<“丰收宝盒”高品质农产品甄选平台>

软件开发计划

版本 <1.0>

[注：以下提供的模板用于 Rational Unified Process。其中包括用方括号括起来并以蓝色斜体（样式=InfoBlue）显示的文本，它们用于向作者提供指导，在发布此文档之前应该将其删除。按此样式输入的段落将被自动设置为普通样式（样式=Body Text）。]

[要定制 Microsoft Word 中的自动字段（选中时显示灰色背景），请选择 File>Properties，然后将 Title、Subject 和 Company 等字段替换为此文档的相应信息。关闭该对话框后，通过选择 Edit>Select All（或 Ctrl-A）并按 F9，或只是在字段上单击并按 F9，可以在整个文档中更新自动字段。对于页眉和页脚，这一操作必须单独进行。按 Alt-F9，将在显示字段名称和字段内容之间切换。有关字段处理的详细信息，请参见 Word 帮助。]

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 8/Sep/2024 | 1.0 | 初始版本 | 魏汝鑫 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 5

1.1 目的 5

1.2 范围 5

1.3 定义、首字母缩写词和缩略语 5

1.4 参考资料 5

1.5 概述 6

2. 项目概述 6

2.1 项目的目的、规模和目标 6

2.2 假设与约束 6

2.3 项目的可交付工件 6

2.4 软件开发计划的演进 6

3. 项目组织 6

3.1 组织结构 6

3.2 对外联系 6

4. 管理流程 6

4.1 项目估计 6

4.2 项目计划 7

4.2.1 阶段计划 7

4.2.2 迭代目标 7

4.2.3 发布版 7

4.2.4 项目时间表 7

4.2.5 项目资源分配 8

4.2.6 预算 8

4.3 迭代计划 9

4.4 项目监测与控制 9

4.4.1 需求管理计划 9

4.4.2 进度控制计划 9

4.4.3 预算控制计划 9

4.4.4 质量控制计划 9

4.4.5 报告计划 9

4.4.6 评测计划 9

4.5 风险管理计划 9

4.6 收尾计划 9

5. 技术流程计划 9

5.1 方法、工具和技巧 9

5.2 基础设施计划 10

6. 支持流程计划 10

6.1 配置管理计划 10

6.2 评估计划 10

6.3 文档计划 10

6.4 质量保证计划 10

6.5 问题解决计划 10

6.6 分包商管理计划 10

软件开发计划

# 简介

## 目的

根据实施“丰收宝盒”高品质农产品甄选平台所需的阶段和迭代，确定有关的开发活动是本项目计划的目的所在。

## 范围

本项目计划说明了用于开发 “丰收宝盒”高品质农产品甄选平台的总体计划。 各次迭代的细节将在迭代计划中说明。

## 定义、首字母缩写词和缩略语

无

## 参考资料

【1】软件需求规范

【2】用户界面原型

【3】详细软件项目计划

【4】软件测试计划

【5】配置管理计划

【6】设计文档

【7】软件测试设计

【8】测试报告

【9】配置库

【10】交付产品

【11】用户手册

## 概述

本软件开发计划文档详细阐述了“丰收宝盒”高品质农产品甄选平台的开发与实施策略。此文档旨在提供一个全面的指南，以指导项目团队在整个开发过程中的行动和决策。

# 项目概述

## 项目的目的、规模和目标

此项目旨在开发一个名为“丰收宝盒”的微信小程序平台，专注于提供一个高效、透明的农产品电商平台，使农民能够直接将自己的高品质农产品推广并销售给消费者。此平台将增强农民的品牌价值，改善农产品销售模式，并提升消费者的购物体验。项目规模包括一个完整的电商系统开发，包含前端用户界面和后端管理系统。

## 假设与约束

**假设**：

* 市场需求持续增长，消费者对高品质农产品的需求稳定。
* 技术解决方案可行，微信平台稳定运行。
* 农民及商家对使用数字平台持开放态度。

**约束**：

* 预算：项目预算上限为5000人民币。
* 人员：项目团队规模不超过6人。
* 设备：开发与测试需依赖现有的硬件与软件资源。
* 时间表：项目需在2个月内完成并上线。

## 项目的可交付工件

|  |  |  |
| --- | --- | --- |
| 工件名称 | 描述 | 预定交付日期 |
| 初始软件项目计划 | 对软件项目实施所涉及的各种要素进行详细的规划和安排 | 8/Sep/2024 |
| 软件需求规范 | 对软件需求进行详细描述和规范化 | 8/Sep/2024 |
| 用户界面原型 | 展示软件如何运作以及其使用体验 | 8/Sep/2024 |
| 详细软件项目计划 | 明确项目的目标、范围、时间表、资源分配、风险管理等因素，以确保项目按照预定的计划顺利进行 | 22/Sep/2024 |
| 软件测试计划 | 描述测试目的、范围、方法和软件测试的重点 | 15/Sep/2024 |
| 配置管理计划 | 阐述了如何管理项目中的配置项，包括识别、状态记录、变更控制、审核等流程 | 15/Sep/2024 |
| 设计文档 | 描述了软件系统的整体架构、‌模块设计、接口定义等详细设计信息，确保正确完成工作 | 29/Sep/2024 |
| 软件测试设计 | 将概括的测试目标转化为具体的测试条件和测试用例的一系列活动 | 13/Oct/2024 |
| 配置库 | 配置管理中用于存储和管理配置项的物理或逻辑容器 | 20/Oct/2024 |
| 交付产品 | 将产品或服务从生产或开发阶段转移到最终用户手中 | 13/Oct/2024 |
| 用户手册 | 用于介绍产品的使用方法、保养方法以及相关的售后服务信息‌ | 20/Oct/2024 |

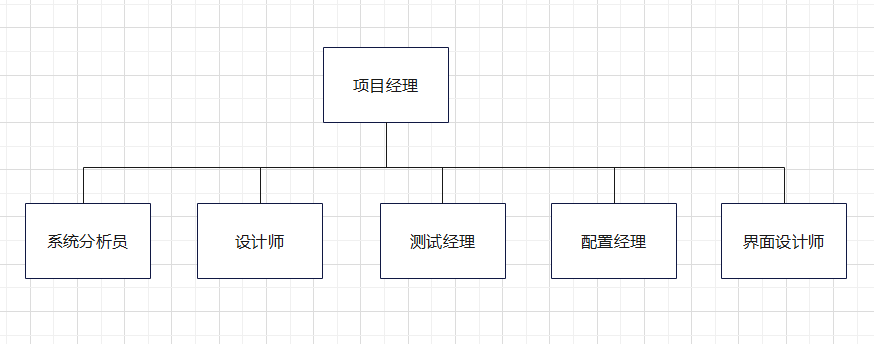
## 软件开发计划的演进

|  |  |  |
| --- | --- | --- |
| 版本 | 阶段 | 目标日期 |
| 1.0 | 先启 | 8/Sep/2024 |
| 2.0 | 精化 | 15/Sep/2024 |
| 3.1 | 构建迭代1 | 22/Sep/2024 |
| 3.2 | 构建迭代2 | 29/Sep/2024 |
| 3.3 | 构建迭代3 | 7/Oct/2024 |
| 4 | 产品化 | 13/Oct/2024 |

# 项目组织

## 组织结构

项目团队由以下部分组成：



## 外部接口

项目需与以下外部组织保持联系：

* **技术供应商**：软件开发工具和服务器托管。
  + 内部联系人：项目经理
  + 外部联系人：供应商代表
* **市场研究机构**：提供市场趋势和用户需求分析。
  + 内部联系人：市场经理
  + 外部联系人：研究员

## 角色与职责

## 下表将列出以上的项目图中的角色并说明这些角色的主要职责。

|  |  |  |
| --- | --- | --- |
| 小组成员 | 角色 | 职责 |
| 魏汝鑫 | 项目经理 | 日常项目管理；组织制定项目计划（初始软件开发计划、详细软件开发计划） |
| 侯杰 | 系统分析员 | 进行系统分析，负责收集整理项目资料进行前期调查，撰写项目可行性报告，对项目的实际需求进行细节划分和分析 |
| 陈伟铭 | 设计师 | 组织开发分析设计模型：子系统、类、接口、用例实现 |
| 吴卓宇 | 测试经理 | 组织开发测试计划、测试设计；组织执行测试；编写测试报告 |
| 李福硕 | 配置经理 | 组织开发配置管理计划；软件开发过程中的配置管理（建立基线） |
| 郗延旭 | 界面设计师 | 设计用户界面；编写用户手册 |

# 管理流程

## 项目估计

 **成本与进度估计**：

* **成本**: 预计总成本为5000人民币，包括人力资源成本、技术和设备购置费、运营和维护费用。
* **进度**: 项目预定在2个月内完成所有开发、测试和部署活动。

 **估计依据**：

* 依据前期市场调研、类似项目经验和资源利用率。

 **重新估计条件**：

* 遇到技术障碍、市场需求变更、关键资源不可用或重大偏差时需重新评估成本和时间表。

## 项目计划

### 阶段计划

**如果在一个阶段内进行了多次迭代，系统的开发就将分阶段进行。下表列出了这些阶段和相关的时间线：**

|  |  |  |  |
| --- | --- | --- | --- |
| **任务** | **工期** | **开始日期** | **结束日期** |
| **需求** | **7个工作日** | **2/Sep/2024** | **8/Sep/2024** |
| 需求分析 | 3个工作日 | 2/Sep/2024 | 4/Sep/2024 |
| 需求分析评审 | 4个工作日 | 5/Sep/2024 | 8/Sep/2024 |
| **设计** | **14个工作日** | **9/Sep/2024** | **22/Sep/2024** |
| 概要设计 | 4个工作日 | 9/Sep/2024 | 12/Sep/2024 |
| 设计原型 | 3个工作日 | 13/Sep/2024 | 15/Sep/2024 |
| 详细设计 | 3个工作日 | 16/Sep/2024 | 18/Sep/2024 |
| 制定详细软件开发计划 | 4个工作日 | 19/Sep/2024 | 22/Sep/2024 |
| **编码** | **14个工作日** | **23/Sep/2024** | **6/Oct/2024** |
| 子系统编写 | 10个工作日 | 23/Sep/2024 | 2/Oct/2024 |
| 集成 | 4个工作日 | 3/Oct/2024 | 6/Oct/2024 |
| **测试** | **7个工作日** | **7/Oct/2024** | **12/Oct/2024** |
| 制定测试计划 | 2个工作日 | 7/Oct/2024 | 8/Oct/2024 |
| 单元测试 | 2个工作日 | 9/Oct/2024 | 10/Oct/2024 |
| 整体测试 | 2个工作日 | 11/Oct/2024 | 12/Oct/2024 |
| 完成测试 | 1个工作日 | 12/Oct/2024 | 12/Oct/2024 |
| **配置** | **7个工作日** | **13/Oct/2024** | **19/Oct/2024** |
| 制定配置管理计划 | 3个工作日 | 13/Oct/2024 | 15/Oct/2024 |
| 部署 | 4个工作日 | 16/Oct/2024 | 19/Oct/2024 |

**下表列出了标志各阶段结束的里程碑。**

|  |  |
| --- | --- |
| **说明** | **里程碑** |
| 需求分析已确立 | 已形成需求规格说明书。 |
| 设计完成 | 已设计出初步的原型，和相应的用户界面说明书；  已形成详细软件开发计划；  已制定配置管理计划；  已制定软件测试测试计划； |
| 编码完成 | 所有子系统的编码全部完成  系统所有程序已经经过调试并确定可以运行  可以进入系统测试阶段 |
| 测试完成 | 系统测试完成，所发现的所有缺陷已得到妥善处理  已完成测试分析报告 |

### 迭代目标

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **阶段** | **迭代** | **说明** | **相关里程碑** | **处理的风险** |
| 先启阶段 | 初步迭代 | 确定业务模型、产品需求、项目计划和商业理由 | 商业理由复审 | 预先明确用户需求。  制订符合实际的项目计划和范围。  从商业角度确定项目的可行性 |
| 精化阶段 | 开发架构原型 | 完成所有用例的分析与设计。开发架构原型。 | 构架原型 | 明确构架方面存在的问题。  降低技术风险。  供用户复审的早期原型 |
| 构建阶段 | C1迭代-开发Beta | 实施和测试用例以提供Beta版本 | Beta | 在 Beta 版本中实现了所有从用户和构架角度提出的关键特性  用户的反馈早于软件的发布 |
|  | C2迭代-开发初始发布版 | 事实并测试其余的用例，修复Beta版中的缺陷，并根据Beta测试的反馈加以改进。 | 软件 | 用户群对软件进行全面复审  产品应该具有较高的质量  缺陷减至最少。  质量成本降低。 |
|  | C3 迭代 — 开发完整的发布版 | 并入初始发布版中的改进并纠正其中的缺陷。  开发完整的系统。 | 软件 | 快速发布版可使客户感到满意。  系统中的所有关键功能通过完整发布版来提供。 |
| 交付阶段 | 软件发布 | 包装、发行并安装发布版。 | 已发布的软件 |  |

### 发布版

此时将计划提供两个发布版。第一个发布版必须及时完成以用于 March Madness，其规模将在精化阶段确定。其余的所有功能将包括在一个后续发布版中（如果需要）。

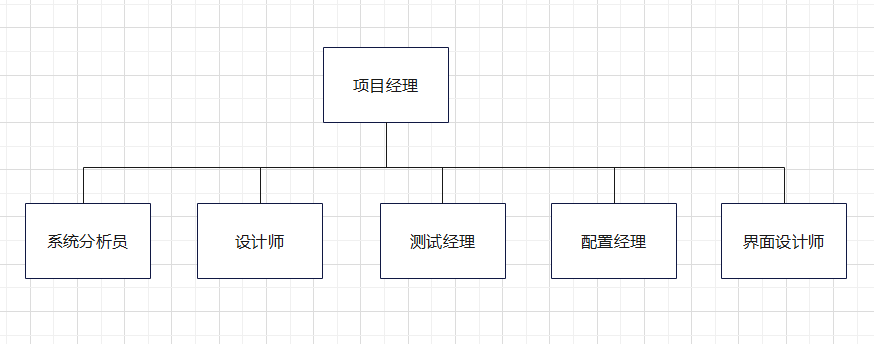
### 项目时间表

|  |  |  |  |
| --- | --- | --- | --- |
| **任务** | **工期** | **开始日期** | **结束日期** |
| **需求** | **7个工作日** | **2/Sep/2024** | **8/Sep/2024** |
| 需求分析 | 3个工作日 | 2/Sep/2024 | 4/Sep/2024 |
| 需求分析评审 | 4个工作日 | 5/Sep/2024 | 8/Sep/2024 |
| **设计** | **14个工作日** | **9/Sep/2024** | **22/Sep/2024** |
| 概要设计 | 4个工作日 | 9/Sep/2024 | 12/Sep/2024 |
| 设计原型 | 3个工作日 | 13/Sep/2024 | 15/Sep/2024 |
| 详细设计 | 3个工作日 | 16/Sep/2024 | 18/Sep/2024 |
| 制定详细软件开发计划 | 4个工作日 | 19/Sep/2024 | 22/Sep/2024 |
| **编码** | **14个工作日** | **23/Sep/2024** | **6/Oct/2024** |
| 子系统编写 | 10个工作日 | 23/Sep/2024 | 2/Oct/2024 |
| 集成 | 4个工作日 | 3/Oct/2024 | 6/Oct/2024 |
| **测试** | **7个工作日** | **7/Oct/2024** | **12/Oct/2024** |
| 制定测试计划 | 2个工作日 | 7/Oct/2024 | 8/Oct/2024 |
| 单元测试 | 2个工作日 | 9/Oct/2024 | 10/Oct/2024 |
| 整体测试 | 2个工作日 | 11/Oct/2024 | 12/Oct/2024 |
| 完成测试 | 1个工作日 | 12/Oct/2024 | 12/Oct/2024 |
| **配置** | **7个工作日** | **13/Oct/2024** | **19/Oct/2024** |
| 制定配置管理计划 | 3个工作日 | 13/Oct/2024 | 15/Oct/2024 |
| 部署 | 4个工作日 | 16/Oct/2024 | 19/Oct/2024 |

### 项目资源分配

#### 人员配备计划

含项目经理6人，分别为系统分析员、设计师、测试经理、配置经理、界面设计师。



#### 资源获取计划

学习群招募

#### 培训计划

目前，本项目所配备的人员都具有适当的技能。为了确保项目人员在产品化阶段后掌握支持系统所需的技能，将在先启阶段制定知识传授计划。

### 预算

先启阶段的预算是 1,000.00。精化阶段的预算价将在先启阶段确定。

## 迭代计划

## 项目监测与控制

### 需求管理计划

首先，通过面谈、调查和工具等方式收集需求，并使用数据流图、用例图等进行分析和建模。然 后，建立变更管理流程来控制需求的变更，并通过审查、原型测试和用户验收确保需求的正确性和 满足。还需追踪需求状态，管理需求文档的版本和存储，并对团队进行培训和沟通。同时，识别和 管理与需求相关的风险，以确保项目顺利进行。

### 进度控制计划

每周发布一次项目状态报告，报告中将包括确保项目按计划进行的里程碑跟踪明细。时间表的变更将提交给项目资助人，由他们来决定是否为满足目标完成日期而更改项目规模。

### 预算控制计划

每周发布一次项目状态报告，报告中将包括确保项目按计划进行的里程碑跟踪明细。时间表的变更将提交给项目资助人，由他们来决定是否为满足目标完成日期而更改项目规模。

### 质量控制计划

将对每个设计和实施子系统执行正式的复审。这将确保被复审的对象满足指定的需求。

### 报告计划

每周发布一次项目状态报告。在适当的时候还将发布阶段和迭代总结报告。

### 评测计划

将花费一定的精力和时间来跟踪项目进展。计划与实际情况的对比报告将由项目经理用来评测进展。

## 收尾计划

在项目结束时，将举行经验总结会议，以获取新技术、新工具或新方法。项目的可交付工件将在知识管理储存库中存档以备将来参考。

# 技术流程计划

## 方法、工具和技巧

**业务建模指南**：

使用BPMN（业务流程模型和符号）进行业务流程建模，确保所有业务逻辑清晰、准确地反映在系统功能中。

**用户界面指南**：

遵循用户中心设计（UCD）原则，优化微信小程序界面，确保易用性和访问性。

**用例建模指南**：

使用UML用例图详细描述所有用户交互，保证需求的完整性和可追溯性。

**设计指南**：

遵循清晰的MVC架构原则，确保系统设计的可维护性和可扩展性。

**编程指南**：

采用统一的编码标准，例如Google的JavaScript风格指南，以确保代码的一致性和质量。

**测试指南**：

提供详细的测试策略，包括单元测试、集成测试和性能测试，确保软件质量。

**手册风格指南**：

为所有文档设定统一的格式和风格，包括技术文档和用户手册，以提升信息的清晰度和易读性。

## 基础设施计划

使用本地服务器，Tomcat9.0