**智能家庭助手**

**开发计划**

***SDP08***

**1.0**

分工说明

|  |  |  |
| --- | --- | --- |
| 小组名称 | 啊对对队 | |
| 学号 | 姓名 | 本文档中主要承担的工作内容 |
| 18375299 | 刘传 | 主要编制人 |
| 18375182 | 范竞元 | 审核人；项目任务概要讨论 |
| 18375200 | 刘裕炜 | 风险管理讨论 |
| 19373106 | 裴宝琦 | 进度计划讨论；图表绘制 |
| 18374457 | 廖纪童 | 资源计划讨论 |

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| 1.0 | 2022.3.13 | 刘传 | 范竞元 | 对项目进行了初步的规划与安排，确定了工作流程与工期 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 范围

## 项目概述

### 嵌入式系统背景

嵌入式系统是以应用为中心，以现代计算机技术为基础，能够根据用户需求(功能、可靠性、成本、体积、功耗、环境等)灵活裁剪软硬件模块的专用计算机系统。

现代嵌入式系统通常是基于微控制器（如含集成内存和/或外设接口的中央处理单元）的，但在较复杂的系统中普通微处理器（使用外部存储芯片和外设接口电路）也很常见。

随着信息和互联网的发展，智能机器人越来越多的出现在人们的日常生活中，在日常生活中扮演着十分重要的角色，而且随着科技的发展，人们不只关注机器人的工作性能，也会关注相应的能耗，希望在尽可能少的能耗下实现更高的响应速度，由于这种需求，基于嵌入式系统的智能机器人相关的研究层出不穷。

基于此，本项目将开发一种机器人，可以通过对室内环境的自动建图，实现动态避障，语音操控，同时该机器人具有很强的普适性，对于不同的环境只需要简单的设定即可实现相关功能。

### 主要功能

* 根据室内环境进行建图
* 根据指令确定目的地并进行路径规划
* 移动过程中自动躲避障碍物
* 语音操控功能
* 使用机械臂进行物品抓取

### 非功能性需求

* 操作简洁
* UI界面优美
* 提供完善的人机交互
* 语音识别效率高，准确度高
* 支持故障处理
* 整体应答速度快

### 应用场景

本家庭服务机器人，对于物品取放等家庭服务，在不同的室内环境下，都有较好的适应性，均可以完成相应的功能，适用于日常生活场景。

## 文档概述

本文档概述了项目整体的工作，让小组成员对项目有了大致的理解，方便后续开发的进行。

任务概要部分介绍了项目的工作内容和分工，介绍了小组成员，以及相关的程序，文档等，简单介绍了使用的运行与开发环境，并设定了相应的时间节点，用于把控项目进度，进行按时的推进。

风险管理部分分析了项目各方面的相关风险，并提出了对应的解决技术，用于处理开发过程中的突发情况和项目上线后的管理。

定义了本次项目开发的过程模型：迭代-增量生命周期模型，定义了整体的开发模式，让小组成员对开发过程有一个整体清晰的认识。

随后的资源计划介绍了项目所需的软硬件资源，进度计划以表格，甘特图的形式介绍了项目的重要时间节点以及整体的时间分配，通过依赖图，展示了项目各部分工作之间的拓扑关系，使得结构更加清晰。

## 术语和缩略词

本文档所涉及的专业的业务和技术术语：

|  |  |
| --- | --- |
| 术语 | 含义 |
| Ubuntu | 一种以桌面应用为主的linux操作系统的发行版 |
| C++ | 一种面向对象的计算机程序设计语言 |
| gazebo | 机器人仿真工具，模拟器，也是一个独立的开源机器人仿真平台 |

文档所涉及的缩略词及其全称：

|  |  |
| --- | --- |
| 缩略词 | 全称 |
| ROS | Robot Operating System |
| SDD | Software Design Description |
| SDP | Software Development Plan |
| SRS | Software Requirement Specification |
| STD | Software Test Description |
| STP | Software Test Procedures |
| WBS | Work Breakdown Structure |

## 引用文档

1. 邹欣,现代软件工程构建之法(第3版)[M].北京:人民邮电出版社, 2017
2. （英）伊恩·萨默维尔（Ian Sommerville）著；彭鑫，赵文耘译.计算机科学丛书 软件工程 原书第10版[M].北京：机械工业出版社,2018.02.

# 项目任务概要

## 工作内容

|  |  |  |
| --- | --- | --- |
| 一级任务 | 二级任务 | 三级任务 |
| 需求分析 | 系统需求分析 |  |
| 撰写文档 |  |
| 系统设计 | 交互界面设计 |  |
| 控制系统设计 |  |
| 系统实现 | 前端界面实现 |  |
| 后端控制实现 |  |
| 系统调试 | 仿真环境调试 |  |
| 实机环境调试 |  |

## 主要人员

主要人员

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 姓名 | 角色 | 工作时间 | 工作经验 | 技术水平 |
| 刘传 | 项目组长  程序开发  文档评审  数据库架构 | 项目全程 | 有小型个人项目及两人合作项目开发经历 | 有Go、Java、Python开发经验和技能 |
| 范竞元 | 程序开发  文档评审  网页前端设计 | 项目全程 | 有小型个人项目及两人合作项目开发经历 | 有C/C++、Java、Python开发经验和技能 |
| 刘裕炜 | 程序开发  文档评审 | 项目全程 | 有小型个人项目及两人合作项目开发经历 | 有Java、Python开发经验和技能 |
| 裴宝琦 | 程序开发  文档评审  系统实施 | 项目全程 | 有小型个人项目及三人合作项目开发经历 | 有Java、Python开发经验和技能 |
| 廖纪童 | 程序开发  文档评审  软硬件协同 | 项目全程 | 有硬件加速机器学习项目经验与小型前端开发经验 | 熟悉C++、Python、Java 等程序设计语言，有相关开发经验与技能 |

## 产品

### 程序、数据或设备

* 项目源代码
* 可执行程序
* 数据文件
* 配置完成的启智ROS机器人及配套机械臂

### 文档

1. SDP-软件开发计划
2. SRS-软件需求规格说明书
3. SDD-软件设计说明
4. STP-软件测试说明

## 运行与开发环境

### 运行环境

本系统运行的环境包括硬件环境和软件环境两种环境，其中硬件环境为启智ROS机器人、配套机械臂和互联网连接。软件环境为Ubuntu 18.04 LTS、ROS Melodic Morenia以及启智ROS机器人相关基础包和驱动。

### 开发环境

本系统开发的环境包括硬件环境和软件环境，其中硬件环境为PC、相关外设及互联网连接。软件环境为Ubuntu 18.04 LTS、ROS Melodic Morenia、启智ROS机器人相关基础包和驱动、rviz可视化平台与gazebo仿真环境和RoboWare Studio集成开发环境。

## 项目期限

项目从2021年3月7日持续到2021年6月19日。

# 风险管理

项目可能的风险分类如下：

项目风险分类

|  |  |
| --- | --- |
| 风险类型 | 可能风险 |
| 估算方面 | 总体开发时间被低估；  各任务持续时间被低估；  人员工作量分配不合理。 |
| 技术方面 | 开发环境搭建存在问题；  仿真与实际行为存在差距。 |
| 人员方面 | 开发人员技术存在短板；  开发小组交流不足；  开发人员积极性不高。 |
| 需求方面 | 需求分析不够完善；  后续对需求进行频繁调整。 |

各项风险的可能性、影响、管理策略如下：

风险及对应管理策略

|  |  |  |  |
| --- | --- | --- | --- |
| 风险 | 可能性 | 影响 | 管理策略 |
| 总体开发时间被低估 | 高 | 延期交付 | 定期计算剩余工作量，并相应调整开发计划 |
| 各任务持续时间被低估 | 高 | 影响后续任务排期，最终可能导致软件延期交付 | 定期计算任务剩余工作量并调整计划 |
| 人员工作量分配不合理 | 中 | 拖累开发进度 | 开发组积极交流，灵活调整工作量 |
| 开发环境搭建存在问题 | 中 | 存在问题的开发者无法从事该环境下的开发工作 | 积极沟通帮助解决，解决不了则及时调整岗位 |
| 仿真与实际行为存在差距 | 高 | 需要返工修改 | 尽早尽快安排实机测试，尽早发现问题并修改 |
| 开发人员技术存在短板 | 中 | 对应工作完成存在困难，进而导致进度拖累 | 组内成员积极交流，并灵活调整分工 |
| 开发小组交流不足 | 高 | 开发效率低下，难以整合不同人开发的组件 | 定期开会并积极讨论进度和工作安排 |
| 开发人员积极性不高 | 低 | 开发进度推进慢，导致延期交付 | 定期开会并讨论，灵活调整任务分配 |
| 需求分析不够完善 | 低 | 后续需要补充分析遗漏需求 | 尽可能完善需求分析 |
| 后续对需求进行频繁调整 | 高 | 增加工作量，难以按照事先拟定的最优架构开发 | 每次需求调整时都尽量调整到位，减少调整次数，并及时修改开发计划 |

# 过程模型

本项目采用迭代-增量模型作为生存周期模型，下面介绍各阶段及其相关工作：

## 学习项目相关知识，搭建平台（3月7日-3月30日）

1. 配置Ubuntu开发环境，学习ROS相应的知识。（3月7日-3月13日）

2. 进行初步的需求分析，学习ROS编程的相关知识，编写简单的demo熟悉相关操作。（3月14日-3月20日）

3. 根据当前的知识再次进行需求分析，修改相应的需求，基本确定重要的里程碑时间节点。（3月21日-3月27日）

4. 细化开发工程的分工及相应时间节点（3月28日-3月30日）

## 初步实现相关功能，再次分析需求（3月31日-4月17日）

1. 整体设计框架，完善接口规范（3月31日-4月3日）
2. 从软硬件方面分别实现一定的基础功能，并进行测试。（4月4日-4月10日）
3. 根据已经实现的功能，确定迭代开发的具体流程，完善项目相应的基础功能，并完善相应的文档，（4月11日-4月17日）

至此，本项目应该完成初步迭代

## 完成项目基本功能，并进行相关测试（4月18日-5月15日）

## 完成项目扩展功能，并进行相关测试（5月16日-6月19日）

# 资源计划

项目所需要的软件资源需求：

* Ubuntu 18.04 LTS
* ROS Melodic Morenia
* Git版本管理工具
* 机器人仿真平台Gazebo 9
* C++
* Python 3

项目所需要的硬件资源需求：

* 机器人硬件平台
* 人机交互界面显示器
* 各硬件间的连接线
* 电源

# 进度计划

初步的进度计划如下，包括了需要进行的任务和其对应的起止时间、里程碑事件和任务之间的联系。

