通过定义某些角色能进

c.编程语言：由于php语言对于服务器开发的强大支持，项目组决定使用php语言进行软件设计

d.过程式命令：数据库管理系统(DBMS)中用于定义表单与报表的菜单选择、用于数据库访问与操纵的联机DBMS查询，由过程式命令组成或包含过程式命令，具体使用参见MySql官方文档。

e. 异常与错误处理：网站运行过程中的一切异常都应被记录到后台日志中，以便系统管理员进行维护。

## 6.7 软件实现和配置项设置

### 6.7.1 软件实现

软件主要有php语言，结合面向对象技术，使用MVC设计模式，采取CI框架开发而成。

### 6.7.2 配置项测试准备

在进行配置项测试前，需要确认服务器等硬件设备运行正常。

### 6.7.3 配置项测试执行

为保证最终的质量，测试应贯穿整个生命周期，在开发过程的每个阶段都保证其过程产品的质量。

尽量避免由本团队测试自己的产品，选择第三方或独立的测试团队进行测试。

测试的标准应尽量结合用户的需求。

### 6.7.4 修改和再测试

设计测试用例时，应该考虑到合法的输入和不合法的输入，以及各种边界条件，特殊情况下要制造极端状态和意外状态，比如网络异常中断、电源断电等情况。

回归测试的关联性一定要引起充分的注意，修改一个错误而引起更多错误出现的现象并不少见。

### 6.7.5 配置项测试结果分析与记录

妥善保存一切测试过程文档，测试的重现性往往要靠测试文档。

## 6.8 软件使用准备

### 6.8.1 可执行软件的准备

将可执行软件拷贝到目标机器上，还需确保目标机器具备6.3.3中陈述的运行环境

### 6.8.2 用户现场的版本说明的准备

项目采用git进行版本控制，在用户演示现场会采用最近一份成熟的版本

### 6.8.3 用户手册的准备

软件维护说明：主要包括软件系统说明、程序模块说明、操作环境、支持软件的说明、维护过程的说明，便于软件的维护。

软件操作说明：本说明为用户提供软件各种运行情况的有关知识，特别是操作方面的具体细节

## 6.9 软件质量保证

### 6.9.1 软件质量保证评估

严格按照项目开发过程中的各项步骤，从项目立项，可行性研究报告，需求分析报告、项目开发计划等，具体实施。软件质量保证小组和软件质量保证人员必须检查和督促本计划的实施。各子系统的软件质量保证人员有权直接向软件质量保证小组报告子项目的软件质量状况。各子系统的软件质量保证人员应该根据对子项目的具体要求，制订必要的规程和规定，以确保完全遵守本计划的所有要求。

### 6.9.2 软件质量保证记录

软件质量保证需涵盖项目进度，需求分析，系统设计，系统实现，系统测试，系统维护等各个方面。这些任务的参与者有两种人：软件开发者和质量保证人员。前者负责技术工作，后者负责质量保证的计划、监督、记录分析和报告工作。

### 6.9.3 软件质量保证的独立性

项目组应该避免检查自己的程序，测试工作应该由独立的专业的软件测试机构来完成，当然开发者需要在交付之前进行相关的自测。测试是带有破坏性的活动，开发人员的心理状态会影响测试的效果。同时对于需求规格说明的理解产生的错误，开发人员自己很难发现。  
　　但是，第三方或者独立的测试团队这个原则，并不是认为所有的测试完全由他们来完成。一定程度的独立测试（可以避免开发人员对自己代码的偏爱），可以更加高效的发现软件缺陷和软件存在的失效。但独立测试不是完全的替代物，因为开发人员也可以高效的在他们的代码中找出很多缺陷。在软件开发的早期，开发人员对自己的工作产品进行认真的测试，这也是开发人员的一个职责之一。

## 6.10 联合评审

### 6.10.1 联合技术评审

阶段评审工作要组织专门的评审小组，原则上由项目总体小组成员或特邀专家担任评审组长，评审小组成员应该包括项目委托单位或用户的代表、质量保证人员、软件开发单位和上级主管部门的代表，其他参加人员视评审内容而定。

每一次评审工作都应填写评审总结报告（RSR）、评审问题记录（RPL）、评审成员签字表（RMT）与软件问题报告单（SPR）等四张表格。

### 6.10.2 联合管理评审

联合管理评审主要涉及两方面工作：

1. 日常检查：在CADCSC软件的工程化生产过程中，各子系统应该填写项目进展报表，即软件进展报表表头、软件阶段进度表、软件阶段产品完成情况表、软件开发费用表等四张表格。项目总体组杨以通过项目进展季报表发现有关软件质量的问题。
2. 软件验收：必须组织专门的验收小组对CADCSC软件系统及其所属各个子系统进行验收。验收工作应按照经项目委托单位“国家自然科学基金委员会信息科学部”与CADCSC总体组双方都认可的验收规程正式履行验收手续。验收内容应包括文档验收、程序验收、演示、验收测试与测试结果评审等几项工作。具体的验收规程另行制订。

## 6.11 文档编制

基本文档为了确保软件的实现满足项目委托单位“国家自然科学基金委员会信息科学部”认可的需求规格说明书中规定的各项需求，CADCSC软件各开发单位至少应该编写以下八个方面内容的文档：a.软件需求规格说明书（SRS）；b.软件设计说明书（SDD），对一些规模较大或复杂性较高的项目，应该把本文档分成概要设计说明书（PDD）与详细设计说明书（DDD）两个文档；c.软件测试计划（STP）；d.软件测试报告（STR）；e.用户手册（SUM）；f.源程序清单（SCL）；g.项目实施计划（PIP）；h.项目开发总结（PDS）。

文档质量的度量准则文档是软件的重要组成部分，是软件生存周期各个不同阶段的产品描述。验证和确认就是要检查各阶段文档的合适性。评审文档质量的度量准则有以下六条：a.完备性、正确性、简明性、可追踪性、自说明性、规范性。

# 7进度表和活动网络图

## 7.1 活动进度表

### 7.1.1 人员分配表

|  |  |
| --- | --- |
| 任务分解 | 人员分工 |
| 市场需求分析调研 | 项目组全体人员 |
| 系统设计 | 系统架构师 |
| 系统实现 | 前端开发人员  后端开发人员 |
| 软件测试 | 测试人员主导，开发组人员参与 |
| 收集用户反馈 | 项目组长 |
| 根据反馈改进 | 开发人员 |
| 项目总结 | 项目组全体成员 |

表 7 - 1 人员分配表

### 7.1.2 活动进度表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 活动阶段 | 具体任务 | 开始时间 | 结束时间 | 时长 |
| 分析调研阶段 | 市场调研 | 2016.10.10 | 2016.10.10 | 1天 |
| 可行性分析 | 2016.10.11 | 2016.10.11 | 1天 |
| 预算估计 | 2016.10.12 | 2016.10.12 | 1天 |
| 撰写软件开发计划 | 2016.10.13 | 2016.10.14 | 1天 |
| 撰写可行性分析报告 | 2016.10.14 | 2016.10.15 | 1天 |
| 设计实现阶段 | 撰写软件需求规格说明 | 2016.10.16 | 2016.10.18 | 2天 |
| UML建模 | 2016.10.19 | 2016.10.19 | 1天 |
| 设计软件架构 | 2016.10.20 | 2016.10.21 | 2天 |
| 撰写软件结构设计说明 | 2016.10.22 | 2016.10.23 | 2天 |
| 确定功能规范 | 2016.10.24 | 2016.10.24 | 1天 |
| 定义数据库模型 | 2016.10.25 | 2016.10.26 | 2天 |
| 编写实现的技术用例 | 2016.10.27 | 2016.10.29 | 2天 |
| 实现后端数据库数据录入 | 2016.10.30 | 2016.11.2 | 3天 |
| 设计网站界面 | 2016.10.3 | 2016.11.7 | 5天 |
| UI及界面优化 | 2016.11.8 | 2016.11.10 | 3天 |
| 初步测试阶段 | 撰写软件测试说明 | 2016.11.11 | 2016.11.11 | 1天 |
| 单元测试 | 2016.11.12 | 2016.11.13 | 2天 |
| 模块集成测试 | 2016.11.14 | 2016.11.14 | 1天 |
| 整体测试 | 2016.11.15 | 2016.11.15 | 1天 |
| 自动化测试 | 2016.11.16 | 2016.11.16 | 1天 |
| 撰写测试报告 | 2016.11.17 | 2016.11.17 | 1天 |
| 修改代码以解决测试中出现的问题 | 2016.11.18 | 2016.11.19 | 2天 |
| 再次测试修改后的代码 | 2016.11.20 | 2016.11.20 | 1天 |
| 上线宣传阶段 | 网站上线运行 | 2016.11.21 | 2016.11.21 | 1天 |
| 宣传网站吸引用户使用 | 2016.11.22 | 2016.11.27 | 6天 |
| 用户测试阶段 | 集中测试 | 2016.11.28 | 2016.11.30 | 3天 |
| 收集测试中遇到的问题 | 2016.12.1 | 2016.12.2 | 2天 |
| 补充测试报告 | 2016.12.3 | 2016.12.3 | 1天 |
| 修改代码解决遇到的问题 | 2016.12.4 | 2016.12.5 | 2天 |
| 版本更新  项目总结  阶段 | 优化前一版本的算法与代码 | 2016.12.6 | 2016.12.6 | 1天 |
| 制定用户手册与使用规范 | 2016.12.7 | 2016.12.7 | 1天 |
| 审阅所有文档  总结项目 | 2016.12.8 | 2016.12.15 | 8天 |

表 7 - 2 活动进度表

## 7.2 活动网络图

详细的甘特图见附件中的mpp文件



图 7 - 1 活动网络阶段图

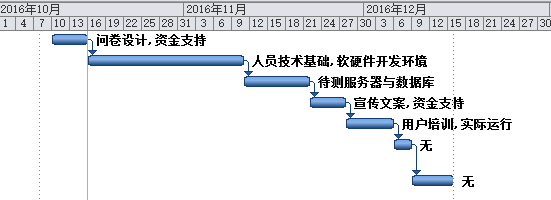


图 7 - 2 活动网络阶段图

# 8 项目组织和资源

## 8.1项目组织

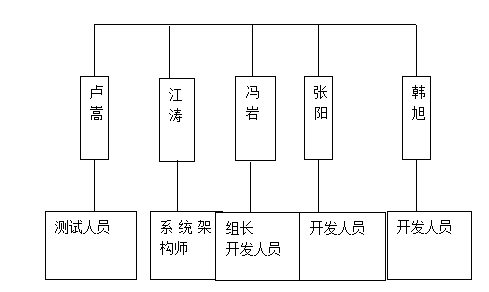


图 7 - 3 项目组织结构图

## 8.2项目资源

### 8.2.1人力资源情况

|  |  |  |  |
| --- | --- | --- | --- |
| 人力资源情况表 | | | |
| 成员 | *角色* | 掌握技术 | 开发经历 |
| 冯岩 | 项目组长 | Python，Php，Java | 网站开发，游戏开发 |
| 卢嵩 | 测试人员 | c++、c、php | 安卓应用程序开发 |
| 江涛 | 系统架构师 | C++,Java,JavaScipt | Web网页开发，安卓开发 |
| 张阳 | 开发人员 | java,c++,html | 安卓开发，Matlab |
| 韩旭 | 开发人员 | C++，Java | 网页开发 |

表 7 - 3 人力资源情况表

### 8.2.2 开发人员所需设备

|  |  |
| --- | --- |
| 开发硬件环境 | |
| 已有 | Win10 4G以上内存开发用计算机 |
| 需要 | Linux服务器 |
| 软件环境 | |
| 已有 | Windows环境，Linux环境，PHP开发环境，Web页面开发环境，Android开发环境 |
| 需要 | 无 |
| 技术力量 | |
| 已有 | 数据库技术，PHP，Git版本控制 |
| 需要 | 无 |

表 7 - 4 开发人员所需设备表

### 8.2.3 其他所需资源

无

# 9培训

## 9.1技能要求

该项目涉及前端、后端、数据库等多方面知识和技能，对编码和开发人员提出了较高的技能要求。需要开发人员有如下技能：

1. 面向对象的抽象建模能力

2. 对数据库的深入理解，熟练使用MySQL

3. 掌握PHP作为动态网页开发工具

4. 了解HTML、CSS和JavaScript等前端技术，会使用模板快速搭建前端页面

## 9.2培训计划

项目组成员已经具备所需技术

# 10项目估算

## 10.1规模估算

项目规模采用基于分解技术的估算方法，首先估算出基本量的乐观值、悲观值和一般值，然后根据经验公式：

{1E5H4506MBI)U]K2I6T7}V

得到估算结果。

采用LOC作为估算规模的指标。LOC的乐观估计值为200000行，悲观估计值为500000行，行业通常值为100000行。因此规模估算结果约为180000行。

## 10.2工作量估算

### 工作量估算表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 开支科目 | 预算/人月 | 主要用途 | 工作量分配计划/人月 | |
| 前半阶段 | 后半阶段 |
| 预算工作总量 | 10.40 | 整个系统开发费用 | 4.70 | 5.70 |
| 分析调研 | 0.10 | 问卷制作 | 0.05 | 0.05 |
| 0.10 | 问卷收发 | 0.05 | 0.05 |
| 0.20 | 数据处理 | 0.10 | 0.10 |
| 设计实现与测试 | 5.00 | 软件功能设计实现 | 3.00 | 2.00 |
| 0.75 | 软件测试审核 | 0.25 | 0.5 |
| 0.75 | 软件优化 | 0.25 | 0.5 |
| 1.50 | 软件维护 | 0.50 | 1.00 |
| 版本更新 | 1.00 | 系统更新 | 0.0 | 1.0 |
| 其他 | 1.00 |  | 0.5 | 0.5 |

表 10 - 1 工作量估算表

## 10.3成本估算

### 成本估算表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 开支科目 | 预算/千元 | 主要用途 | 阶段下达经费计划/千元 | |
| 前半阶段 | 后半阶段 |
| 预算经费总额 | 14.70 | 整个系统开发费用 | 4.00 | 10.70 |
| 硬件资源 | 2.50 | 支付服务器等硬件资源费用 | 0.50 | 2.00 |
| 市场调查 | 0.15 | 问卷制作 | 0.10 | 0.05 |
| 0.10 | 问卷收发 | 0.05 | 0.05 |
| 0.20 | 数据处理 | 0.10 | 0.10 |
| 软件开发 | 4.25 | 软件功能设计实现 | 1.25 | 3.00 |
| 1.50 | 软件测试审核 | 0.50 | 1.00 |
| 1.50 | 软件优化 | 0.50 | 1.00 |
| 2.00 | 软件维护 | 0.50 | 1.50 |
| 其他 | 1.50 | 应急费用 | 0.50 | 1.00 |

表 10 - 2 成本估算表

## 10.4关键计算机资源估算

服务器：Pentium III 500以上或更高， 内存4G以上； 硬盘至少80G以上；费用约为200元/月

网络适配器：10MB/100MB自适应； 费用约为50元/月

一台 UPS, 费用约为2000元/台

工作站：Pentium 4以上微机；内存1G；硬盘至少80以上；网络适配器10MＢ/100MＢ自适应；费用约为8000~10000元/台。

# 11风险管理

### 风险管理表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 描述 | 发生概率 | 危害程度 | 规避措施 | 缓减策略 | 状态 |
| 1 | 人员变动 | 低 | 中 | 与小组成员加强沟通，增强小组的凝聚力 | 立即吸纳新成员 | 未发生 |
| 2 | 成本估算是不准确 | 中 | 中 | 按照wbs 将工作分解，精确到天甚至工时，来估算人力成本 | 尝试向投资方继续申请投资 | 未发生 |
| 3 | 硬件资源损毁 | 低 | 高 | 在每个开发的基线或里程碑对工程进行多处备份，保证更换物理设备后项目仍能正常进行 | 将项目更换到备用设备上，回退到当前所能恢复的最后一个稳定版本进行返工 | 未发生 |
| 4 | 软件资源管理 | 低 | 中 | 在每个开发的基线或里程碑对工程在云端进行备份，已获得快速恢复能力 | 回退到当前所能恢复的最后一个稳定版本进行返工 | 未发生 |

表 11 - 1 风险管理表

# 12支持条件

## 12.1计算机系统支持

### 12.1.1 硬件环境

服务器：Pentium III 500以上或更高， 内存4G以上； 硬盘至少80G以上；

网络适配器：10MB/100MB自适应； 一台 UPS(选配)

工作站：Pentium 4以上微机；内存1G；硬盘至少80以上；网络适配器10MＢ/100MＢ自适应；

### 12.1.2软件环境

服务器端软件：WinXP/Win2003/Vista/Win7/Win8/Win10或Linux系列

服务器环境：Apache，php

数据库管理软件：MySQL

开发工具：eclipse、sublime

客户服务端软件：Firefox、Chrome等浏览器

运行时需要的支持条件：

1．服务器的中央处理部件（CPU）建议使用PIII 1G（以上） Xeon处理器芯片。

2．服务器内存必须使用服务器专用ECC内存

3．为了保证数据存储的绝对可靠，硬盘应使用磁盘冗余阵列（RAID 01）

4．为了防止服务器不可预测的故障，或者服务器的定期维护对公司整个业务造成的影响，所有建议使用两台服务器。两台服务器应构成双机热备份。中间使用Watchdog电路。这样的结构可以保证整个系统的长时间不间断工作，即使在服务器定期维护的时候也可以使用后备另一台服务器工作。

5．服务器应支持热插拔电源

6．服务器必须配备UPS（不间断电源）。

7．服务器应该放在学校内部。不然无法进行程序调试。 8．服务器应该必须有固定IP地址。

9．其他性能在经济条件允许的情况下，应该尽量使用高速稳定的配件。

## 12.2需要需方承担的工作和提供的条件

无

## 12.3需要分包商承担的工作和提供的条件

本系统为独立开发，不需要分包商承担工作