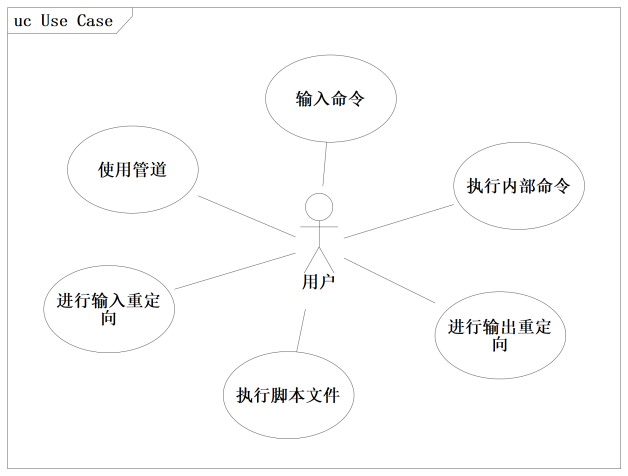
# 项目开发文档

## 项目概述

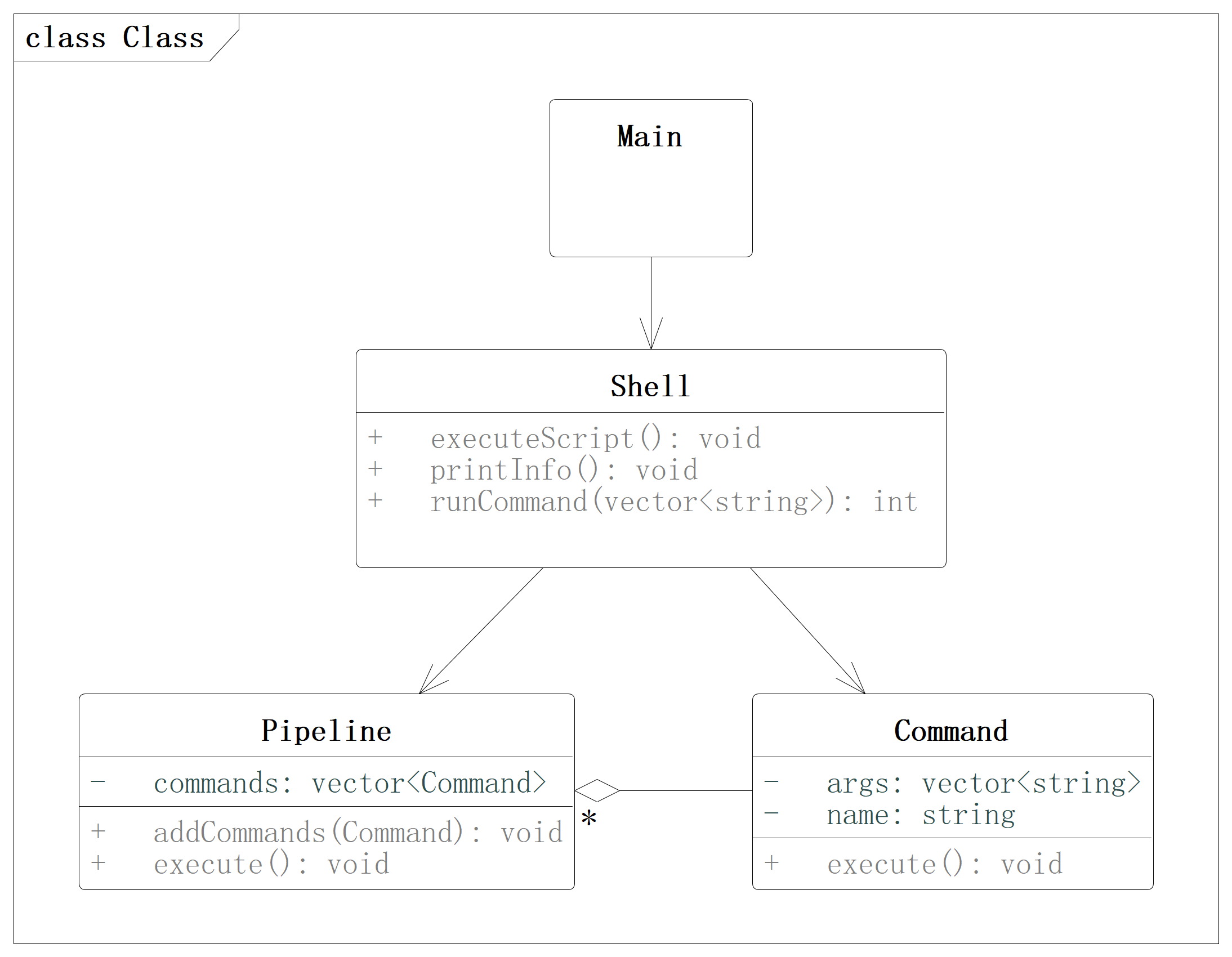
本项目开发了一个简单但功能强大的Shell程序，旨在模拟Unix/Linux系统中的Shell功能。该Shell程序能够处理用户输入的命令，并支持多种功能，如管道、输入输出重定向、脚本执行和内部命令等。整个项目的开发过程中，应用了各种软件工程设计方法和工具，包括UML建模和单元测试，确保了项目的可读性、可维护性和稳定性。

## UML模型图

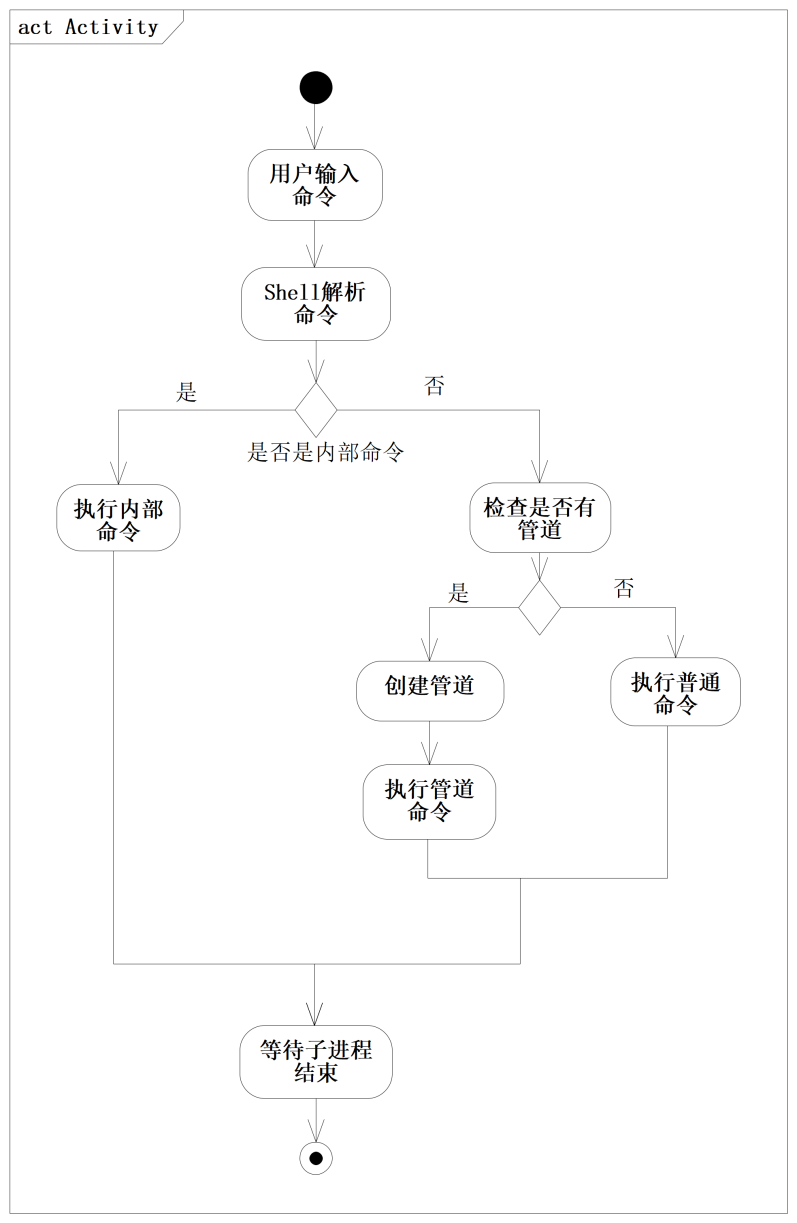
用例图



类图



活动图



## 设计文档

### 功能模块

**1.命令解析模块**

·功能：解析用户输入的命令字符串，分割成命令和参数。

·输入：用户输入的命令字符串。

·输出：解析后的命令和参数列表。

**2.命**令执行模**块**

·功能：根据解析后的命令和参数执行命令。

·输入：命令和参数列表。

·输出：执行结果。

**3.管道处理模块**

·功能：处理管道符号|，实现命令之间的数据传递。

·输入：包含管道符号的命令列表。

·输出：管道执行结果。

4.输入输出重定向模块

·功能：处理输入重定向<和输出重定向>。

·输入：包含重定向符号的命令列表。

·输出：重定向后的执行结果。

5.脚本执行模块

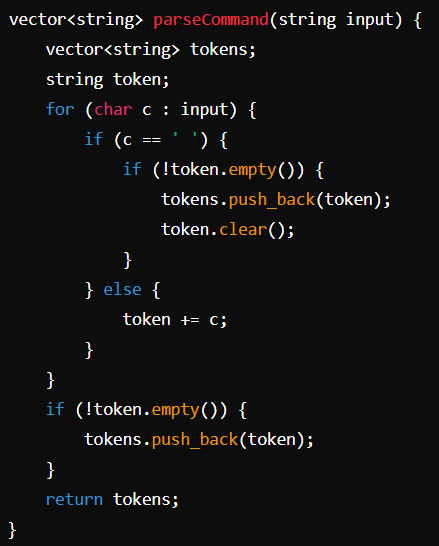
·功能：从脚本文件中读取命令并依次执行。

·输入：脚本文件路径。

·输出：脚本执行结果。

### 算法描述

1. 命令解析算法



1. **管道处理算法**



## 测试文档

### 测试目的

验证自定义Shell程序的功能是否正常，包括基本命令执行、目录切换、内部命令、输入输出重定向和管道操作。

### 测试环境

* 操作系统：Linux/Unix
* Shell程序：myshell

### 测试用例

测试用例1：普通命令执行

命令： ls

预期结果：列出当前目录的文件和文件夹。

实际结果：列出当前目录的文件和文件夹。

测试用例2：切换到上一级目录

命令： cd ..

预期结果；切换到上一级目录。

实际结果：成功切换到上一级目录。

测试用例3：切换到指定目录

命令： cd MyShell

预期结果：切换到MyShell目录。

实际结果：成功切换到MyShell目录。

测试用例4：显示帮助信息

命令： help

预期结果：显示Shell程序支持的内部命令及其用法。

实际结果：成功显示帮助信息，包括cd, exit, help等内部命令。

测试用例5：输出重定向

命令； ls > output.txt

预期结果：将ls命令的输出写入output.txt文件。

实际结果：ls命令的输出被成功写入output.txt文件，内容与在终端执行ls命令的结果相同。

测试用例6：输入重定向

命令：wc -l < output.txt

预期结果：计算output.txt文件的行数并输出结果。

实际结果：成功输出output.txt文件的行数，与直接执行wc -l output.txt命令的结果相同。

测试用例7：管道操作

命令： ls | grep myshell

预期结果：列出当前目录中包含myshell的文件和文件夹。

实际结果：成功列出当前目录中包含myshell的文件和文件夹，结果与在终端执行相同命令的一致。

### 结论

所有测试用例均通过，Shell程序功能正常，满足预期需求。测试覆盖了普通命令执行、目录切换、内部命令、输入输出重定向和管道操作，确保了程序的基本功能和可靠性。