# 设计文档

* 系统名称： minishell
* 文档作者：苏志伟
* 创作时间：2023-8-4
* 最新修改时间：2023-8-4

## 背景描述

minishell是命令行操作系统属于shell的迷你版，功能不如shell强大。

* 客户名称： GUO
* 开发者名称：苏志伟

## 概要设计文档

### 2.1系统结构图

数据输入与显示（控制台）

基于网络爬虫技术的网络数据查询

图1.

缓存区域

### 2.2模块介绍：

* 数据输入与显示（控制台）： 主要用于用户输入天气数据查询的条件信息， 以及显示查询结果
* 数据检查模块： 用于检查用户输入的数据是否有效，格式是否正确
* 时间段数据查询： 用于收集时间段数据查询所需要的所有数据， 调用网络数据查询模块， 同时对网络数据查询模块的返回值进行整理， 使其按照规定的格式返回控制台进行显示， 并且把查询结果暂存到缓存区用于后续的打印。查询时间以用户输入的起始日期开始，输入的终止日期结束
* 实时数据查询： 用于收集实时数据查询所需要的所有数据， 调用网络数据查询模块， 同时对网络数据查询模块的返回值进行整理， 使其按照规定的格式返回控制台进行显示，并且把查询结果暂存到缓存区用于后续的打印，查询时间区间为以输入有效的终止时间的时刻为准的操作系统时间为起始时间， 以输入的终止时间为结束时间
* 数据打印模块： 当用户选择数据打印时， 从暂存区读出数据进行打印
* 缓存区域： 暂存最后一次查询的数据， 用于用户数据的打印 *（注：本人对嵌入式技术不太了解，这部分的设计有待商榷改正）*

## 详细设计文档

*（注：详细设计是相对于概要设计而言的， 可以作为需求人员、总体设计人员与开发人员的沟通工具，以确保三方始终保持在同一理解的层面上。详细设计的目的主要是把静态页面无法体现的设计和逻辑呈现出来，包括：算法设计、接口设计、数据结构设计、模块之间的交互设计等等。 良好的详细设计可以帮助开发人员能快速进入开发，提高沟通效率，以及减少沟通成本。通常也可以使用其他绘图工具预先画出一个假的界面，以供讨论使用。 详细设计文档中会使用到流程图等*

### 3.1数据输入与显示

* 流程图：(见附录图2.)

### 3.2数据检查模块

* 接口说明：

每次用户输入查询所需要的条件数据时都需要及时检查数据的有效性， 如果输入的数据无效，应保持当前 步骤，等待用户重新输入

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 参数说明 | 返回值 | 抛出异常 | 接口描述 |
| Cp\_handle() | String provinceName | boolean  true: 有效信息， false： 无效信息 |  | 检查输入的省份信息是否有效 |
| checkCity | String cityName | boolean  true: 有效信息， false： 无效信息 |  | 检查输入的城市信息是否有效 |
| checkDate | String startDate, String endDate | String  “”: 日期有效  当日期无效时返回具体信息 |  | 检查输入的日期区间是否有效（时间段数据查询时使用） |
| checkEndTime | String endTime | String  “”: 时间有效  当时间无效时返回具体信息 |  | 检查输入的结束时间是否有效（实时数据查询时使用），如果数据有效，需要同时设置开始时间，即操作系统当前时间 |
| checkQueryType | int | boolean  true: 有效信息， false： 无效信息 |  | 检查输入的查询类型是否有效：  1：时间段数据查询  2：实时数据查询 |

### 3.3时间段数据查询

该模块可以根据输入的省份， 城市，以及起止日期信息到网络中查询匹配的天气数据。

* 接口说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 参数说明 | 返回值 | 抛出异常 | 接口描述 |
| queryWeatherData | String provinceName,  String cityName,  String startDate,  String endDate | List<String> |  | 1 列表中的每一个元素对应于控制台上的每一行输出  2 列表中的每一个元素会按照定义的输出格式准备 |

### 3.4实时数据查询

该模块可以根据输入的省份， 城市，以及终止时间信息到网络中查询匹配的天气数据，起始时间以输入有效的终止时间的时刻为准的操作系统时间

* 接口说明:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 参数说明 | 返回值 | 抛出异常 | 接口描述 |
| queryRealTimeWeatherData | String provinceName,  String cityName,  String startTimeDate,  String endTimeDate | List<String> |  | 1 列表中的每一个元素对应于控制台上的每一行输出  2 列表中的每一个元素会按照定义的输出格式准备 |

### 3.5基于爬虫技术的网络数据查询

此模块应为一个通用组件， 主要实现了安装预设条件到特定网络中查询匹配的数据

* 接口说明:

*（注：由于本人不太了解网络爬虫技术的实现原理，此处不再细说。唯一建议是尽可能的把接口设计的通用一些， 而不是仅仅为了查询天气数据）*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 参数说明 | 返回值 | 抛出异常 | 接口描述 |
| queryData |  |  |  | 根据条件到网络中获取相关数据 |

### 3.6数据打印模块

此模块主要用于数据的打印。正常情况下还需要包括打印机管理（例如， 新增一台打印机， 删除一条打印机信息， 修改打印机信息等等），以及检测当前使用的打印机是否在线等功能。目前仅仅默认为打印机以及就绪， 可以随时打印数据，因此只有一个借口print

* 接口说明:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口名称 | 参数说明 | 返回值 | 抛出异常 | 接口描述 |
| print | Int cacheID | boolean  true: 打印成功  false： 打印失败 |  | 打印缓存区域的暂存数据 |

## 附录：

### 文档修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 修改日期 | 参与者 | 版本号 | 修改描述 |
| 2019-07-23 | Xhan | 1.2 | 添加了详细设计中对每个模块的描述 |
|  |  |  |  |

### 4.2数据输入与现实模块流程图

开始

N

时间格式？是否大于当前系统时间？

N

Y

日期格式？

终止>起始？

输入1or2？

显示查询结果

输入终止时间

输入终止日期

输入起始日期

输入城市

输入省份

选择查询类型

结束

图2. 流程图

打印查询结果

Y

N

N

是否打印？

查询结果是否为空

Y

2：实时数据查询

1：时间段查询

Y

N

Y

N

输入省份是否存在

Y

N

根据查询类型显示

输入城市是否存在