项目名称：网上书店运营系统

的设计与实现

2023/10/23

TEAM 11

版本 <1.0>

配置管理计划

修订页

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **版本** | **修订日期** | **修订内容简述** | **修订人** |
| 1 | <1.0> | 2023.10.23 | 首次编写 | 55200701姜园鹤 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1.概述 4](#_Toc149055142)

[1.1目的 4](#_Toc149055143)

[1.2适用范围 4](#_Toc149055144)

[1.3配置管理相关概念 4](#_Toc149055145)

[1.4人员和职责 5](#_Toc149055146)

[1.5变更控制委员会 (CCB) 5](#_Toc149055147)

[2.软件配置管理 6](#_Toc149055148)

[2.1组织与任务 6](#_Toc149055149)

[2.2接口与管理流程 6](#_Toc149055150)

[2.3环境和工具 7](#_Toc149055151)

[2.4产品目录结构 8](#_Toc149055152)

[3.配置管理活动 9](#_Toc149055153)

[3.1配置标识 9](#_Toc149055154)

[3.1.1 标识方法 9](#_Toc149055155)

[3.1.2 项目基线 10](#_Toc149055156)

[3.2配置和变更控制 11](#_Toc149055157)

[3.3配置状态统计 11](#_Toc149055158)

[4.里程碑 12](#_Toc149055159)

[5.培训和资源 13](#_Toc149055160)

配置管理计划

## 1.概述

#### 1.1目的

本文档目的在于对网上书店运营系统进行配置管理，指导项目配置管理作业的过程与步骤，为项目经理、配置管理员及相关人员提供日常的配置管理操作步骤，以便全面地管理、保存软件生命周期各个配置项，监控各配置项的状态，让小组所有成员能及时了解软件基线的状态和内容，从而实现对软件过程的控制，持续改进软件流程，保证软件产品质量、降低风险，实现项目规划的所有需求，同时提高开发团队的工作效率、降低软件开发成本，保证团队的有效协作。

#### 1.2适用范围

此配置管理计划的范围：包括开发中所有的相关模型，以及受到此文档影响的其他所有相关文档。本计划适用于网上书店运营系统项目的整个生命周期。

#### 1.3配置管理相关概念

1. 软件配置管理：简称SCM（Software Configuration Management），是在项目开发中，标识、控制和管理软件变更的一种管理。配置管理的使用取决于项目规模和复杂性以及风险水平。

2. 基线：基线是项目储存库中每个工件版本在特定时期的一个“快照”。

* + 它提供一个正式基准，随后的工作基于此基准，并且只有经过授权后才能变更这个基准。
  + 建立一个初始基线后，以后每次对其进行的变更都将记录为一个差值，直到建成下一个基线。
  + 参与项目的开发人员将基线所代表的各版本的目录和文件填入各自的工作区。
  + 随着工作的进展，基线将合并自从上次建立基线以来开发人员已经交付的工作。
  + 变更一旦并入基线，开发人员就采用新的基线，以与项目中的变更保持同步。
  + 调整基线将把集成工作区中的文件并入开发工作区。

3. 产品目录结构：产品目录结构为所有可具有版本号的、与产品相关的工件提供逻辑嵌套的占位符。

4. 配置项：指定为配置管理的对象且作为单个实体进行处理的硬件、软件或两者的集合。

5. 工作区：工作区是指“私有”区域，开发人员可以根据项目所采用的标准在其中实施并测试代码，而与其他开发人员相对保持隔离。配置经理需为项目的每个开发人员创建一个工作区环境。

6. 配置审计：审核配置管理库系统的结构和设施，验证软件基线库内容的完备性和正确性，验证与适用的配置管理标准和规程的符合性。

#### 1.4人员和职责

|  |  |  |
| --- | --- | --- |
| **角色** | **职责** | **责任人** |
| 项目经理 | 负责审批项目配置管理计划，支持项目配置管理人员工作，对有关软件配置管理的各项工作全面负责，对更改建议的审批和评审负责 | 孙雁南 |
| 配置经理 | 负责制定项目配置管理计划；创建和维护配置库；按照计划实施项目配置管理活动；将配置状态报告及时提交给相关人员 | 姜园鹤 |
| 开发人员 | 根据项目组织确定的配置管理计划和相关规定，按照配置管理工具的使用模型来完成开发任务 | 其他人员 |

#### 1.5变更控制委员会 (CCB)

负责指导和控制配置管理的各项具体活动的进行，为项目经理的决策提供建议。其具体工作职责如下：

1. 批准配置项的标志，以及软件基线的建立

　　2. 制定访问控制策略

　　3. 建立、更改基线的设置，审核变更申请

　　4. 根据配置管理员的报告决定相应的对策

变更控制委员会由项目经理孙雁南及系统分析员费悦组成。在审批变更时应综合考虑该需求的必要性及对项目进度的影响，确保审批通过的变更的确能够起到修复bug、改善性能或其他必要作用并且不影响项目最终进度。

## 2.软件配置管理

#### 2.1组织与任务

在网上书店运营系统开发期间，需要成立软件配置管理小组负责配置管理工作，小组由项目经理孙雁南和配置经理姜园鹤组成，软件配置管理人员应该根据项目的具体要求，制订必要的规定，以确保完全遵守本计划规定的所有要求，软件配置管理小组必须检查和督促本计划的实施。

在各个阶段，与本阶段的产品有关的全部信息在软件开发库存放，与前面各个阶段的产品有关的信息则在配置库存放。在研制与开发产品的过程中，开发者有权对本阶段的阶段产品作必要的修改，如果开发者认为有必要更新前面相关产品时，就必须通过项目的配置管理小组办理正规的审批手续。

软件开发库受开发此阶段产品的开发小组管理，而配置库由配置管理小组管理。软件经过组装与系统测试后，应该送入软件产品库，如欲对其修改，必须经软件配置管理小组研究同意，然后报项目总体组组长批准。

#### 2.2接口与管理流程

对各类接口进行严格、合理的控制，是软件配置管理中最重要的任务之一。整个软件项目及其各子系统都必须对进行严格的控制，主要的接口有如下五类：用户界面、系统内部接口、标准程序接口、设备接口和软件接口。对这五类接口中的任一类接口进行修改时，都必须办理正规的审批手续，最后要经项目总体组批准。

在项目总体组批准软件配置管理计划之后，立即成立软件配置管理小组；建立各阶段的配置基线；在本项目所属的各个子系统的研制工作的开始，就建立起各个子系统的软件开发库，并在本项目配置管理小组的计算机上建立起有关该系统及其子系统的配置库。

#### 2.3环境和工具

考虑到稳定性，易用性，购买和维护成本，本项目选择Git作为版本控制工具，依托于Github平台。项目团队中的每个人在访问仓库之前，都需要首先在自己工作电脑上安装一个Git工具，进行代码的拉取和推送。每个开发人员的新增代码经过审查后合并到主分支中。

**Git说明：**



workspace：工作区

staging area：暂存区/缓存区

local repository：版本库或本地仓库

remote repository：远程仓库

**Git命令：**

git commit 将暂存区内容添加到仓库中

git init 初始化仓库

git clone 拷贝一份远程仓库，也就是下载一个项目

git add 添加文件到暂存区

git status 查看仓库当前的状态，显示有变更的文件

git diff 比较文件的不同，即暂存区和工作区的差异

git commit 提交暂存区到本地仓库

git reset 回退版本

git rm 将文件从暂存区和工作区中删除

git mv 移动或重命名工作区文件

git checkout 分支切换

git switch （Git 2.23 版本引入） 更清晰地切换分支

git restore （Git 2.23 版本引入） 恢复或撤销文件的更改

git log 查看历史提交记录

git blame <file> 以列表形式查看指定文件的历史修改记录

git remote 远程仓库操作

git fetch 从远程获取代码库

git pull 下载远程代码并合并

git push 上传远程代码并合并

#### 2.4产品目录结构

|  |  |
| --- | --- |
| **目录** | **内容** |
| PLN（项目计划文件） | 1. 初始软件项目计划 2. 详细软件项目计划 |
| REQ（需求文件） | 1. 软件需求规范.doc 2. 补充规约.doc 3. 用户界面原型 |
| DSN（系统设计与实施文件） | 1. 设计文档.doc 2. 设计模型   3. 用例实现.doc |
| DOC（文档） | 1. 用户手册.doc |
| TST（系统测试文件） | 1. 软件测试计划.doc 2. 软件测试设计.doc 3. 测试报告.doc |
| BIN（可执行文件） | 1. 可执行文件.exe |
| SRC（源代码文件） | 1. 源代码文件 |
| SCM（配置管理文件） | 1. 配置管理计划.doc |

## 3.配置管理活动

#### 3.1配置标识

###### 3.1.1 标识方法

所有为本项目编制的文档，都应按照如下格式命名：

<SYSTEM>\_<A> \_<SUBSYSTEM>[ \_<A>]\_<R|A|B><X>.<Y>[.<Z>][.BL<#>]

**注：**

* 1. < SYSTEM> 标识系统，本项目为<网上书店运营系统>。
  2. <A> 代表由三个字母组成的首字母缩写词，用来表示系统创建中所使用的各种工件，如PLN为项目计划，REQ为需求文件，BIN为可执行文件，TST为测试脚本和结果等。
  3. <SUBSYSTEM> 标识子系统，本项目标识资料名称，如初始软件项目计划，软件需求规范，用户界面原型，用户手册等。
  4. <A> 代表由三个字母组成的首字母缩写词，用来表示在子系统创建使用的各种工件。与2)保持一致。
  5. R|A|B代表发布版、Alpha版或Beta版
  6. <X> 整数，代表主发布版本号（例如 1）
  7. <Y> 整数，代表次发布版本号
  8. <Z> 整数（可选），代表备选发布版本号（修补程序、移植程序等）
  9. BL代表基线（内部发布版本）
  10. # 整数，代表内部发布版本号

**举例：**

|  |  |
| --- | --- |
| 命名 | 含义 |
| 网上书店运营系统\_PLN\_初始软件项目计划\_B1.0 | 网上书店运营系统的项目计划中1.0Beta发布版的初始软件计划 |
| 网上书店运营系统\_ BIN\_可执行文件X\_A2.0 | 网上书店运营系统的可执行文件中2.0Alpha版的可执行文件X |

###### 3.1.2 项目基线

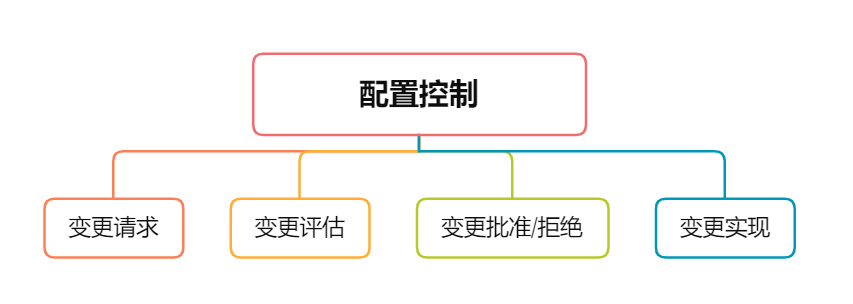
基线提供一项正式基准，随后的工作都基于此基准，并且只有经过授权后才能对此基准进行变更。所有属于本项目及其各子系统的各类基线要按照[本文档命名](#_附录1 文档命名规定)规定的命名要求来标识。

**基线列表：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **阶段** | **内容（基线产品）** | **纳入时间** | **基线建立条件** | **授权人** | **备注** |
| 计划基线 | 初始软件项目计划  软件测试计划  配置管理计划  用户界面原型 | 2023-10-5 | 评审通过 | CCB成员 |  |
| 需求基线 | 软件需求规范 | 2023-10-15 | 评审通过，并得到客户确认 | CCB成员 |  |
| 设计基线 | 设计文档  软件测试设计  详细软件项目计划 | 2023-10-25 | 评审通过 | CCB成员 |  |
| 产品基线 | 可执行文件  源代码  用户手册  测试报告  配置库 | 2023-12-6 | 产品最终完成 | CCB成员 |  |

#### 3.2配置和变更控制

基线变更应受到控制，这种变化要经过 CCB 授权，按程序进行控制并记录基线修改的过程。



**变更请求的处理和审批流程：**

1 发现问题，填写软件问题报告单

2 项目组长评审

3 软件配置管理小组评审

4 项目总体组批准，子系统负责人批准

5 修改配置并填写软件修改报告单

6 项目组长评审

7 软件质量保证小组评审，子系统质量保证人员评审

8 总体组批准项目的软件配置管理小组与子系统负责人共同批准并报项目总体组备索

#### 3.3配置状态统计

**配置审计：**

在软件开发周期各阶段的评审与检查工作中，要对该阶段所进行的配置管理工作进行必要的评审和检查。应该进行评审与检查的内容与次数，由相关项目质量计划规定。配置审计分为功能审计和物理审计。

功能审计：

1、 检查备份内容是否完整

2、 检查源代码是否通过编译，编译后程序是否能够通过冒烟测试

3、 程序功能与需求设计文档是否保持一致

4、 相关说明文档是否完整和一致

物理审计：

检查配置项的名称、配置项的版本命名、配置项的存储地址是否规范。

## 4.里程碑

|  |  |
| --- | --- |
| **里程碑** | **特点** |
| 需求分析已经确立 | 1. 系统的需求分析全部完成 2. 已经形成相应的需求分析说明书及其它文档 3. 需求分析说明书已通过评审且与客户一致认为需求分析阶段已结束，可以进入设计阶段 |
| 初步设计完成 | 1. 系统的概要初步全部完成 2. 已形成相应的设计说明书及其它文档 3. 初步设计说明书已通过评审且与客户一致认为此已结束，可以进入详细设计阶段 |
| 详细设计完成 | 1. 系统的详细设计全部完成 2. 已形成相应的详细设计说明书及其它文档 3. 详细设计说明书已通过评审或且客户一致认为详细设计阶段已结束，可以进入编码阶段 |
| 编码完成 | 1. 系统的编码全部完成 2. 系统所有程序已经经过调试并确定可以运行 3. 已通过评审且与客户一致认为编码阶段已结束，可以进入系统测试阶段 |
| 系统测试完成 | 1. 系统测试完成，发现的所有缺陷已得到处理 2. 完成测试报告 |
| 项目结束 | 1. 交付成功 2. 已得到客户的确认并通过验收测试 3. 与客户一致认为该项目已结束 |

## 5.培训和资源

配置管理和软件版本控制将使用Git，并通过GitHub完成云端同步工作，由项目经理孙雁南和配置经理姜园鹤为团队进行相关培训工作。