**详细设计文档**

**学 院：电气信息**

**专 业：软件工程**

**课 程 名 称：团队项目开发**

**团 队 编 号：第六组**

**团 队 成 员：黄骏力（组长）、**

**王惠琳（会议秘书）、**

**陈晓红、郭梓明、**

**王俊谕、朱俊廷**

**2018年5月10日星期四**

目录

[1导言](#_Toc2680_WPSOffice_Level1) [1](#_Toc2680_WPSOffice_Level1)

[1.1目的](#_Toc8132_WPSOffice_Level1) [1](#_Toc8132_WPSOffice_Level1)

[1.2范围](#_Toc22417_WPSOffice_Level1) [1](#_Toc22417_WPSOffice_Level1)

[1.3编写说明](#_Toc24139_WPSOffice_Level1) [1](#_Toc24139_WPSOffice_Level1)

[1.4术语定义](#_Toc16301_WPSOffice_Level1) [2](#_Toc16301_WPSOffice_Level1)

[1.5引用标准](#_Toc31489_WPSOffice_Level1) [2](#_Toc31489_WPSOffice_Level1)

[1.6参考资料](#_Toc10764_WPSOffice_Level1) [2](#_Toc10764_WPSOffice_Level1)

[1.7版本更新信息](#_Toc15245_WPSOffice_Level1) [2](#_Toc15245_WPSOffice_Level1)

[2 系统设计概述](#_Toc8132_WPSOffice_Level2) [2](#_Toc8132_WPSOffice_Level2)

[3. 详细设计概述](#_Toc22417_WPSOffice_Level2) [3](#_Toc22417_WPSOffice_Level2)

[3.1版本更新](#_Toc8132_WPSOffice_Level3) [3](#_Toc8132_WPSOffice_Level3)

[3.2管理配置](#_Toc22417_WPSOffice_Level3) [3](#_Toc22417_WPSOffice_Level3)

[4. 详细设计](#_Toc24139_WPSOffice_Level2) [4](#_Toc24139_WPSOffice_Level2)

[4.1版本更新界面](#_Toc24139_WPSOffice_Level3) [4](#_Toc24139_WPSOffice_Level3)

[4.2管理配置界面](#_Toc16301_WPSOffice_Level3) [8](#_Toc16301_WPSOffice_Level3)

[5. 程序提交清单](#_Toc16301_WPSOffice_Level2) [17](#_Toc16301_WPSOffice_Level2)

**1导言**

**1.1目的**

对自动版本更新软件的内容实现进行详细设计

**1.2范围**

系统设计概述，详细设计概述，详细设计，程序提交清单

**1.3编写说明**

PHP（外文名:PHP: Hypertext Preprocessor，中文名：“[超文本](https://baike.sogou.com/v42705.htm" \t "https://baike.sogou.com/_blank)预处理器”）

MVC，全名是Model View Controller，是软件工程中的一种软件架构模式，把软件系统分为三个基本部分：模型（Model）、视图（View）和控制器（Controller）

**1.4术语定义**

无

**1.5引用标准**

无

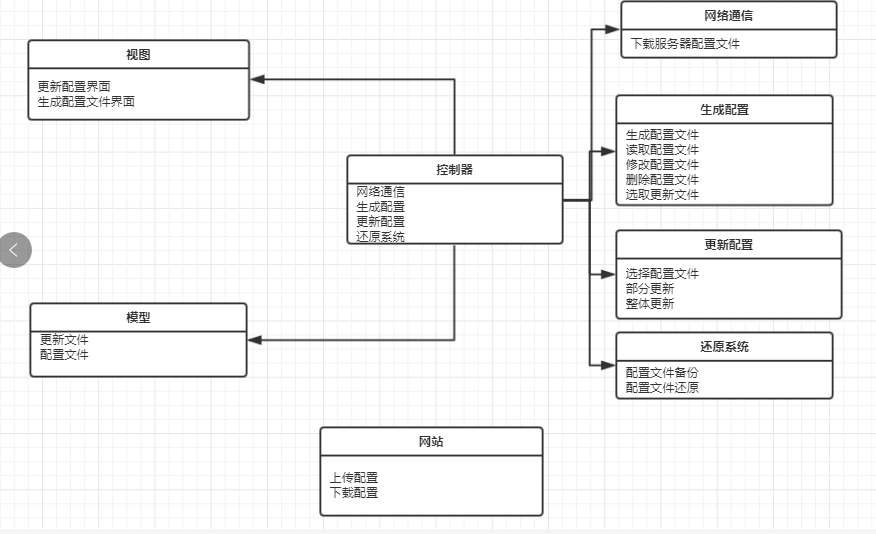
**1.6参考资料**

课本案例

**1.7版本更新信息**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 修改编号 | 修改日期 | 修改后版本 | 修改位置 | 修改内容概述 |
| 1.0 | 6.13 | 2.0 | 底层实现部分 | 将更新功能实现为一个库，允许其他软件对该功能进行调用 |

**2 系统设计概述**



1. **详细设计概述**

**3.1版本更新**

**3.1.1显示配置差异**

用途：显示两个版本的配置信息，从而让用户直观了解新旧版本之间的区别。

功能: 读取最新的配置版本和本机应用的版本信息并显示。

**3.1.2更新实现**

用途：为用户提供全部更新和部分更新两种更新方式。

功能：根据用户对更新方式和更新文件的具体选择生成新的配置文件，对软件进行更新。

**3.1.3更新保护机制**

用途：防止由于安装过程中可能由于各种因素（如网络中断或者硬件故障等）导致文件安装失败，导致软件无法使用。

功能：在更新失败时弹窗提示用户是否还原原本配置，用户确认后按照日志文件的记录将原有的版本配置文件复制回去。

3.2管理配置

3.2.1历史管理

用途：显示历史配置及详情，并作出相应操作。

功能：能够对配置文件进行修改、删除和另存为操作

3.2.2生成新配置

用途：根据用户需要生成所需的配置文件

功能：可以选择历史配置进行覆盖，也可以选择新的文件生成。

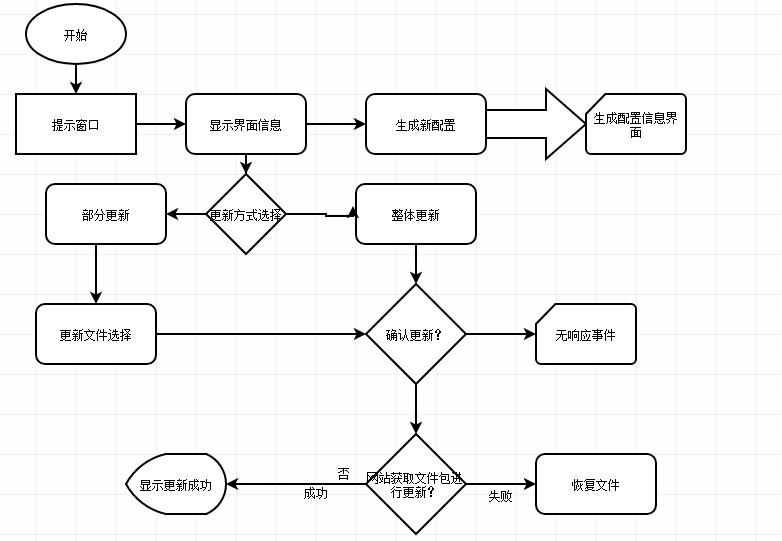
1. **详细设计**

**4.1版本更新界面**

1. **界面基本流程**

**更新界面**

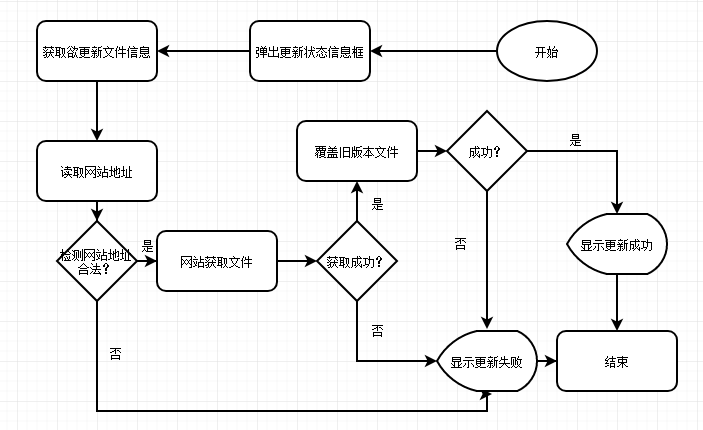
更新界面弹出显示版本更新的相关信息，并进行版本更新的操作，可以选择部分或整体更新，更新过程中从网站获取文件。如果更新成功，则显示更新成功提示；如果失败，则显示更新失败提示。



基本流流程：

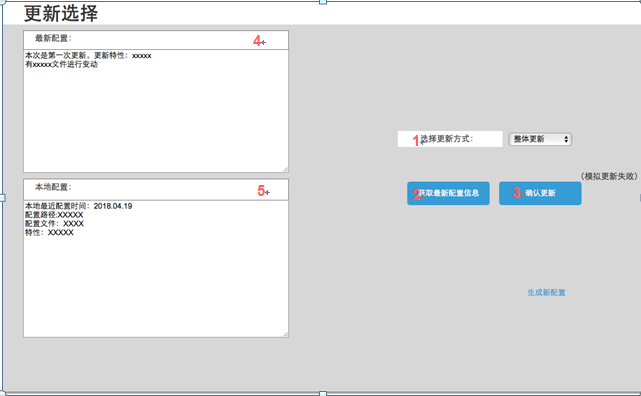
1. 用户进入更新界面，左侧是版本信息的显示框，右侧为更新模块，选择更新方式。
2. 用户如果选择部分更新，则弹出文件选择框，可以复选文件；如果选择整体更新，则直接进入下一步。
3. 用户单击“确定更新”按钮，系统将执行其监听事件。
4. 如果更新成功，将在用户界面显示成功更新提示。

**更新确定按钮的监听事件：**

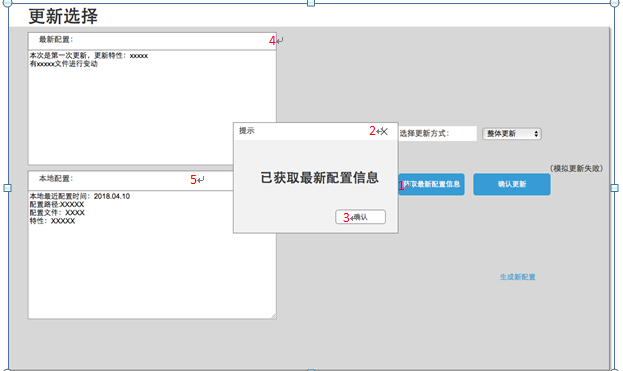


基本流流程：

1. 用户进入“确定更新”按钮，会弹出更新状态信息框，显示更新的最新动态情况。
2. 系统将获取欲更新的文件信息，并读取系统已经设置好的网站的地址。
3. 系统检测是否合法，如果合法，则执行下一步；否则显示更新失败并结束事件。
4. 系统从网站获取文件，如果成功，则执行下一步；否则显示更新失败并结束事件。
5. 系统覆盖旧版本文件，如果成功，则显示更新成功，正常结束事件。否则显示更新失败，异常结束事件。
6. **界面初始状态**

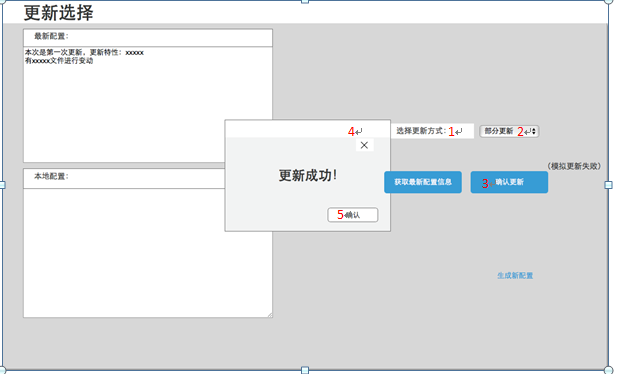
****

|  |  |  |  |
| --- | --- | --- | --- |
| 控件 | 名称 | 控件类型 | 状态 |
| 1 | 选择更新方式列表 | 下拉列表 | 等待点击 |
| 2 | 获取最新配置信息按钮 | 按钮 | 等待点击 |
| 3 | 确认更新按钮 | 按钮 | 等待点击 |
| 4 | 最新配置显示区 | 文本框 | 等待更新，显示字符串为当前上次传入数据，即上次传入版本信息，包括更新序号，特性，及与上一版本间差异 |
| 5 | 本地配置显示区 | 文本框 | 等待更新，显示字符串为上次传入数据信息，描述上次配置时间、配置路径、配置特性即对应描述； |

1. 事件响应
2. 获取最新配置信

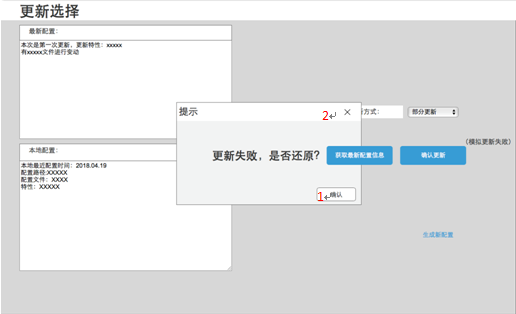
|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 动作 | 响应 | 状态变化 |
| 获取最新配置信息 | 点击获取最新配置信息按钮（控件1） | 后台获取最新配置，并且传入最新配置显示区中（控件4），更新控件4完成后弹出提示弹窗2显示获取结束；弹窗关闭前“获取最新配置按钮”无法再次点击 | 获取最新配置按钮：从等待被点击状态->点击被触发->无法再次点击 |
| 确认获取最新配置信息 | 点击确认获取最新配置信息按钮（控件3） | 关闭当前弹窗，改变“获取最新配置按钮”状态，可再次点击 | 部分更新按钮：等待被点击状态->点击被触发->等待取消点击  更新方式下拉列表：被触发并等待点击完成->点击完成->等待被点击  获取最新配置按钮：无法点击->等待再次 |

II．版本更新->更新成功：



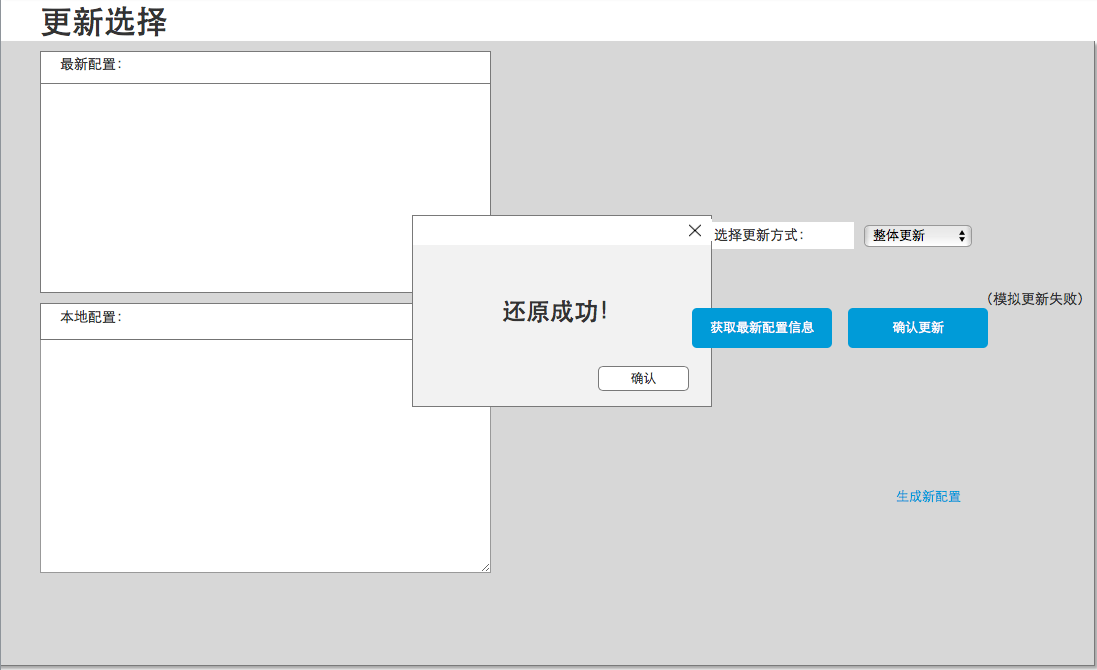
|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 动作 | 响应 | 状态变化 |
| 选择更新方式 | 点击更新方式列表 | 列表显示下拉选项，并且在当前默认选项前打勾 | 下拉列表从等待被点击状态->点击被触发并等待点击完成 |
| 选择部分更新 | 部分更新按点击钮 | 更新方式下拉列表收起，并显示当前的更新方式为“部分更新”；将改变的选择状态传入后台 | 部分更新按钮：等待被点击状态->点击被触发->等待取消点击  更新方式下拉列表：被触发并等待点击完成->点击完成->等待被点击 |
| 选择整体更新 | 点击整体更新按钮 | 更新方式下拉列表收起，并显示当前的更新方式为“整体”；将改变的选择状态传入后台 | 整体更新按钮：等待被点击状态->点击被触发->等待取消点击  更新方式下拉列表：被触发并等待点击完成->点击完成->等待被点击 |
| 确认更新 | 点击确认更新按钮 | 获取当前更新方式，并开始更新；更新期间按钮无法被点击 | 确认更新按钮：等待点击->点击本触发->更新结束时恢复等待点击状态 |
| 更新结果提示 | 被动触发，无用户动作 | 更新结束，在当前界面弹出弹窗，系统根据不同完成状态（更新成功或失败）显示弹窗内容。 | 提示弹窗：被动弹出并等待关闭 |
| 关闭提示弹窗 | 点击弹窗中确认按钮或右上角关闭按钮 | 弹窗关闭，系统按照更新状态选择触发更新“最新配置区”和“本地配置”区。分别更新最新配置信息和本地配置信息 | 提示弹窗：等待关闭->被关闭  最新配置区：等待更新->更新触发->等待更新  本地配置区：等待更新->更新触发->等待更新 |

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 动作 | 响应 | 状态变化 |
| 确认还原 | 点击确认按钮（控件1） | 系统进行更新还原操作，在此期间关闭按钮（控件2）无法被继续点击；还原完成后，弹出提示弹窗显示还原完成。 | 确认按钮：等待点击->点击触发->无法再次点击  关闭按钮：等待点击->无法点击 |

III．更新失败：

IIII．还原成功状态：

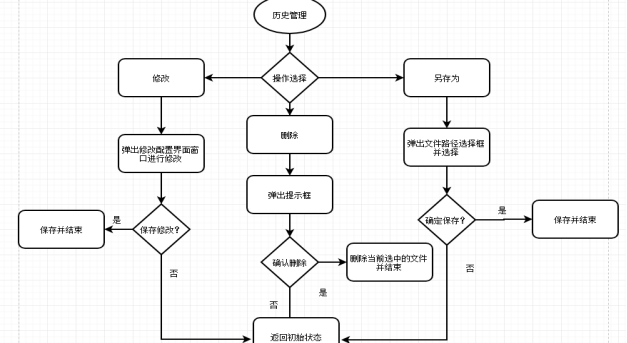
|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 动作 | 响应 | 状态变化 |
| 确认还原成功 | 点击确认按钮 | 弹窗关闭 | 提示弹窗：显示->隐藏 |

****

**4.2管理配置界面**

1. **界面基本流程**

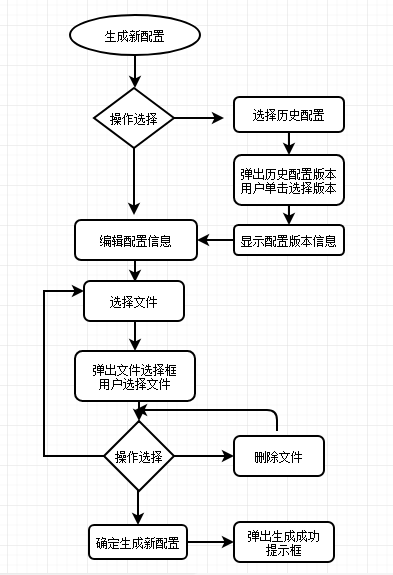
**历史版本管理模块**



基本流流程：

1. 用户进入管理配置界面，点击左上方历史配置版本，可看到该版本的配置信息。
2. 用户单击“修改”按钮可对历史配置版本进行修改操作，弹出修改配置窗口，修改后保存，回到初始状态。
3. 用户单击“删除“按钮可对历史配置版本进行删除操作，弹出操作提示框，若选择是，则删除并保存，若选择否，则回到初始状态。
4. 用户单击“另存为”按钮可对历史配置版本进行备份操作，在弹出的文件路径选择窗口进行路径选择，确定后保存，若选择否回到初始状态。

**配置文件模块**

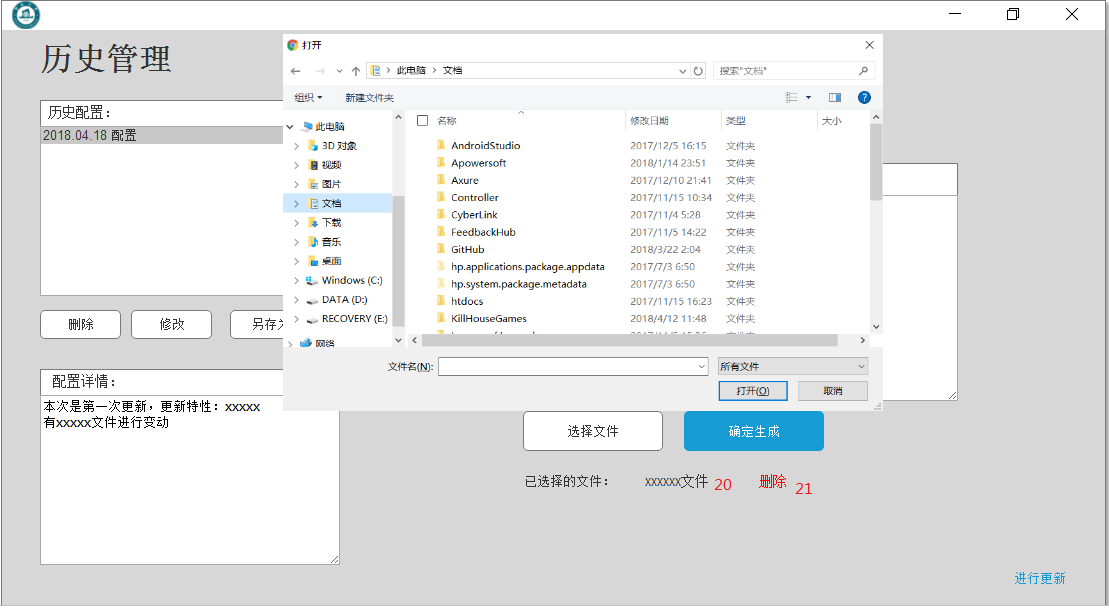
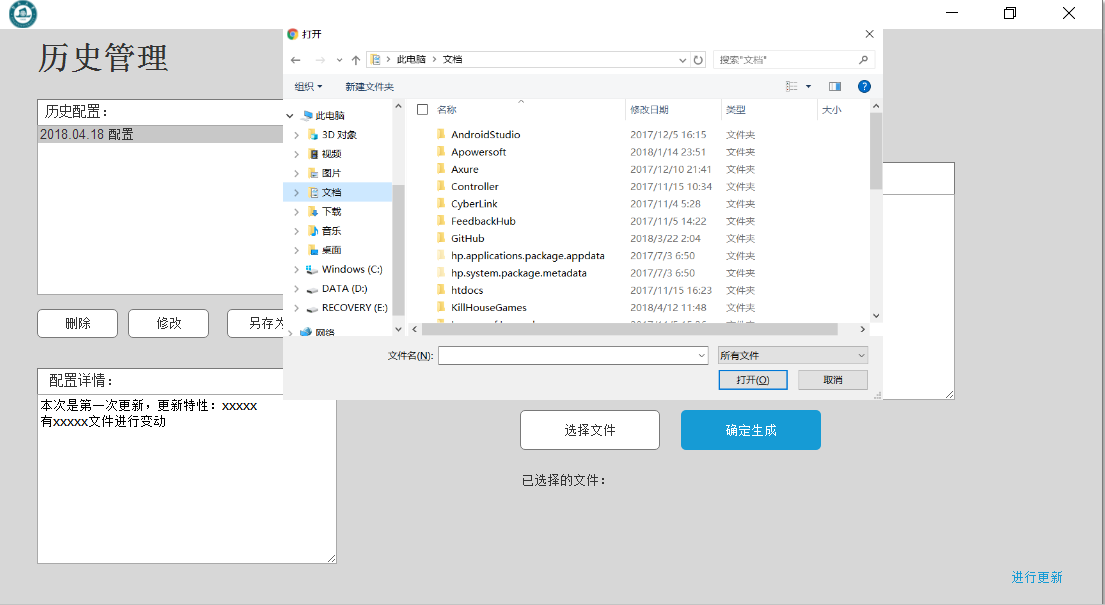
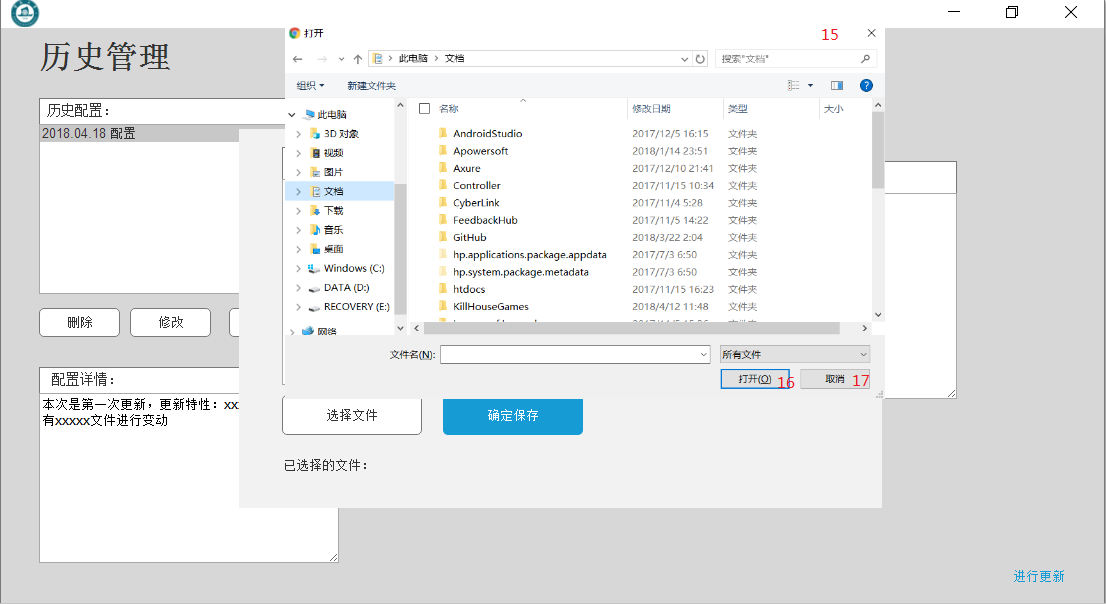


基本流流程：

1. 用户进入管理配置界面，右侧为生成新配置模块，单击“历史配置”按钮，在弹出窗口中单击选择历史版本，界面显示配置版本信息。
2. 用户可在空白配置编辑区或者已显示的配置版本信息上编辑配置信息。
3. 用户单击“选择文件”按钮，在弹出的文件选择窗口中选择文件，单击确定后选中的文件显示在界面中。
4. 用户可重复操作步骤（3）
5. 用户可单击选中的文件后的“删除”选项删除选中的文件。

用户单击“确定生成”按钮可以生成新的配置文件，弹出生成成功的提示框。

1. **界面初始状态**



|  |  |  |  |
| --- | --- | --- | --- |
| 控件 | 名称 | 控件类型 | 状态 |
| 1 | 某一个历史配置 | 数据项 | 默认未点击，点击后在配置详情显示区显示该配置的信息 |
| 2 | 历史配置显示区 | 列表框 | 默认未选择 |
| 3 | 删除历史配置 | 按钮 | 默认不可点击，选择某个历史配置后可点击 |
| 4 | 修改历史配置 | 按钮 | 默认不可点击，选择某个历史配置后可点击 |
| 5 | 另存为历史配置 | 按钮 | 默认不可点击，选择某个历史配置后可点击 |
| 6 | 配置详情显示区 | 文本框 | 不可操作，选择某个历史配置后显示该配置的信息 |
| 7 | 选择历史配置 | 按钮 | 可点击 |
| 8 | 编写配置区 | 文本框 | 默认不可输入 |
| 9 | 选择编写配置的文件 | 按钮 | 可点击 |
| 10 | 确定生成新配置 | 按钮 | 默认不可点击，用后完成编写配置和选择配置文件后可点击 |
| 11 | 进行更新 | 按钮 | 可点击，点击跳转更新界面 |
| 12 | 修改配置区 | 文本框 | 可输入 |
| 13 | 选择文件 | 按钮 | 可点击 |
| 14 | 确定保存修改的配置 | 按钮 | 默认不可点击，用户修改操作后可点击 |
| 15 | 浏览选择文件 | 窗体 | 未弹出 |
| 16 | 打开选择的文件 | 按钮 | 默认不可点击，选择文件后可点击 |
| 17 | 取消浏览文件 | 按钮 | 可点击，点击后弹出窗体消失 |
| 18 | 某一个历史配置 | 数据项 | 默认未点击，双击后可在编写配置区输入生成新配置 |
| 19 | 历史配置选择 | 列表框 | 默认为选择 |
| 20 | 已选择的文件 | 文本显示 | 不可操作 |
| 21 | 删除已选择文件 | 按钮 | 可点击 |
| 22 | 提示新配置生成 | 窗体 | 未弹出 |
| 23 | 确认新配置生成 | 按钮 | 可点击，点击后弹出窗体消失 |

（3）事件响应

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 动作 | 响应 | 状态变化 |
| 显示配置详情 | 点击控件1（4.18配置） | 配置详情文本框显示点击配置的详情 | 控件3、4、5：不可点击->可点击 |
| 删除配置 | 点击控件3 | 删除选中的配置；配置详情显示区变空白 | 控件1消失；  控件3、4、5：可点击->不可点击 |
| 修改配置 | 点击控件4 | 弹出修改配置窗口 | 弹出窗口 |
| 选择修改的配置 | 点击控件13 | 弹出浏览文件窗口 | 弹出窗口 |
| 打开选择的修改文件 | 点击控件16 | 显示选择文件的路径，浏览文件窗口消失 | 控件14：不可点击->可点击 |
| 取消选择修改文件 | 点击控件17 | 取消浏览文件，浏览文件窗口消失 | 浏览文件窗口消失 |
| 保存修改的配置 | 点击控件14 | 保存修改的配置 | 选中配置的信息被新修改的覆盖 |
| 另存为历史配置 | 点击控件5 | 弹出浏览文件窗口，选择保存路径 | 弹出窗口 |
| 选择历史配置 | 点击控件7 | 显示浏览历史配置窗口 | 弹出窗口 |
| 选择某个历史配置 | 双击控件18 | 编写配置区显示选中配置的信息，浏览历史配置窗口消失 | 控件8：不可输入->可输入  控件9、10：不可点击->可点击 |
| 选择新建配置的文件 | 点击控件9 | 弹出浏览文件窗口 | 弹出窗口 |
| 打开选择新建的文件 | 选择所需文件并点击打开 | 显示已选择的文件 | 删除已选择文件的按钮：可点击 |
| 生成新配置 | 点击控件10 | 弹出提示新配置生成窗口 | 弹出控件22；控件23：可点击；  历史配置区新建新的数据项（4.19） |
| 确认生成新配置 | 点击控件23 | 提示窗口消失 | 控件22消失 |

1. **程序提交清单**
2. **日志规范**

6.1.日志库的选择

使用**[SLF4J](https://link.jianshu.com?t=http://www.slf4j.org/" \t "_blank)**（Simple Logging Facade for Java）作为日志的api，SLF4J是一个用于日志系统的简单Facade，允许最终用户在部署其应用时使用其所希望的日志系统。与使用apache commons-logging和直接使用log4j相比，SLF4J提供了一个名为参数化日志的高级特性，可以显著提高在配置为关闭日志的情况下的日志语句性能。

6.2.日志级别设置

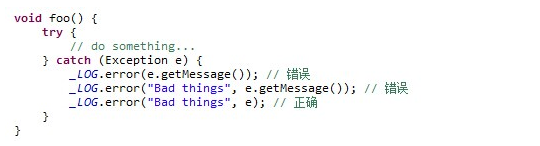
日志使用debug，info，warn，error四个级别，对于不同的级别设置不同的输出路径,debug，info输出到一个文件，warn，error输出到一个带error后缀的文件。

6.3 日志编码规范

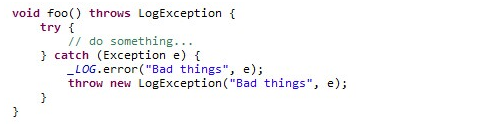
1）.在一个对象中通常只使用一个Logger对象，Logger应该是static final的，只有在少数需要在构造函数中传递logger的情况下才使用private final。



2）.输出Exceptions的全部Throwable信息，因为logger.error(msg)和logger.error(msg,e.getMessage())这样的日志输出方法会丢失掉最重要的StackTrace信息。



3）.不允许记录日志后又抛出异常，因为这样会多次记录日志，只允许记录一次日志。



4）.不允许出现System print(包括System.out.println和System.error.println)语句。  


5）.不允许出现printStackTrace。



6）.日志性能的考虑

如果代码为核心代码，执行频率非常高，则输出日志增加判断，尤其是低级别的输出<debug、info、warn>。

debug日志太多后可能会影响性能，有一种改进方法是：  


但更好的方法是Slf4j提供的[最佳实践](http://www.slf4j.org/faq.html" \l "logging_performance" \t "https://www.cnblogs.com/Asen0713/p/_blank):



一方面可以减少参数构造的开销，另一方面也不用多写两行代码。

7）.有意义的日志

通常情况下在程序日志里记录一些比较有意义的状态数据：程序启动，退出的时间点；程序运行消耗时间；耗时程序的执行进度；重要变量的状态变化。

初次之外，在公共的日志里规避打印程序的调试或者提示信息。

6.4.配置规范

1）统一使用log4j.xml、log4j2.xml、logback配置。

2）所有的jar包中不包含log4j.xml、log4j.properties、logback.xml文件，避免干扰实际的业务系统。

3）Log文件位置和命名，目前Log文件的位置统一放在相同目录下面，Log名字通常以业务名开头，如xxx.log.2015-11-19等。

4）日志格式：必选打印数据项: 发生时间、日志级别、日志内容,可选文件和行号。

5）远程日志的输出需要注意host和port，区分cagegory。