文档编号：

迅管家应用程序

概要设计说明书

石家庄铁道大学

2019年6月

**变更历史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 版本日期 | 修订人 | 核准人 | 备注 |
| v1.0 |  |  |  | 初稿 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

目录

[1 引言 4](#_Toc11444641)

[1.1 编写目的 4](#_Toc11444642)

[1.2 编写背景 4](#_Toc11444643)

[1.3 定义 4](#_Toc11444644)

[1.4 参考资料 4](#_Toc11444645)

[2 系统设计 5](#_Toc11444646)

[2.1 总体设计原则 5](#_Toc11444647)

[2.2 总体架构 5](#_Toc11444648)

[2.3 技术架构 6](#_Toc11444649)

[2.4 运行环境 8](#_Toc11444650)

[3 功能设计 9](#_Toc11444651)

[3.1 系统整体业务功能 9](#_Toc11444652)

[3.1.1 进程系统业务功能关系 9](#_Toc11444653)

[3.1.2 性能系统业务功能关系 10](#_Toc11444654)

[3.1.3 应用系统业务功能关系 10](#_Toc11444655)

[3.1.4 网络系统业务功能关系 10](#_Toc11444656)

[3.2 系统功能清单 11](#_Toc11444657)

[3.3 各系统模块功能的设计 12](#_Toc11444658)

[3.3.1 进程系统功能模块 12](#_Toc11444659)

[3.3.2 性能系统功能模块 12](#_Toc11444660)

[3.3.3 应用系统功能模块 12](#_Toc11444661)

[3.3.4 网络系统功能模块 12](#_Toc11444662)

[4 系统性能保障措施 13](#_Toc11444663)

[4.1 架构设计性能保障原则 13](#_Toc11444664)

[4.2 多线程的系统体系架构 13](#_Toc11444665)

# 引言

## 编写目的

本阶段将会在需求分析阶段的基础上对迅管家做进一步的概要设计，主要包括进程、性能、应用、联网系统设计等。以上系统模块的设计将结合需求分析阶段的功能需求，把各模块间的关系给建立起来，从而完成整个系统的概要设计需求。

另外，在下一阶段的详细设计中，本阶段的概要设计将作为参考，以方便完成整个系统的设计工作

## 编写背景

项目名称：迅管家

项目提出者：同方股份有限公司

项目开发者：王莉 赵腾 杨蕾

## 定义

用户：迅管家使用者。

## 参考资料

1. 河北省迅管家审核系统-需求说明书 v1.0

# 系统设计

## 总体设计原则

为确保系统实施各阶段的质量要求得到满足，项目小组将按照ISO9001系列标准对项目进行质量管理和控制。各阶段包括分析、设计、开发、安装和维护等活动均按以下要求监控质量：

|  |  |
| --- | --- |
| **衡量标准** | **具体考虑因素** |
| 功能度 | 适配性、准确度、互操作性、依从性、安全性 |
| 可靠性 | 成熟度、容错性、可恢复性 |
| 可用性 | 可理解性、可学习性、可操作性 |
| 效率 | 时间特性、资源特性 |
| 可维护性 | 分析性、可改变性、稳定性、可测试性 |
| 可移植性 | 适应性、可安装性、一致性、可置换性 |

此外还提供与项目有关的质量保证计划，说明项目所提供的质量保证措施，所选择的措施的有效性及其影响。

## 总体架构

整个迅管家核系统的架构采用基于的三层架构模式，三层架构模式是一种软件架构模式。它把软件系统分为三个部分：业务逻辑层（BLL），数据访问层（DAL）和表现层（UI）。三层架构模式的目的是实现一种动态的程序设计，使后续对程序的修改和扩展简化，并且使程序某一部分的重复利用成为可能。除此之外，此模式通过对复杂度的简化，使程序结构更加直观。软件系统通过对自身基本部份分离的同时也赋予了各个基本部分应有的功能

表示层

业务逻辑层

数据访问层

底层API

1. **总体架构图**

数据访问层（DAL）：使用Linux命令获取底层的API来获取对进程、性能、应用、联网的数据。

业务逻辑层（BLL）：业务逻辑层是在数据访问层和表示层之间进行数据交换的桥梁，按业务需求调用数据访问层中的方法组合，集合了各种业务规则到一个业务逻辑层中，例如通过条件进行判断的数据操作或“事务”处理。业务逻辑层都是以类库（Class Library）的形式来实现的。

表示层（UI）：表示层是为客户提供用于交互的应用管家图形界面，帮助用户使用迅管家来查看进程、性能、应用、联网的数据。

三层结构是一种严格分层方法，即数据访问层(DAL)只能被业务逻辑层(BLL)访问，业务逻辑层只能被表示层(UI)访问，用户通过表示层将请求传送给业务逻辑层，业务逻辑层完成相关业务规则和逻辑，并通过数据访问层访问数据库获得数据，然后按照相反的顺序依次返回将数据显示在表示层

## 技术架构

整个迅管家的开发采用的是Spring Cloud应用体系架构,平台以Linux基本API数据管家作为平台的运行基础，在整个管家的过程中，以底层API作为资源，整个业务围绕着底层API展开。所有的显示数据数据都属于最底层的数据层可视化展示。整个系统的体系结构来划分的话可以分为表示层，业务逻辑层，数据访问层。如图2所示：



1. 技术架构图

**表示层：**

表示层由AWT和控制逻辑组成。

* AWT界面

AWT界面是由当前本机的进程、性能、应用、联网的数据和AWT的选项卡组成，主要作用是负责请求数据，并将数据传递给业务层处理，然后将结果呈现出来。

* 控制逻辑

控制逻辑是迅管家中页面的代码部分，负责处理和业务层之间的数据交互，状态流程的控制，同时负责简单的数据验证和格式化等功能。具体的说在Spring Cloud事件驱动的编程模型下，控制逻辑以各种事件函数方式实现。在这些事件函数中，主要任务就是实现迅管家的前台控件与业务实体的数据交换与业务调用。

**业务逻辑层：**

业务数据访问层由系统管家层组成。系统管家层为系统全局提供与业务关系不大的通用数据访问管家。

迅管家系统的业务数据主要就是进程、性能、应用、联网数据信息，业务数据访问层是一个针对访问这些数据的具体应用的专属层，它为业务层提供与数据源交互的最小操作方式，仅仅是业务层需要的数据访问接口，业务层完全依赖业务数据访问层所提供的管家。这些管家负责从业务层接收数据或返回业务实体，它屏蔽了实际业务数据与机器存储方式的差别。业务数据访问层最大的特点就是针对具体业务做抽象，而抽象的数据层访问方案是针对通用做抽象。

**数据访问层：**

数据访问层的目的就是使用linux命令获取底层的数据，为存储各进程、性能、应用、联网信息提供一个可供外界访问的接口。

## 运行环境

1. Linux 14及更高版本管家器
2. Java 1.8

# 功能设计

## 系统整体业务功能

迅管家应用程序分为进程、性能、应用、联网四大系统，其中进程系统分为进程内容显示、结束进程、结束进程树、设置优先级、内存显示更换、打开文件位置、属性；性能系统分为：CPU、内存、磁盘、网络相关信息显示；应用系统分为应用内容显示、结束任务、内存、打开文件位置、属性；联网系统分为应用网络状态显示、打开文件位置、属性。



1. 系统总体功能结构图

### 进程系统业务功能关系



1. **进程系统功能结构图**

### 性能系统业务功能关系

性能系统管理业务功能如下：



1. **性能功能结构图**

### 应用系统业务功能关系

应用系统业务功能如下：  


1. **应用功能结构图**

### 网络系统业务功能关系

网络系统业务功能如下：



1. **网络功能结构图**

## 系统功能清单

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **系统模块** | **一级功能** | **二级功能** |
| 1 | 进程系统 | 进程内容显示 | 内存显示更换 |
| 2 | 属性 |  |
| 3 | 进程操作 | 结束进程/树 |
| 4 | 设置优先级 |
| 5 | 打开文件位置 |  |
| 6 | 性能系统 | CPU | 图表及信息显示 |
| 7 | 内存 | 图表及信息显示 |
| 8 | 磁盘 | 图表及信息显示 |
| 9 | 网络 | 图表及信息显示 |
| 10 | 应用系统 | 应用内容显示 | 内存显示改变 |
| 11 | 属性 | 结束任务 |
| 12 | 打开文件位置 |  |
| 13 | 联网系统 | 应用网络状态显示 |  |
| 14 | 打开文件位置 |  |
| 15 | 属性 |  |

## 各系统模块功能的设计

### 进程系统功能模块

* 进程内容显示描述：通过linux命令显示进程的基本信息，进程名、优先级、PID、状态等。更改内存的显示形式。
* 属性描述：显示该进程的详细信息。
* 进程操作描述：可以更改进程的优先级，结束进程、结束进程树。
* 打开文件位置描述：打开该进程所在目录。

### 性能系统功能模块

* CPU功能描述：显示CPU的速度、型号等基本信息。
* 内存功能描述：显示内存的速度等基本信息。
* 磁盘功能描述：显示磁盘的速度等基本信息。
* 网络功能描述：显示网络的速度等基本信息。

### 应用系统功能模块

* 应用内容显示描述：通过linux命令显示应用的基本信息，进程名、优先级、PID、状态等。更改内存的显示形式。
* 属性描述：打开应用的详细信息。
* 打开文件位置描述：打开该应用所在目录。
* 进程系统功能模块

### 网络系统功能模块

* 打开网络状态显示描述：显示应用的网络信息。
* 属性描述：显示使用网络的应用的详细信息。
* 打开文件位置描述：打开使用网络的应用的位置。

# 系统性能保障措施

## 架构设计性能保障原则

* 面向性能的业务流程分析

明确性能问题在整个软件生产与运行过程中的重要意义，在考虑架构分析与设计的时候，出发点与指导原则是帮助用户使用某种技术手段来高效地完成业务流程，其本质是“高效的业务流程”，而不是一个计算机系统或计算机应用。在这一原则之下，我们的应用开发是围绕着开发高效“业务流程”展开的，spring cloud技术只是我们的一种技术手段而已，避免由于具体的技术实现方案对业务流程分析中的性能指标的束缚。

## 多线程的系统体系架构

采取不同的进程与线程进行实时展示。加快数据的更新频率，提高系统的性能，增强用户的体验，提供用户的满意度。