

Shopping Site

购物网站

软件开发计划书



北京航空航天大学

2016-10

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 |
| V1.0 |  |  |  | 最初版本 |
| V1.1 |  |  |  | 最终版本 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 软件开发计划书

目录

[Shopping Site I](#_Toc464198918)

[购物网站 I](#_Toc464198919)

[软件开发计划书 I](#_Toc464198920)

[软件开发计划书 2](#_Toc464198921)

[1引言 6](#_Toc464198922)

[1.1编写目的 6](#_Toc464198923)

[1.2系统概述 6](#_Toc464198924)

[2引用文件 6](#_Toc464198925)

[3 交付产品 7](#_Toc464198926)

[3.1 程序 7](#_Toc464198927)

[3.2 文档 7](#_Toc464198928)

[3.3 服务 7](#_Toc464198929)

[3.4 非移交产品 7](#_Toc464198930)

[3.5 验收标准 8](#_Toc464198931)

[3.6 项目最迟交付期限 8](#_Toc464198932)

[4所需工作概述 8](#_Toc464198933)

[4.1工作内容 8](#_Toc464198934)

[4.2 主要开发人员 8](#_Toc464198935)

[5 实施整个软件开发活动计划 9](#_Toc464198936)

[5.1软件开发过程 9](#_Toc464198937)

[5.2 软件开发总体计划 9](#_Toc464198938)

[5.2.1 软件开发方法 9](#_Toc464198939)

[5.2.2 软件产品验收标准 9](#_Toc464198940)

[5.3 软件开发预算 10](#_Toc464198941)

[FP功能点度量表 10](#_Toc464198942)

[5.4 关键问题 10](#_Toc464198943)

[6实施详细软件开发活动的计划 10](#_Toc464198944)

[6.1项目计划和监督 10](#_Toc464198945)

[6.1.1软件开发计划 10](#_Toc464198946)

[6.1.2CSCI测试计划 11](#_Toc464198947)

[6.1.3系统测试计划 11](#_Toc464198948)

[6.1.4软件安装计划 11](#_Toc464198949)

[6.1.5软件移交计划 11](#_Toc464198950)

[6.1.6跟踪和更新计划，包括评审管理的时间间隔 11](#_Toc464198951)

[6.2建立软件开发环境 11](#_Toc464198952)

[6.2.1软件工程环境 11](#_Toc464198953)

[6.2.2软件测试环境 12](#_Toc464198954)

[6.2.3软件开发库 12](#_Toc464198955)

[6.2.4软件开发文档 12](#_Toc464198956)

[6.2.5非交付软件 12](#_Toc464198957)

[6.3系统需求分析 12](#_Toc464198958)

[6.3.1用户输入分析 12](#_Toc464198959)

[6.3.2运行概念 12](#_Toc464198960)

[6.3.3系统需求 12](#_Toc464198961)

[6.4系统设计 13](#_Toc464198962)

[6.4.1系统级设计决策 13](#_Toc464198963)

[6.4.2系统体系结构设计 13](#_Toc464198964)

[6.5软件需求分析 13](#_Toc464198965)

[6.6软件设计 13](#_Toc464198966)

[6.6.1CSCI级设计决策 13](#_Toc464198967)

[6.6.2CSCI体系结构设计 13](#_Toc464198968)

[6.6.3CSCI详细设计 14](#_Toc464198969)

[6.7软件实现和配置项测试 14](#_Toc464198970)

[6.7.1软件实现 14](#_Toc464198971)

[6.7.2配置项测试准备 14](#_Toc464198972)

[6.7.3配置项测试执行 14](#_Toc464198973)

[6.7.4修改和再测试 14](#_Toc464198974)

[6.7.5配置项测试结果分析与记录 14](#_Toc464198975)

[6.8配置项集成和测试 15](#_Toc464198976)

[6.8.1配置项集成和测试准备 15](#_Toc464198977)

[6.8.2配置项集成和测试执行 15](#_Toc464198978)

[6.8.3修改和再测试 15](#_Toc464198979)

[6.8.4配置项集成和测试结果分析与记录 15](#_Toc464198980)

[6.9CSCI合格性测试 15](#_Toc464198981)

[6.9.1CSCI合格性测试的独立性 15](#_Toc464198982)

[6.9.2在目标计算机系统(或模拟的环境)上测试 15](#_Toc464198983)

[6.9.3CSCI合格性测试准备 16](#_Toc464198984)

[6.9.4CSCI合格性测试演练 16](#_Toc464198985)

[6.9.5CSCI合格性测试执行 16](#_Toc464198986)

[6.9.6修改和再测试 16](#_Toc464198987)

[6.9.7CSCI合格性测试结果分析与记录 16](#_Toc464198988)

[6.10CSCI/HWCI集成和测试 16](#_Toc464198989)

[6.10.1CSCI/HWCI集成和测试准备 16](#_Toc464198990)

[6.10.2CSCI/HWCI集成和测试执行 17](#_Toc464198991)

[6.10.3修改和再测试 17](#_Toc464198992)

[6.10.4CSCI/HWCI集成和测试结果分析与记录 17](#_Toc464198993)

[6.11系统合格性测试 17](#_Toc464198994)

[6.11.1系统合格性测试的独立性 17](#_Toc464198995)

[6.11.2在目标计算机系统(或模拟的环境)上测试 17](#_Toc464198996)

[6.11.3系统合格性测试准备 17](#_Toc464198997)

[6.11.4系统合格性测试执行 18](#_Toc464198998)

[6.12软件使用准备 19](#_Toc464198999)

[6.12.1可执行软件的准备 19](#_Toc464199000)

[6.12.2用户现场的版本说明的准备 19](#_Toc464199001)

[6.12.2用户手册的准备 19](#_Toc464199002)

[6.13软件移交准备 20](#_Toc464199003)

[6.13.1可执行软件的准备 20](#_Toc464199004)

[6.13.2源文件准备 20](#_Toc464199005)

[6.13.3支持现场的版本说明的准备 20](#_Toc464199006)

[6.13.4支持手册准备 20](#_Toc464199007)

[6.14软件配置管理 21](#_Toc464199008)

[6.14.1配置标识 21](#_Toc464199009)

[6.14.2配置控制 21](#_Toc464199010)

[6.14.3配置状态统计 21](#_Toc464199011)

[6.14.4配置审核 21](#_Toc464199012)

[6.15件质量保证 22](#_Toc464199013)

[6.15.1软件质量保证评估 22](#_Toc464199014)

[6.16文档编制 22](#_Toc464199015)

[6.17其他软件开发活动 23](#_Toc464199016)

[6.17.1风险管理，包括已知的风险和相应的对策 23](#_Toc464199017)

[6.17.1.1计划编制风险 23](#_Toc464199018)

[6.17.1.2组织和管理风险 23](#_Toc464199019)

[6.17.1.3人员风险 23](#_Toc464199020)

[6.17.1.4客户风险 24](#_Toc464199021)

[6.17.1.5过程风险 24](#_Toc464199022)

[6.17.1.6产品风险 24](#_Toc464199023)

[6.17.1.7需求风险 25](#_Toc464199024)

[6.17.1.8风险控制 25](#_Toc464199025)

[6.17.2保密性和私密性 26](#_Toc464199026)

[7 进度表和活动网络图 1](#_Toc464199027)

[7.1 进度表 1](#_Toc464199028)

[7.2 活动网络图 2](#_Toc464199029)

[8 项目组织和资源 2](#_Toc464199030)

[8.1 项目组织 2](#_Toc464199031)

[8.2项目资源 2](#_Toc464199032)

[8.2.1 人力资源 2](#_Toc464199033)

[8.2.2 其他资源 3](#_Toc464199034)

[9 培训 3](#_Toc464199035)

[9.1 项目的技术要求 3](#_Toc464199036)

[9.2 培训计划 3](#_Toc464199037)

[10 项目估算 3](#_Toc464199038)

[10.1 规模估算 3](#_Toc464199039)

[10.2 工作量估算 5](#_Toc464199040)

[10.3 成本估算 5](#_Toc464199041)

[10.4关键计算机资源估算 6](#_Toc464199042)

[10.5管理预留 6](#_Toc464199043)

[11 风险管理 6](#_Toc464199044)

[11.1计划编制风险 6](#_Toc464199045)

[11.1.1可能存在的问题： 6](#_Toc464199046)

[11.1.2解决方案： 7](#_Toc464199047)

[11.2 组织和管理风险 7](#_Toc464199048)

[11.2.1可能存在的问题： 7](#_Toc464199049)

[11.2.2解决方案： 7](#_Toc464199050)

[11.3开发环境风险 7](#_Toc464199051)

[11.3.1可能存在的问题： 7](#_Toc464199052)

[11.3.2解决方案： 8](#_Toc464199053)

[11.4最终用户风险 8](#_Toc464199054)

[11.4.1 可能存在的问题： 8](#_Toc464199055)

[11.4.2解决方案： 8](#_Toc464199056)

[11.5需求风险 8](#_Toc464199057)

[11.5.1可能存在的问题： 8](#_Toc464199058)

[11.5.2解决方案： 8](#_Toc464199059)

[11.6产品风险 9](#_Toc464199060)

[11.6.1可能存在的问题： 9](#_Toc464199061)

[11.6.2解决方案： 9](#_Toc464199062)

[11.7人员风险 9](#_Toc464199063)

[11.7.1 可能存在的问题： 9](#_Toc464199064)

[11.7.2解决方案： 10](#_Toc464199065)

[11.8 设计和实现风险 10](#_Toc464199066)

[11.8.1可能存在的问题： 10](#_Toc464199067)

[11.8.2解决方法 10](#_Toc464199068)

[11.9过程风险 10](#_Toc464199069)

[11.9.1可能存在的问题： 10](#_Toc464199070)

[11.9.2解决方法： 11](#_Toc464199071)

[12 支持条件 11](#_Toc464199072)

[12.1计算机系统支持 11](#_Toc464199073)

[12.2需要需方承担的工作和提供的条件 11](#_Toc464199074)

[12.3需要分包商承担的工作和提供的条件 11](#_Toc464199075)

[13 注解 11](#_Toc464199076)

## 1引言

### 1.1编写目的

本文档适用于购物网站，即Shopping Site。此开发计划旨在对该项目的开发做出整体的定位。为本组开发人员做出一个项目开发计划流程，以便更好地一步一步的完成该项目的设计与实现。为了保证项目团队按时保质地完成项目目标，便于项目团队成员更好地了解项目情况，使项目工作开展的各个过程合理有序，因此以文件化的形式， 把对于在项目生命周期内的工作任务范围、各项工作的任务分解、项目团队组织结构、各团队成员的工作责任、团队内外沟通协作方式、开发进度、经费预算、项目 内外环境条件、风险对策等内容做出的安排以书面的方式，作为项目团队成员以及项目干系人之间的共识与约定，项目生命周期内的所有项目活动的行动基础，项目 团队开展和检查项目工作的依据。

### 1.2系统概述

软件名称：Shopping Site（购物网站），本网站维护商品目录信息，处理用户订购（退订）、查询请求。

主要功能包括在线浏览、购物车管理、提交及支付、收货确认、服务评价等，通过本系统，可以更好的有效率的完成网上购物业务。

团队A负责该系统的开发，以及之后为维护工作。

## 2引用文件

## 3 交付产品

### 3.1 程序

本系统基于web开发，最终向客户的交付产品为网页文件。

### 3.2 文档

开发完成后，向客户交付：

《产品使用说明手册》

《系统需求规格说明书》

### 3.3 服务

通过本网站完成网上购物的维护商品目录信息，处理用户订购（退订）、查询请求，包括在线浏览、购物车管理、提交及支付、收货确认、服务评价等。

### 3.4 非移交产品

* 系统源代码。
* Web服务器端程序
* 数据库配置
* 《软件开发计划书》项目的计划。
* 《软件测试计划书》用于系统测试的设计。
* 《软件测试文档》记录系统测试完成情况。
* 《系统设计说明》用于说明系统模块和结构设计，以及各个场景的操作步骤以及数据库的设计。

### 3.5 验收标准

客户根据软件需求规格说明书逐项对系统的相应功能进行检查验证，全部通过即为通过验收。

### 3.6 项目最迟交付期限

此项目完成的最迟期限是2016年12月31号。

## 4所需工作概述

### 4.1工作内容

本网站面向客户为网上购物企业。实现一个网站，为维护商品目录信息，处理用户订购（退订）、查询请求提供帮助。能够更加合理有效的更好的为顾客进行服务。基于B/S (Browser/Server)模式进行设计，包含的主要工作有：网页的设计、服务器端的设计、数据库的设计等。

### 4.2 主要开发人员

|  |  |
| --- | --- |
| 团队A成员 | 简单介绍 |
| 杨汀阳 | 计算机学院学生，技术良好 |
| 祝星馗 | 计算机学院学生，技术良好 |
| 崔煜昆 | 计算机学院学生，技术良好 |
| 李游 | 计算机学院学生，技术良好 |

图4.2-1

## 5 实施整个软件开发活动计划

### 5.1软件开发过程

在系统开发过程中，有需求分析过程，系统设计过程，系统实现过程，系统测试过程，以及系统运行及维护过程；同时还有文档的编写、打印、分发工作；用户培训工作。现将上述工作分配到明确的组内人员。

项目计划与需求分析：杨汀阳 祝星馗

系统设计：崔煜昆

系统实现：杨汀阳 祝星馗 崔煜昆 李游

系统测试：李游

### 5.2 软件开发总体计划

### 5.2.1 软件开发方法

本系统开发过程中采用面相对象的编程方法。

将软件生命周期划分为制定计划、需求分析、软件设计、程序编写、软件测试和运行维护等六个基本活动，在本次开发中，各项活动将严格按照线性方式进行，同时根据文档作为指导，正确的完成每个活动阶段要完成的工作，并且进行验收工作。最终完成系统的全部开发。

### 5.2.2 软件产品验收标准

在系统开发过程中，有如下要求：

1. 格式标准：完全按照JAVA语言编写风格进行，包括代码的缩进以及变量的大小写等要求。
2. 注释要求：对开发中涉及到的全部变量需要明确的注释，同时对于重要的、复杂的逻辑部分代码需要明确注释。

### 5.3 软件开发预算

### FP功能点度量表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测量参数 | 数量 | 简单4 | 平均8 | 复杂12 | Fp计数 |
| 外部输入和输出数 | 27 | 10 | 9 | 8 | 208 |
| 用户交互数 | 54 | 32 | 14 | 8 | 336 |
| 外部接口数 |  |  |  |  |  |
| 系统要用的文件数 | 61 | 37 | 18 | 6 | 364 |
| 总计数值 | 142 | 79 | 41 | 22 | 908 |

### 5.4 关键问题

选择JSP、JavaBean、Servlet、JavaScript等作为应用程序开发语言，采用Struts、Hibernate、Spring框架，运用MVC三层设计模式及CSS+DIV网页设计模式，运用Tomcat服务器技术，并选择Mysql作为后台的数据库，整个系统完全基于B/S (Browser/Server)模式进行设计。

## 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

计划四人组队完成购物网站的开发，采取瀑布式开发流程，以Java为主体，结合MySQL数据库的管理使用，实现购物网站的正常使用，并随着课程的正常进展进行工程的改进和完善。

### 6.1.2CSCI测试计划

通过所有队员的共同测试和引进其他同学的测试，对工程的工程功能进行测试和完善，并在课程推进的过程中对工程代码进行修改完善。

### 6.1.3系统测试计划

在大作业提交前期，实现对全部工程正常功能的复查和改进完善，并进行反复的测试确保无误。

### 6.1.4软件安装计划

使用eclipse的ee版本，MySQL55版本，在开学前期就已经能够搭建完毕，并进行些小工程的实现，以保证大作业的顺利无误。

### 6.1.5软件移交计划

队员之间的合作分工确定之后，进行队员间的交流，对于相互之间有交集的部分进行协商讨论，确保各部分融合的时候省却不必要的麻烦。并在工程完成之后提交github上进行管理和共享。

### 6.1.6跟踪和更新计划，包括评审管理的时间间隔

在工程完成之后，每周至少进行一次全面的测试，对出现的问题及时修正，十天一次进行小组集会讨论，并对新的想法进行讨论和有选择的实现。

### 6.2建立软件开发环境

建立、控制、维护软件开发环境所遵循的方法。

### 6.2.1软件工程环境

Win10操作系统。

### 6.2.2软件测试环境

Win10操作系统。

### 6.2.3软件开发库

Java自带的库和链接数据库所需jar包。

### 6.2.4软件开发文档

在着手开发前进行开发文档的撰写。

### 6.2.5非交付软件

eclipse的ee版本，MySQL55版本。

### 6.3系统需求分析

### 6.3.1用户输入分析

用户需要进行注册登录，注册名、密码需要符合相关的设计规定。对于不符合要求的输入进行提示。

### 6.3.2运行概念

满足用户在登录系统之后的订购查询等请求，具体的使用方式和京东APP类似。

### 6.3.3系统需求

需要支持订购业务流程，具体包括在线浏览、购物车管理、提交及支付、收货确认、服务评价等。

### 6.4系统设计

### 6.4.1系统级设计决策

用户通过登录系统进入到内部订购系统，在订购系统里面进行一系列的订购服务流程。

### 6.4.2系统体系结构设计

用户通过在浏览器里输入网址进入登录界面，之后再页面上输入账号密码登录进系统里，随后在系统内进行订购业务流程。

### 6.5软件需求分析

需要支持订购业务流程，具体包括在线浏览、购物车管理、提交及支付、收货确认、服务评价等。实际使用起来更京东商城类似。这就需要在设计的时候，对登录界面的图形处理和网页上信息的传递接收进行处理。

### 6.6软件设计

### 6.6.1CSCI级设计决策

对不同输入用户密码进行不同的反应处理；对不同的商品请求事件进行不同的响应处理。

### 6.6.2CSCI体系结构设计

通过不同的用户账号密码登录到不同的界面，通过对不同的商品响应到不同的响应界面上去，并由不同的请求信息得到不同的响应结果。

### 6.6.3CSCI详细设计

当用户的账号和密码匹配成功时，进行正常的登录界面响应，否则进行报错处理。当对商品进行系统允许的操作请求时，进行正常的反应回应，否则进行适当的容错处理。

## 6.7软件实现和配置项测试

### 6.7.1软件实现

软件以eclipse为开发编辑平台，通过自己搭配的MySQL的数据库系统，借助java、jsp等编程工具组队完成整个系统的正常开发。

### 6.7.2配置项测试准备

对不同的账号密码输出进行设计，和多种点击事件进行设计。

### 6.7.3配置项测试执行

按照预先设计的情况进行实际的操作，观察系统的反应情况。

### 6.7.4修改和再测试

对于违反设计需求的情况进行更改，同时还在有必要的情况下进行些美化设计。

### 6.7.5配置项测试结果分析与记录

对于各种违反设计者最初意愿的结果进行记录和说明，并进行原因的分析。并对正常的反应现象进行标注说明即可。

### 6.8配置项集成和测试

### 6.8.1配置项集成和测试准备

对每项事务和属性都进行多状况的考虑和分析，并进行可能的违反设计原则的设计考虑。

### 6.8.2配置项集成和测试执行

按照预先设计的情况进行实际的操作，观察系统的反应情况。

### 6.8.3修改和再测试

对于违反设计需求的情况进行更改，同时还在有必要的情况下进行些美化设计。

### 6.8.4配置项集成和测试结果分析与记录

对于各种违反设计者最初意愿的结果进行记录和说明，并进行原因的分析。并对正常的反应现象进行标注说明即可。

## 6.9CSCI合格性测试

### 6.9.1CSCI合格性测试的独立性

在不受其他条件的干扰下进行的正常的测试环境下，完成对目标系统的合格性检测。在每个类的物体属性分析的过程中并不参考其他类型相近的物体的状况。

### 6.9.2在目标计算机系统(或模拟的环境)上测试

在目标计算机里，在正常的通用系统内进行CSCI合格性测试。

### 6.9.3CSCI合格性测试准备

首先调查相关物体的各项属性，进行多状况的考虑和分析，并进行可能的违反设计原则的设计考虑。

### 6.9.4CSCI合格性测试演练

随意抽选一类物体进行测试，熟悉正常的测试流程。

### 6.9.5CSCI合格性测试执行

按照预先设计的情况进行实际的操作，观察系统的反应情况，将正常的情况与实际情况对比，留心反常状况。

### 6.9.6修改和再测试

对于不合格的设计的情况进行更改，同时还在有必要的情况下进行些美化设计。

### 6.9.7CSCI合格性测试结果分析与记录

对于各种违反设计者最初意愿的结果进行记录和说明，并进行原因的分析。并对正常的反应现象进行标注说明即可。

### 6.10CSCI/HWCI集成和测试

### 6.10.1CSCI/HWCI集成和测试准备

首先调查相关物体的各项属性，进行多状况的考虑和分析，并进行可能的违反设计原则的设计考虑。

### 6.10.2CSCI/HWCI集成和测试执行

按照预先设计的情况进行实际的操作，观察系统的反应情况，将正常的情况与实际情况对比，留心反常状况。

### 6.10.3修改和再测试

对于不合格的设计的情况进行更改，同时还在有必要的情况下进行些美化设计。

### 6.10.4CSCI/HWCI集成和测试结果分析与记录

对于各种违反设计者最初意愿的结果进行记录和说明，并进行原因的分析。并对正常的反应现象进行标注说明即可。

### 6.11系统合格性测试

### 6.11.1系统合格性测试的独立性

该测试仅针对开发的系统进行，只要满足系统运行的环境，系统的测试不会因为电脑的不同，测试人员的不同等与该系统不相关的因素影响。

### 6.11.2在目标计算机系统(或模拟的环境)上测试

系统测试包括系统的易用性、可靠性、安全性、可维护性进行测试，整个系统集成后提供服务的能力，还包括系统服务性能测试、疲劳测试。此次测试在组员计算机上进行。

### 6.11.3系统合格性测试准备

测试准备主要是环境的配置与相关软件的准备。本系统利用JAVA语言实现，同时选用MySql数据库作为系统后台数据库。为了更好的完成开发工作，推荐使用运行windows操作系统的PC，同时要在开发机器上配置好环境变量。配置环境变量步骤，以windows xp系统为例：

1.安装JDK，安装过程中可以自定义安装目录等信息，例如我们选择安装目录为D:/java/jdk1.5.0\_08；

2.安装完成后，右击“我的电脑”，点击“属性”，选择“高级”选项卡，点击“环境变量”；

3．在“系统变量”中，设置3项属性，JAVA\_HOME,PATH,CLASSPATH(大小写无所谓),若已存在则点击“编辑”，不存在则点击“新建”；

4.JAVA\_HOME指明JDK安装路径，就是刚才安装时所选择的路径D:/java/jdk1.5.0\_08，此路径下包括lib，bin，jre等文件夹（此变量最好设置，因为以后运行tomcat，eclipse等都需要依\*此变量）；Path使得系统可以在任何路径下识别java命令，设为：%JAVA\_HOME%/bin;%JAVA\_HOME%/jre/bin CLASSPATH为java加载类(class or lib)路径，只有类在classpath中，java命令才能识别，设为：;%JAVA\_HOME%/lib/dt.jar;%JAVA\_HOME%/lib/tools.jar (要加.表示当前路径)

5.开始”－>;“运行”，键入“cmd”，键入命令“java -version”，“java”，“javac”几个命令，出现画面，说明环境变量配置成功；

6.同时还需要安装tomcat和编辑java的IDE。

### 6.11.4系统合格性测试执行

1. 可靠性

由于小组成员水平有限所以不可避免会有设计错误，缺乏对特殊情况和错误处理的考虑等。编码错误，变量初始化错误等。众所周知程序代码越长，结构越复杂，其可靠性越难保证。我们会在软件开发的各个阶段进行测试，尽量减少错误，同时进行代码优化，提高系统可靠性。

2. 易用性

购物网站系统是直接面对广大普通消费者的，而这些消费者对计算机并不一定是非常熟悉。这就要求系统能够提供良好的用户接口，易用的人机交互界面，从而缩短用户对系统熟悉的过程。使得该系统更能面向广大的群众来进行更良好的服务。经过对于容量为100的样本进行调查后，大家对于该系统的使用表示并不困难。

3. 安全性

本系统涉及到密码登陆，在前端与后台传输过程中有可能泄露用户密码等信息，所以在传输过程中，用户名和密码使用密文传输。经过测试，用户的信息没办法通过DBMS以外的方式进行查询，保证了用户的信息安全。

4. 可维护性

该购物网站系统在开发过程中，应该充分考虑以后的可维护性。因为对于用户查询的需求也会不断的更新和完善，一些新的功能也可以逐渐嵌入。所以，要求系统提供足够的手段进行功能的调整和扩充。我们开发采用的分功能块开发，使系统具有良好的维护性，可以很好的对该系统进行维护。

### 6.12软件使用准备

### 6.12.1可执行软件的准备

本工具部署在服务器上可直接通过网址访问。方便快捷，除浏览器外，用户不需要其他工具对本软件进行使用。

### 6.12.2用户现场的版本说明的准备

用户现场的版本属于Demo（演示版），仅集成了正式版中的几个功能，提供演示的作用，并不是最终的发布版。正式版拥有更丰富的功能。具体展示商品浏览及在线支付的功能。

### 6.12.2用户手册的准备

用户手册是详细描述软件的功能、性能和用户界面，使用户了解到如何使用该软件。主要包括(1)引言(2)软件概述(3)运行环境(4)使用说明(5)运行说明(6)非常规过程(7)操作命令一览表(8)程序文件(或命令文件)和数据文件一览表(9)用户操作举例。

### 6.13软件移交准备

### 6.13.1可执行软件的准备

本工具部署在服务器上可直接通过网址访问。方便快捷，除浏览器外，用户不需要其他工具对本软件进行使用。

### 6.13.2源文件准备

Web形式的工具软件不提供源文件，只提供访问的网址。

### 6.13.3支持现场的版本说明的准备

现场版本即为最终发布版的Full version，完成了计划中说明的所有功能。

最终版的功能主要有：

1.在线浏览商品以及商品的搜索。

2.商品提交及支付。

3.购物车商品的增删和结算。

4.已购商品的物流查询。

5.收货确认。

6.售后功能。

### 6.13.4支持手册准备

支持手册包括软件的简介，软件支持的服务方式，软件bug的反馈方式等。

计划提供以下服务：

课堂培训：以讲课形式对客户进行培训，使其能够正确使用软件。

免费咨询：客户可以在工作时间向技术人员提出问题并获得解答。

技术支持：对于某些客户，采取上门指导的方式。

软件维护：获取软件使用中的问题，提供补丁程序。

升级通知：在软件新版本发布的时候，及时通知(Email或邮寄信件)注册的用户，并提供试用版本。

软件升级：对于注册用户，只需较少的费用即可升级到新的版本。

### 6.14软件配置管理

本产品使用git进行管理

### 6.14.1配置标识

标识软件系统的结构，标识独立部件（工作产品），并使它们是可访问的。配置标识的目的，是在整个生命周期中标识系统各部件并提供对软件过程及其软件产品的跟踪能力。即：怎么命名？版本如何设置？放到哪里？哪些是受控的？受控的级别是什么？读写的权限是什么？

### 6.14.2配置控制

软件生命周期中控制软件产品的发布和变更，目的是建立确保软件产品质量的机制。即怎么变更？谁控制变更？谁来分析变更的影响范围？变更后如何验证、入库以及恢复？

### 6.14.3配置状态统计

记录和报告变更过程，目标是不间断记录所有基线项的状态和历史，并进行维护。每次基线的生成和变更都能让相关者知道变了什么？为什么变？变化前后的状态是什么？

### 6.14.4配置审核

验证软件产品的构造是否符合需求、标准、或合同的要求，目的是根据配置管理的过程和程序，验证所有的软件产品已经产生并有正确标识和描述，所有阶段的工作产品都一致并满足系统的需求，并且所有的变更需求都已解决。

### 6.15件质量保证

### 6.15.1软件质量保证评估

该软件需求覆盖率大于95%，无严重bug和遗留问题。最大并发用户数为2000，页面响应时间小于3s，与各种操作系统兼容，代码编写内容完全按照编写规范进行，代码注释量大于1：5。

### 6.16文档编制

我们为该软件编写了一系列文档，这些文档都满足以下要求：

1.针对性：文档编制以前应分清读者对象。按不同的类型、不同层次的读者，决定怎样适应他们的需要。例如，管理文档主要是面向管理人员的，用户文档主要是面向用户的，这两类文档不应像开发文档(面向开发人员)那样过多使用软件的专用术语。

2.精确性：文档的行文应当十分确切，不能出现多义性的描述。同一课题几个文档的内容应当是协调一致，没有矛盾的。

3.清晰性：文档编写应力求简明，如有可能，配以适当的图表，以增强其清晰性。

4.完整性：任何一个文档都应当是完整的、独立的，它应自成体系。例如，前言部分应做一般性介绍，正文给出中心内容，必要时还有附录，列出参考资料等。

5.同一课题的几个文档之间可能有些部分内容相同，这种重复是必要的。不要在文档中出现转引其他文档内容的情况。

6.灵活性：各个不同软件项目，其规模和复杂程度有着许多实际差别，能一律看待。

## 6.17其他软件开发活动

### 6.17.1风险管理，包括已知的风险和相应的对策

### 6.17.1.1**计划编制风险**

① 计划、资源和产品定义全凭客户或上层领导口头指令,并且不完全一致;

②计划是优化的,是"最佳状态",但计划不现实,只能算是"期望状态";

③计划基于使用特定的小组成员,而那个特定的小组成员其实指望不上;

④产品规模(代码行数、功能点、与前一产品规模的百分比)比估计的要大;

⑤完成目标日期提前,但没有相应地调整产品范围或可用资源;

⑥涉足不熟悉的产品领域,花费在设计和实现上的时间比预期的要多。

### 6.17.1.2组织和管理风险

①仅由管理层或市场人员进行技术决策,导致计划进度缓慢,计划时间延长;

②低效的项目组结构降低生产率;

③管理层审查 决策的周期比预期的时间长;

④预算削减,打乱项目计划;

⑤管理层作出了打击项目组织积极性的决定;

⑥缺乏必要的规范,导至工作失误与重复工作;

⑦非技术的第三方的工作(预算批准、设备采购批准、法律方面的审查、安全保证等)时间比预期的延长。

### 6.17.1.3人员风险

①作为先决条件的任务(如培训及其他项目)不能按时完成;

②开发人员和管理层之间关系不佳,导致决策缓慢,影响全局;

③缺乏激励措施,士气低下,降低了生产能力;

④某些人员需要更多的时间适应还不熟悉的软件工具和环境;

⑤项目后期加入新的开发人员,需进行培训并逐渐与现有成员沟通,从而使现有成员的工作效率降低;

⑥由于项目组成员之间发生冲突,导致沟通不畅、设计欠佳、接口出现错误和额外的重复工作;

⑦不适应工作的成员没有调离项目组,影响了项目组其他成员的积极性;

⑧没有找到项目急需的具有特定技能的人。

### 6.17.1.4客户风险

①客户对于最后交付的产品不满意,要求重新设计和重做;

②客户的意见未被采纳,造成产品最终无法满足用户要求,因而必须重做;

③客户对规划、原型和规格的审核 决策周期比预期的要长;

④客户没有或不能参与规划、原型和规格阶段的审核,导致需求不稳定和产品生产周期的变更;

⑤客户答复的时间(如回答或澄清与需求相关问题的时间)比预期长;

⑥客户提供的组件质量欠佳,导致额外的测试、设计和集成工作,以及额外的客户关系管理工作。

### 6.17.1.5过程风险

①大量的纸面工作导致进程比预期的慢;

②前期的质量保证行为不真实,导致后期的重复工作;

③太不正规(缺乏对软件开发策略和标准的遵循),导致沟通不足,质量欠佳,甚至需重新开发;

④过于正规(教条地坚持软件开发策略和标准),导致过多耗时于无用的工作;

⑤向管理层撰写进程报告占用开发人员的时间比预期的多;⑥风险管理粗心,导致未能发现重大的项目风险。

### 6.17.1.6产品风险

①矫正质量低下的不可接受的产品,需要比预期更多的测试、设计和实现工作;

②开发额外的不需要的功能(镀金),延长了计划进度;

③严格要求与现有系统兼容,需要进行比预期更多的测试、设计和实现工作;

④要求与其他系统或不受本项目组控制的系统相连,导致无法预料的设计、实现和测试工作;

⑤在不熟悉或未经检验的软件和硬件环境中运行所产生的未预料到的问题;

⑥开发一种全新的模块将比预期花费更长的时间;⑦依赖正在开发中的技术将延长计划进度。

### 6.17.1.7需求风险

①需求已经成为项目基准,但需求还在继续变化;

②需求定义欠佳,而进一步的定义会扩展项目范畴;

③添加额外的需求;

④产品定义含混的部分比预期需要更多的时间;

⑤在做需求中客户参与不够;

⑥缺少有效的需求变化管理过程。

### 6.17.1.8风险控制

(1)建立有效的风险控制的组织机构项目管理者联盟文章

①设置风险管理岗位：在软件开发项目管理过程中设置风险管理岗位，该岗位的主要职责是在制订与评估规划时，从风险管理的角度对项目规划或计划进行审核并发表意见，不断寻找可能出现的任何意外情况，试着指出各个风险的管理策略及常用的管理方法，以随时处理出现的风险，风险管理者最好是由项目主管以外的人担任。风险管理岗位的人数依据项目大小来决定，一般2—3人较为适合。

 ②双项目经理：为项目开发项目设定两个项目经理岗位，一个负责技术岗位，另一个负责管理岗位。目前，国内的软件开发企业的项目经理一般都是一名，而且是技术出生的占绝对多数，他们主要擅长的是技术研发，在管理方面先天不足，这不利于项目风险管理和控制。通过增加专门的管理经理岗位，可以弥补技术出生的项目经理的不足，提升软件开发项目的管理水平。而且这样的经验也已得到了国外业界大多企业的认可。

  (2) 建立有效的风险控制管理过程

风险管理过程包括培训，风险识别、风险分析、风险计划、执行计划、跟踪计划等活动，有效的风险管理过程应是学习型的、持续的和不断改进的。软件企业应建立自己的风险管理数据库作为风险管理的基础，并在实施中不断地更新和完善。

根据企业和项目的实际情况，进行科学的项目风险和控制，对项目的成功研发有着举足轻重的意义。在项目开发的过程中，进行必要的项目风险分析，制定符合项目特点的风险评估和监督机制，特别是要定期对项目的风险状况进行评估和监管，发现意外风险或者是风险超出预期的一定要重点关照。发现问题要立即上报，尽快解决。并建立风险监管日志，实行“岗位负责制”，将软件开发项目的风险降到最低。

### 6.17.2保密性和私密性

保密性：我们的软件仅用于简单的展示与使用。工作在不同环境的软件对其安全、保密的要求显然是不同的。我们所做的购物网站系统对于保密性没有太大的需求，因此不存在保密性的问题。

私密性：我们对于不同的用户设置了用户名检测，防止产生用户名相同的用户，这样每个用户都有自己用户名对应的密码，用户只能管理个人信息，无法对他人的账号进行操作。

## 7 进度表和活动网络图

### **7.1 进度表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 任务开始时间 | 任务结束时间 | 工作任务 | 预计工作量一般用人月表示工作量 | 任务责任人 |
|  |  | 完成项目开发计划文档 | 项目计划文档内容一万字 |  |
|  |  | 完成项目需求规格说明书 | 文档内容一万字 |  |
|  |  | 完成项目计划书 | 文档内容一万字 |  |
|  |  | 完成项目测试计划书 | 文档内容一万字 |  |
|  |  | 完成系统数据库设计以及相应操作函数 | 预计代码行数为1300行 |  |
|  |  | 完成管理员身份的全部功能（部分界面可以复用） | 预计代码行数为1500行 |  |
|  |  | 完成其余部分编码工作 | 预计代码行数为1600行 |  |
|  |  | 完成系统测试工作及测试文档 |  |  |

图7.1-1

### **7.2 活动网络图**

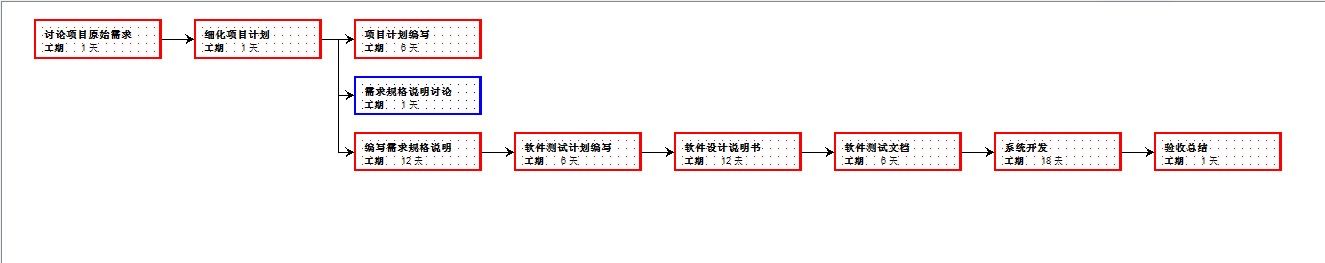


图7.2-2

## 8 项目组织和资源

### 8.1 项目组织

项目组长：杨汀阳

项目计划与需求分析：杨汀阳 祝星馗

系统设计：崔煜昆

系统实现：杨汀阳 祝星馗 崔煜昆 李游

系统测试：李游

### 8.2项目资源

### 8.2.1 人力资源

|  |  |  |  |
| --- | --- | --- | --- |
| **姓 名** | **性别** | **技 能** | **在本项目中承担的工作** |
| 杨汀阳 | 男 | Java,mysql,office | 项目计划与需求与分析，系统实现 |
| 祝星馗 | 男 | Java,mysql,office | 项目计划与需求与分析，系统实现 |
| 崔煜昆 | 男 | Java,mysql,office | 系统设计，系统实现 |
| 李游 | 男 | java,office | 系统实现，系统测试 |

图8.2.1-1

### 8.2.2 其他资源

计算机，服务器，网络宽带。

## 9 培训

### **9.1 项目的技术要求**

java,mysql,office

### **9.2 培训计划**

视具体情况而定。

## 10 项目估算

### **10.1 规模估算**

根据项目特点我准备用功能点估算方法来进行做出规模估算，首先计算未调整功能点数；然后计算调整后的功能点数作为软件规模估算的功能点数；最后根据功能点与源代码行之间的关系可将软件规模用源代码行表示。

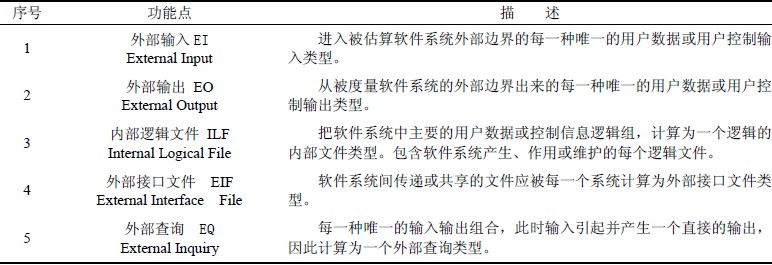


图10.1-1

未经校正的功能点数:



图10.1-2

FP=UFC\*TCF

UFC：未调整功能点计数

TCF：技术复杂度因子= 0.65+0.01

UFC=4\*4+2\*6+5\*2+4\*3=50

TCF=0.65+0.01(6\*3)=0.83

FP=50\*0.83=41.5

估算结果：

所以总的工作量估计值为：LOC=41.5\*53=2199.5

注：

1.不同编程语言下FP 与LOC 间换算关系

编程语言 LOC/FP

Java 53

Visual C++ 34

Visual Basic 29

PowerBulider 16

Ada95 49

Unix Shell Scripts 107

Lisp 64

4GL 20

Prolog 64

2. 技术复杂因子的组成

序号 名称 序号 名称

1 可靠的备份和恢复 8 联机更新主文件

2 数据通信 9 复杂的输入输出

3 分布式处理 10 复杂的内部处理

4 系统的重要性 11 代码的可重用性

5 稳定实用的操作环境 12 数据的转换与安装

6 联机数据处理 13 完善的功能和性能

7 多重屏幕和多重操作 14 易于修改和维护

影响程度有六个等级对应5个权值0--5

### 10.2 工作量估算

工作量与项目规模有关，求出了项目规模根据人员数量及相应工作分配基本上每个人工作量大致是400行代码和一篇文档

### 10.3 成本估算

成本估算主要考虑三种模型:类比法,自下而上法,参数法。我们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=（3000+2200\*4+1500）/6=2216

由于是作业（通过学习来增长知识）无人力成本，假设1行代码1元则直接成本为2216元

### 10.4关键计算机资源估算

需要一个服务器，网络宽带，计算机。

### 10.5管理预留

把工程源代码在另一台机器上进行备份，以及相关的数据资源也都进行备份。在网站工作的期间都进行保证有备份的设施以免出现空档。

## 11 风险管理

### **11.1计划编制风险**

### 11.1.1可能存在的问题：

计划、资源和产品的定义完全由客户或上层领导决定，忽略了软件项目组整体意见，并且这些决定不完全一致。 计划忽略了必要的任务和活动。致使 计划不切实际。计划基于特定的项目组人员、需要特定的技能，但是在有限的时间，精力下，难以得到这样的项目组人员或者无法完成特定技术的学习。 软件规模估算过于乐观。 工作量估算过于乐观。 进度的压力造成生产率的下降。 目标日期提前，但没有相应地调整产品范围和可用资源。 一个关键任务的延迟导致其他相关任务的连锁反应。

### 11.1.2解决方案：

进行项目计划之前，充分和项目组组员进行讨论，了解各项目组员的具体情况。并且对项目进行深入的分析，客观的分析项目组存在的问题，以及可能导致的后果，提高项目规模估计的准确度。可使用项目管理工具进行管理跟踪和控制。

### **11.2 组织和管理风险**

### 11.2.1可能存在的问题：

缺乏强有力、有凝聚力的领导。 无法正确协调项目组成员之间的关系，导致软件项目小组能力下降。 削减预算打乱软件项目计划。 低效的项目组织结构降低软件开发的生产率。 管理层审查/决策的周期比预期时间长。 管理层作出了打击软件项目组积极性的决定。 计划性太差，无法适应期望的开发速度。 软件项目计划由于压力而放弃，导致开发混乱。 管理方面的英雄主义，忽视客观确切的状态报告，降低发现和改正问题的能力。

### 11.2.2解决方案：

选择合适的领导人员。项目组组长应该有统筹、协调项目组的能力，能够听取组员的意见和建议。充分、合理的分析每一个组员的具体情况，合理分工，合理计划。在出现意外突发情况的时候，组长可以及时的发现，并且找到合理的解决方案。组长在计划项目的时候应该细化项目计划，充分发挥各个项目组组员的优点。通过多种方法，提高组员的开发积极性等。

### **11.3开发环境风险**

### 11.3.1可能存在的问题：

开发设施和工具不能及时到位。开发设施和工具到位但不配套。 开发设施和工具不如期望的那样有效。 开发人员需要更换开发设施和工具。 开发设施和工具的学习期比预期要长。 开发设施和工具的选择不是基于技术需求，不能提供计划要求的功能。

### 11.3.2解决方案：

项目组进行项目开发之前应该充分了解开发项目所需要的各项软硬件需求。主要有需要的电脑，软件等。并且要充分分析采用的开发工具是否合理。

### **11.4最终用户风险**

### 11.4.1 可能存在的问题：

最终用户坚持新的需求。最终用户对交付的软件产品不满意，要求重新开发。 最终用户的意见未被采纳，造成软件产品无法满足用户要求。

### 11.4.2解决方案：

在开发项目之前，充分进行用户需求分析，开发项目应该具有前瞻性。开发人员应该站在用户的角度进行分析、设计。提高软件产品的适用性。

### **11.5需求风险**

### 11.5.1可能存在的问题：

需求已经成为软件项目基准，但仍在变化。需求定义欠佳：不清晰、不准确、不一致。  增加了额外的需求。

### 11.5.2解决方案：

在进行需求分析的时候应该充分，充分了解、分析二手书买卖交易的需求内容，类比分析相关产品的需求，力争做到分析全面、彻底。

### **11.6产品风险**

### 11.6.1可能存在的问题：

错误发生率高的模块，需要更多的时间对它进行测试和重构。  矫正质量低下的软件产品需要更多的时间对它进行测试和重构。  由于功能错误，导致需要重新进行设计和实现。 开发额外不需要的功能延长了进度。  要满足软件产品规模和速度要求，需要更多的时间。 严格要求与现有系统兼容，需要更多的时间。 要求软件重用，需要更多的时间。

### 11.6.2解决方案：

进行详细的计划，按照以后的计划进行开发、测试。不要进行不必要的开发，测试。

### **11.7人员风险**

### 11.7.1 可能存在的问题：

招聘人员所需的时间比预期要长。作为开发人员参与工作的先决条件（如培训、其它项目的完成等）不能按时完成。 开发人员与管理层关系不佳导致决策迟缓、影响全局。 项目组人员没有全身心地投入到项目中，无法达到所需的软件产品功能和性能需求。 缺乏激励措施、士气低下。 缺乏必要的规范，增加工作失误，重复工作，降低工作质量。 缺乏工作基础（语言、经验、工具等）。 项目结束前，项目组人员离开软件项目组。  由于项目组人员间的冲突，导致沟通不畅，设计欠佳，接口错误和额外重复的工作。 有问题的项目组人员没有及时调离软件项目组，影响其他人员的工作积极性。 最佳人选没有加入软件项目组，或者加入软件项目组但没有合理使用。 项目组关键人员只能兼职参与。 项目组人员的数目不足。 任务的分配和人员的技能不匹配。 人员工作的进展比预期的要慢。 项目管理人员怠工，导致计划和进度失效。 技术人员怠工导致工作遗漏、质量低下，工作需要重做。

### 11.7.2解决方案：

项目组组员之间应该相互协助，相互学习，相互激励。在开发之前，项目组员应该充分学习开发语言，开发方法等相关知识。组长应做好领导作用，充分发挥组员的优势、特点。提高整个团队的积极性，监督整个团队严格按照计划进行开发，规范化开发。减少工作的失误，提高工作质量以及工作效率。

### **11.8 设计和实现风险**

### 11.8.1可能存在的问题：

设计过于简单，考虑不仔细、不全面，导致重新设计和实现。 设计过于复杂，导致一些不必要的工作，影响效率。 设计质量低下，导致重新设计和实现。 使用不熟悉的方法，导致需要额外的培训时间。 产品使用低级语言编写，导致开发效率较低。 分别开发的模块无法有效集成，需要重新设计和实现。

### 11.8.2解决方法

开发之前要对项目进行全面考虑，撰写详细的需求说明、开发计划等文档。对项目进行分模块开发，对每个模块进行详细的分析、设计。开发前选用合适的开发方法，提高项目组的开发效率。

### **11.9过程风险**

### 11.9.1可能存在的问题：

跟踪不准确，导致无法预知项目进展是否落后于计划。 前期的质量保证行为不真实，导致后期的重复工作。 没有遵循标准，导致沟通不足，质量问题和重复工作。 风险管理粗心，导致没有发现重大的软件项目风险。

### 11.9.2解决方法：

项目在开发过程中严格按照计划行事，按标准行事。每周进行例会，总结前段时间的成果，计划下一阶段的开发。对发现的问题要及时沟通，及时解决。

## 12 支持条件

### **12.1计算机系统支持**

开发过程要求计算机有DBMS和web服务器的支持

用户使用时要求计算机安装有支持html5的浏览器

### **12.2需要需方承担的工作和提供的条件**

需求的提出和讨论以及使用的反馈

## **12.3需要分包商承担的工作和提供的条件**

无

## 13 注解

暂无