<AI\_project>

软件项目计划

**（简化版）**

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2024.6.24 | 1.0 | 初步设定了项目计划与迭代方案 | 赵楷越 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 范围 4

1.3 定义、首字母缩写词和缩略语 4

1.4 参考资料 4

2. 项目概述 4

2.1 项目的目的、规模和目标 4

2.2 假设与约束 4

2.3 项目的可交付成果 4

3. 项目组织 4

4. 项目计划 4

4.1 风险分析 4

4.2 方法和工具 5

4.3 开发计划 5

4.4 质量保证计划 5

4.5 项目沟通计划 5

4.6 培训计划 5

软件项目计划

# 简介

## 目的

本软件项目计划的目的在于规划和管理一个具体的软件开发项目，确保按照预期时间表和质量标准交付可用的软件产品。通过明确项目的范围、目标和交付成果，以及分配清晰的项目组织和角色职责，旨在成功交付满足用户需求和预期的软件产品。

## 范围

相关项目：所有doc目录下的文档（迭代计划、迭代评估报告、软件架构文档、软件项目计划、软件需求规约、系统测试报告、系统测试用例、项目总结报告）以及所有code目录下的代码

## 定义、首字母缩写词和缩略语

|  |  |
| --- | --- |
| 术语 | 解释 |
| Product Positioning | 产品定位 |
| Stakeholder | 涉众 |
| Dependency Relationship | 依赖关系 |
| Constraints | 约束 |
| Usability | 易用性 |
| Reliability | 可靠性 |
| Priority | 优先级 |
| Artificial Intelligence | 人工智能 |

## 参考资料

[1]沈备军,陈昊鹏,陈雨婷.软件工程原理[M]北京:高等教育出版社,2013.

# 项目概述

## 项目的目的、规模和目标

目的：开发一款AI秘书移动端应用程序，旨在为用户提供便捷的日常生活辅助服务。该应用程序整合人工智能技术，帮助用户管理日程安排、订购外卖、购物等多种功能，以提高生活效率和便利性。

规模：根据本项目软件开发语言和技术选择，确定了每个功能点的平均代码行数。通过历史数据，估算得到本项目代码行数在1万行左右。

目标：

1、功能完备： 提供稳定、流畅的用户界面和功能操作，确保各项服务的可靠性和效率。

2、用户体验优化： 通过用户反馈和持续优化，提升应用的易用性和用户满意度。

3、推广软件： 在应用市场中积极推广，吸引更多用户使用和推荐该应用程序。

## 假设与约束

预算：3000元华为云代金券

人员：小组成员共5人

设备：购置的华为云服务器、以及小组成员的电脑设备

时间表：2024年6月24日 至 2024年9月13日

## 项目的可交付成果

|  |  |
| --- | --- |
| 文档 | 迭代计划、迭代评估报告、软件架构文档、软件项目计划、软件需求规约、系统测试报告、系统测试用例、项目总结报告 |
| 代码 | 前端代码、后端代码 |
| 数据 | 数据库sql文件 |

# 项目组织

项目管理：赵楷越

代码控制：朱涵

界面设计：孙恬然

文档管理：齐佳怡

质量控制：骆镒妤

# 项目计划

## 风险分析

|  |  |  |  |
| --- | --- | --- | --- |
| **优先级** | **风险名称** | **风险描述** | **风险缓解措施** |
| 1 | 需求风险 | 项目的需求可能不明确或不断变化，导致开发过程中频繁调整，影响进度和质量。 | 1. 确保需求收集的全面性和准确性，明确项目范围。  2. 定期在组内沟通确认需求变化，并及时更新需求文档。实施变更控制流程，通过评审评估和批准需求更改。 |
| 2 | 技术风险 | 使用新技术或未成熟技术可能导致开发困难、性能问题或技术不可行性。 | 1. 在项目初期进行技术可行性研究和技术选型。  2. 制定备用技术方案，以备主要技术遇到问题时切换。 |
| 3 | 时间风险 | 项目时间会受到外力事务的影响，可能无法按时完成每个迭代，影响项目交付。 | 1. 制定详细的项目计划并设定合理的里程碑和时间节点。  2. 定期监控项目进展，通过敏捷方法及时调整工作计划。 |
| 4 | 管理风险 | 项目管理不善可能导致资源分配不均、团队协作问题和项目失控。 | 1. 每天早上进行立会、并定期召开项目会议，跟踪项目进展和解决问题。  2. 建立清晰的沟通渠道和责任分配机制，促进团队协作。 |

## 方法和工具

建模工具:Figma

IDE:VSC，IDEA

测试工具:JUnit，JMeter，PyTest

版本管理工具:Git

项目管理工具:CodeArts

## 开发计划

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **迭代名称** | **起止时间** | **缓解的风险** | **需完成的任务** | **需提交的成果** |
| 需求与界面原型迭代 | 6月24日 ~7月3日 | 需求风险 | 1. 确定项目基本需求、完成需求文档和用例模型 2. 根据需求文档完成界面原型的绘制 | 软件需求规约和用例模型  前端界面原型  软件架构文档 |
| 技术原型迭代 | 7月4日 ~7月14日 | 技术风险 | 1、实现前后端代码（至少包括需求中的3个最主要的用例）  2、根据需求规约设计对应的单元测试用例  3、根据所采用的技术栈、进行软件架构文档和UML分析模型的编写 | 基于UML的分析设计模型  单元测试用例  实际在本机上可执行的软件代码 |
| 部署与详细测试迭代 | 7月15日 ~7月19日 | 技术风险、质量风险 | 1、根据需求文档、编写完善的系统测试用例、测试完后撰写系统测试报告  2、将项目自动持续部署至华为云服务器上、并进行集成测试与部署测试  3、根据代码、导出单元测试报告 | 系统测试用例  系统测试报告  单元测试报告  初步上线的软件 |
| 项目完善迭代 | 7月20日 ~9月12日 | 质量风险 | 1、最终对软件项目进行需求检查  2、对项目进行可能的性能优化 | 最终部署完毕、可展示的软件 |

## 质量保证计划

需求评审：在项目开始时先进行主体需求的设想、在项目进行中实时跟进新需求的设定与评审、由小组成员共同评审。

设计评审：在界面或架构设计完后进行评审、根据需求文档一一对应、对界面或架构进行评审，由小组成员共同评审。

代码评审：合并时进行、由朱涵进行主要管理、保证代码的规范及一致性。

单元测试：代码编写完后、需要自行根据编写的单元测试用例进行测试，测试通过方可提交。

集成测试：各部分代码编写完毕后、整合时进行测试。

系统测试：项目部署后、根据系统测试用例一条一条进行测试、根据系统测试报告进行评审。

## 项目沟通计划

每天在早上9点15分进行每日立会。每日立会时检查前一天的进度、成果与问题，以及当天的计划。在必要时择时进行线下或线上会议、沟通进度和思路。

## 培训计划

对于项目中的关键实践部分（例：单元测试等），要求每个小组成员都掌握，计划由一位或几位小组成员学习之后、整理好对应资料、在固定的时间分享实践方法，达成快速学习，共同学习的目标。