

属性名	多重性	解释	数据类型	聚合关系	组合关系	关联关系
userInfoField	1	用户信息域	Field	无	无	无
fileInformationField	1	文件信息域	Field	无	无	无
pageTurnButtons	1...10	页跳转按钮	Button	无	无	无

c) 操作说明

操作名	主动性	多态性	解释	约束条件及其他
getUserInformation	无-	无	获取该页用户信息	页码有效时执行
getFileInformation	无	无	获取该页文件信息	页码有效时执行

d) 关联

无

e) 泛化

无

f) 依赖

依赖 Administrator 类。

1.2.2 LoginInterface 类

a) 类的整体说明：

类名为 LoginInterface，对应人机交互部分的登录界面，负责为系统管理员或用户提供可视化的登录操作接口。需要保证密码项加密传输。

b) 属性说明

属性名	多重性	解释	数据类型	聚合关系	组合关系	关联关系
emailBox	1	邮箱输入框	Box	无	无	无
passwordBox	1	密码输入框	Box	无	无	无
loginButton	1	登录按钮	Button	无	无	无

c) 操作说明

操作名	主动性	多态性	解释	约束条件及其他
login	无-	无	根据输入用户/管理员信息进行登录	无

d) 关联

无

e) 泛化

无

f) 依赖

依赖 User 类与 Administrator 类。

1.2.3 RegisterInterface 类

- a) 类的整体说明：
类名为 RegisterInterface，对应人机交互部分的注册界面，负责为用户提供可视化的注册操作接口。需要保证密码项加密传输。
- b) 属性说明
- | 属性名 | 多重性 | 解释 | 数据类型 | 聚合关系 | 组合关系 | 关联关系 |
|----------------|-----|-------|--------|------|------|------|
| nicknameBox | 1 | 昵称输入框 | Box | 无 | 无 | 无 |
| emailBox | 1 | 邮箱输入框 | Box | 无 | 无 | 无 |
| passwordBox | 1 | 密码输入框 | Box | 无 | 无 | 无 |
| registerButton | 1 | 注册按钮 | Button | 无 | 无 | 无 |
- c) 操作说明
- | 操作名 | 主动性 | 多态性 | 解释 | 约束条件及其他 |
|----------|-----|-----|--------------|-------------------------|
| Register | 无- | 无 | 根据输入信息进行用户注册 | 合法的注册中必须 email 不与已有账号重复 |
- d) 关联
无
- e) 泛化
无
- f) 依赖
依赖 User 类。

1.2.4 FileProcessingInterface 类

- a) 类的整体说明：
类名为 FileProcessingInterface，对应人机交互部分的文件处理界面，负责为用户提供可视化的文件处理操作接口。
- b) 属性说明
- | 属性名 | 多重性 | 解释 | 数据类型 | 聚合关系 | 组合关系 | 关联关系 |
|------------------|-----|--------|-----------|------|------|------|
| fileBox | 1 | 文件上传框 | Box | 无 | 无 | 无 |
| fileUploadButton | 1 | 文件上传按钮 | Button | 无 | 无 | 无 |
| processingItems | 1 | 处理选项 | Table* [] | 无 | 无 | 无 |
| processingButton | 1 | 处理按钮 | Button | 无 | 无 | 无 |
| downloadButton | 1 | 下载按钮 | Button | 无 | 无 | 无 |
- c) 操作说明
- | 操作名 | 主动性 | 多态性 | 解释 | 约束条件及其他 |
|-----------------------|-----|-----|--------|---------------|
| upload | 无- | 无 | 上传文件 | 文件格式必须与处理功能对应 |
| selectProcessingItems | 无 | 无 | 选择处理选项 | 格式必须与处理功能对应 |
| processFile | 无 | 无 | 处理对应文件 | 无 |

downloadFile	无	无	下载对应文件	无
--------------	---	---	--------	---

- d) 关联
无
- e) 泛化
无
- f) 依赖
依赖 User 类。

1.2.5 HomepageInterface 类

- a) 类的整体说明：
类名为 HomepageInterface，对应人机交互部分的个人主页界面，负责为用户提供可视化的修改个人信息和下载历史处理文件操作接口。

- b) 属性说明

属性名	多重性	解释	数据类型	聚合关系	组合关系	关联关系
passwordBox	1	新密码框	Box	无	无	无
phoneNumberBox	1	手机号输入框	Box	无	无	无
modifyButton	1	修改按钮	Button	无	无	无
pageTurnButton	1...10	页面跳转按钮	Button	无	无	无
downloadButton	1	下载按钮	Button	无	无	无

- c) 操作说明

操作名	主动性	多态性	解释	约束条件及其他
modifyProfile	无-	无	修改个人资料	手机号长度限制 11 位
getFileInformation	无	无	获取用户文件信息	无
downloadFile	无	无	下载对应文件	无

- d) 关联
无
- e) 泛化
无
- f) 依赖
依赖 User 类。

1.2.6 MySQLStorage 类

- a) 类的整体说明：
类名为 MySQLStorage，对应数据管理部分的 MySQL 存储器，负责为存取 User 对象、Administrator 对象、File 对象的非文件型属性部分。

- b) 属性说明

属性名	多重	解释	数据类	聚合关	组合关	关联关
-----	----	----	-----	-----	-----	-----

	性		型	系	系	系
databaseConfiguration	1	数据库配置	Dict	无	无	无
administratorTableInformation	1	管理员表结构	List	无	无	无
userTableInformation	1	用户表结构	List	无	无	无
fileTableInformation	1	文件表结构	List	无	无	无

c) 操作说明

操作名	主动性	多态性	解释	约束条件及其他
queryAdministrator	无-	无	查询管理员对象信息	无
addUser	无	无	增加用户对象信息	无
updateUser	无	无	更新用户对象信息	无
queryUser	无	无	查询用户对象信息	无
addFile	无	无	增加文件信息	无
queryFile	无	无	查询文件信息	无
delFile	无	无	删除文件信息	无

d) 关联

无

e) 泛化

无

f) 依赖

分别被 Administrator 类、User 类、File 类依赖。

1.2.7 FileStorage 类

a) 类的整体说明：

类名为 FileStorage，对应数据管理部分的文件系统存储器，负责为存取 File 对象的文件内容部分。

b) 属性说明

属性名	多重性	解释	数据类型	聚合关系	组合关系	关联关系
folderPath	1	文件存储路径	String	无	无	无

c) 操作说明

操作名	主动性	多态性	解释	约束条件及其他
saveFile	无-	无	保存文件	无
delFile	无	无	删除文件	无
queryFile	无	无	查询文件	无

d) 关联

无

e) 泛化

无

f) 依赖

被 File 类依赖。

1.3 新增关联描述（对比 OOA）

无

1.4 新增泛化描述（对比 OOA）

无

1.5 新增依赖描述（对比 OOA）

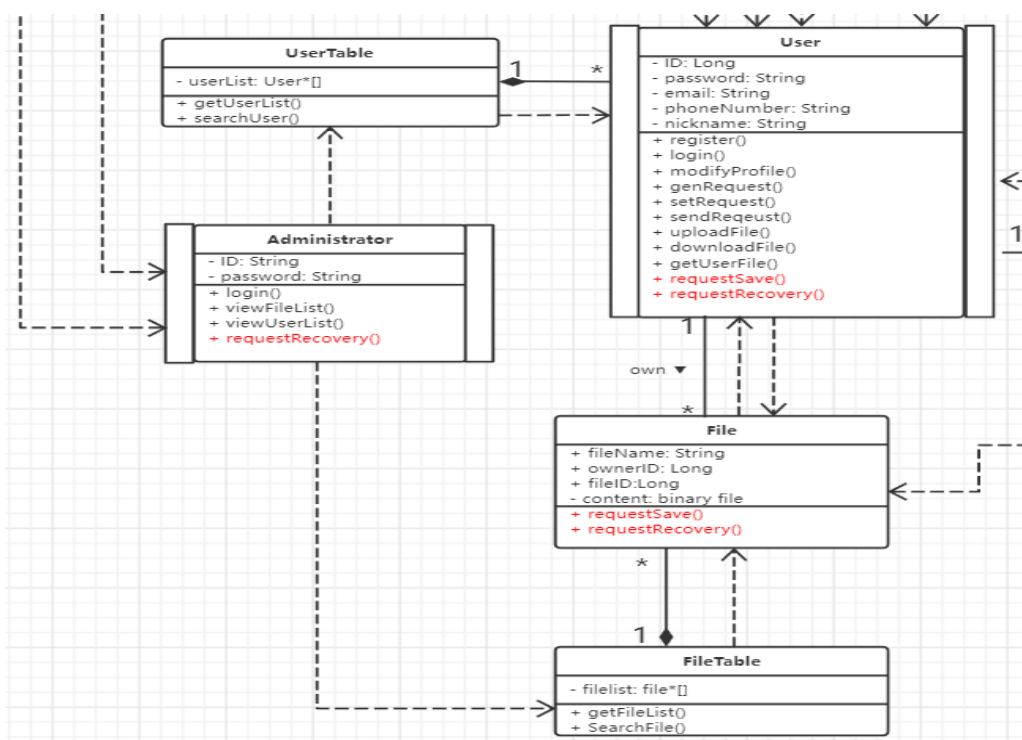
依赖名称	依赖涉及的类名称	依赖类型
管理员界面对象使用管理员对象的操作完成自身提供管理信息的操作	AdministrationInterface/ Administrator	Use
登录界面对象使用管理员对象的操作完成自身登录操作	LoginInterface/ Administrator	Use
登录界面对象使用用户对象的操作完成自身登录操作	LoginInterface/ User	Use
注册界面对象使用用户对象的操作完成自身注册操作	RegisterInterface/ User	Use
文件处理界面对象使用用户对象的操作完成自身上传文件、处理文件等操作	FileProcessingInterface/ User	Use
个人主页界面对象使用用户对象的操作完成自身修改信息、下载文件操作	HomepageInterface/ User	Use
管理员对象使用 MySQL 存储器对象的操作完成自身请求恢复操作	Administrator/ MySQLStorage	Use
用户对象使用 MySQL 存储器对象的操作完成自身请求保存、请求恢复操作	User/ MySQLStorage	Use
文件对象使用 MySQL 存储器对象的操作完成自身除文件类型属性外，其他属性的请求保存、请求恢复操作	File/ MySQLStorage	Use
文件对象使用文件系统存储器对象的操作完成文件内容的存储、恢复操作	File/ FileStorage	Use

1.6 设计说明

1.6.1 问题域部分的设计

从以下五个方面考虑问题域部分设计带来的类图变化：

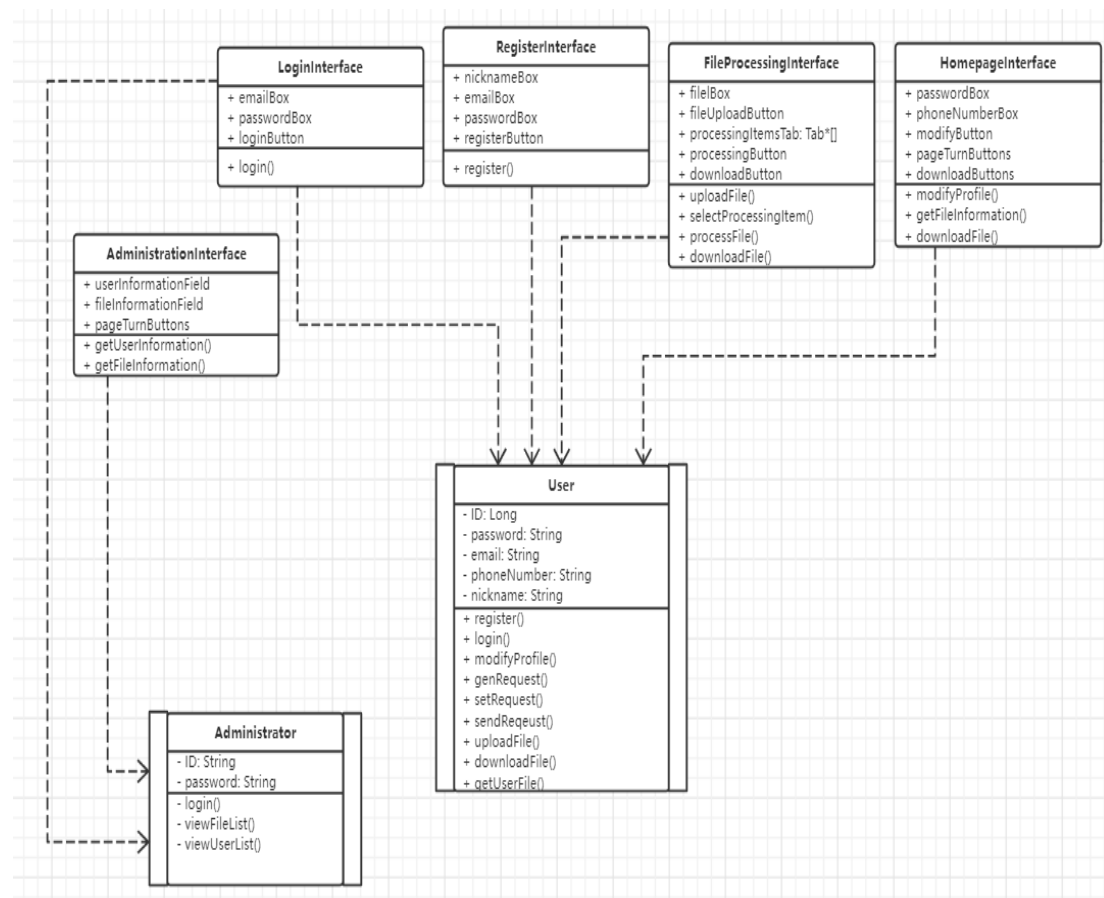
- (1) 编程语言：前端使用 Vue.js，后端使用 Python Flask 框架。编程语言支持类图中用到的单继承特性，同时支持类图中用到的各类数据结构的表达。因此编程语言对问题域部分没有影响。
- (2) 硬件、操作系统、网络设施：本项目使用 Ubuntu 系统、双核 CPU、4G 内存、2M 带宽。由于本项目无需大量的网络通信或是使用大量算力，因此以上各类设施具备支持本系统运行的基本条件。这一部分也对问题域部分没有影响。
- (3) 复用支持：本项目无可用的复用组件支持。
- (4) 数据管理系统：考虑到需要保存的对象中，同时有关系型数据与文件资源，我们同时使用 MySQL 关系型数据库和文件系统作为数据管理系统——使用文件系统管理文件型数据的存取；使用 MySQL 数据库管理用户对象、管理员对象以及文件对象非文件类型属性部分。为需要存取自身的类分别增加请求保存和请求恢复操作，即这里需要对问题域部分作如下修改：



- (5) 界面支持系统：我们项目中使用基于 Vue.js 及相关类库提供的丰富的图形界面支持，无需额外的实现，因此也对问题域部分不产生影响。

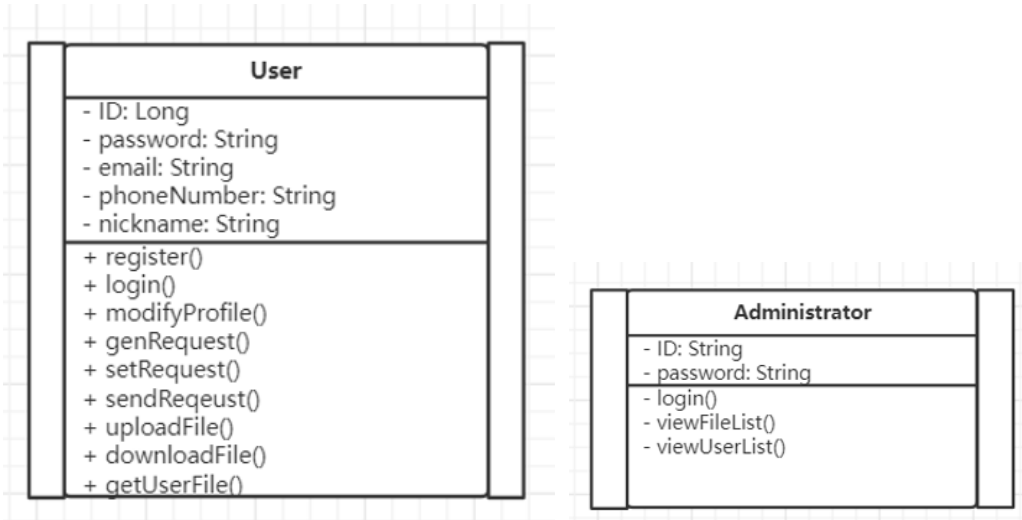
1.6.2 人机交互部分的设计

- 人机交互部分的需求分析
 - 人员活动者：大部分为学生、年轻职员。因此在设计页面风格方面我们尽可能使用年轻化的简约风格，以迎合用户喜好。
 - 从用况中抽取人机交互内容及过程：
 - (1) 用户登录/注册
 - (2) 用户上传文件并处理、下载处理完的文件
 - (3) 用户访问个人主页、下载历史处理文件
 - (4) 管理员管理用户、文件
 - 输入/输出细化、分解并组合命令：明确每一个输入/输出的具体交互类型与形式、并通过组合与分解优化整体命令结构。
- 界面设计准则：简便、一致、容错、及时反馈
 - 简便：界面简约、交互元素尽量清晰、易用
 - 一致：保持各个页面的风格一致，即年轻化的简约风格
 - 容错：处理文件时检查文件扩展名、对用户的其它非法交互操作也具有一定忍耐力
 - 及时反馈：处理文件时返回进度条，避免用户“哑等待”
- 人机界面的面向对象设计
 - 界面支持系统：使用 Vue.js、相关类库以及浏览器等组成的可视化编程环境。
 - 界面元素：窗口、菜单、对话框、选项卡、滚动条、光标、按钮、输入框等。
 - 界面成分的面向对象表达：如下图



1.6.3 控制驱动部分的设计

控制驱动部分由全部的主动类构成。在我们的设计中，只有“用户”类和“管理员”类是主动类。因此控制驱动部分由以下两个类组成：



这两个类的每一个操作都是一个主动操作。因此其中的每一个操作都对应着一个**控制流**，是一个控制流的源头。

1.6.4 数据管理部分的设计

在我们的设计中需要长久存储的对象有：User、Administrator、File。由于 File 对象中的内容属性的类型是文件，在数据管理部分的设计我们同时使用 MySQL 数据库和文件系统作为数据管理系统。其中：

- 使用文件系统存储 File 对象的 content 属性；
- 使用 MySQL 数据库管理 User 对象、Administrator 对象以及 File 对象的其它属性数据。

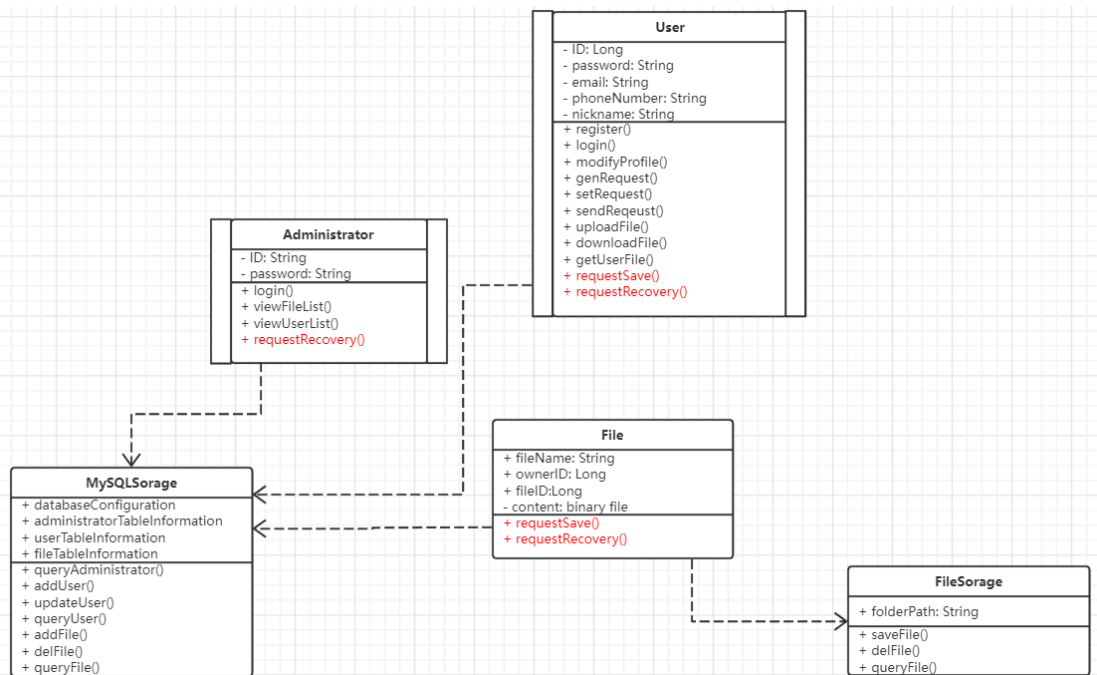
存放策略：

- 文件系统：每个 File 对象的 content 文件在文件系统中单独存为一个文件，并用文件名“{userID} + {fileID} + {fileName}”唯一标识；
- MySQL：数据库表如下

User		Administrator	
id	用户ID	id	管理员ID
password	用户密码	password	管理员密码
email	邮箱		
phoneNumber	手机号		
nickname	昵称		
fileCount	拥有文件数量		
permission	权限		
		File	
		fileName	文件名
		ownerID	所属用户ID
		fileID	文件ID

由 File 表中的“{ownerID} + {fileID} + {fileName}”即可组成 URL，从文件系统中索引出对应的文件。

设计数据管理对象类/问题域部分修改则见下图：

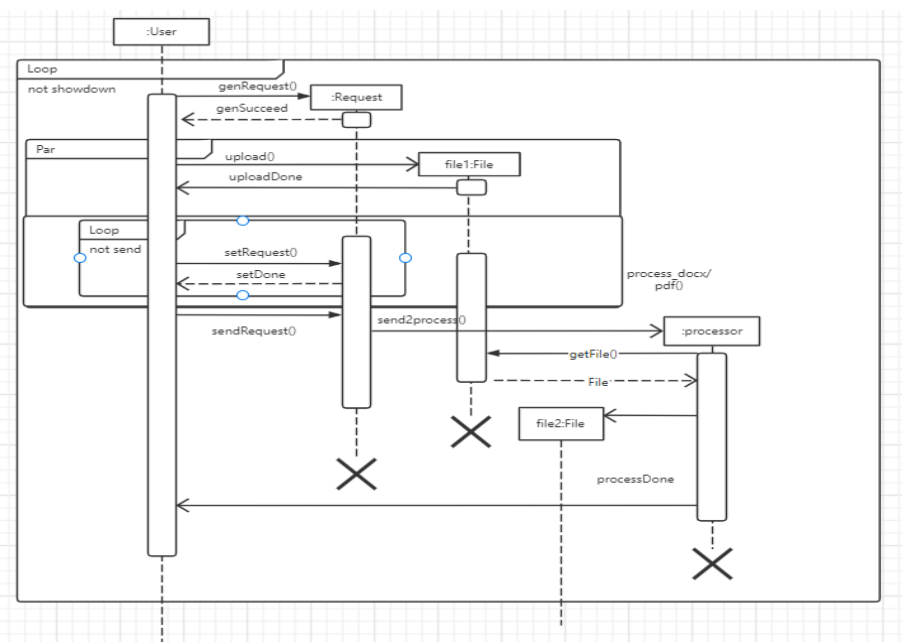


如上图所示增加了两个类：MySQL 存储器与文件系统存储器，并且为需要存、取自身的对象类增加了相应的“请求保存”、“请求恢复”操作。

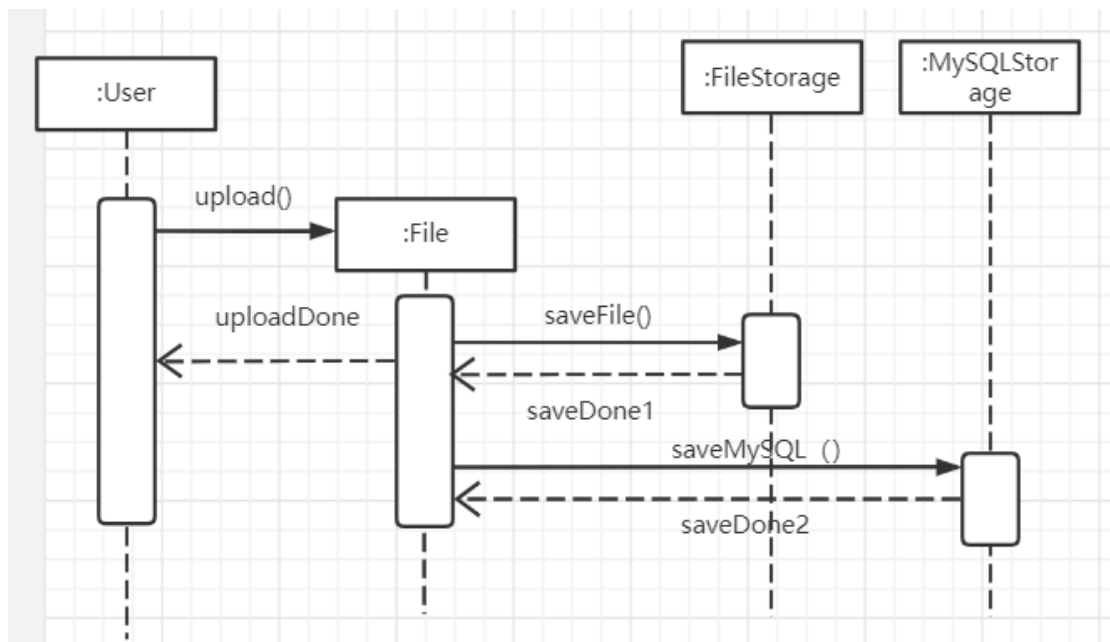
2 顺序图文档

2.1 顺序图综述

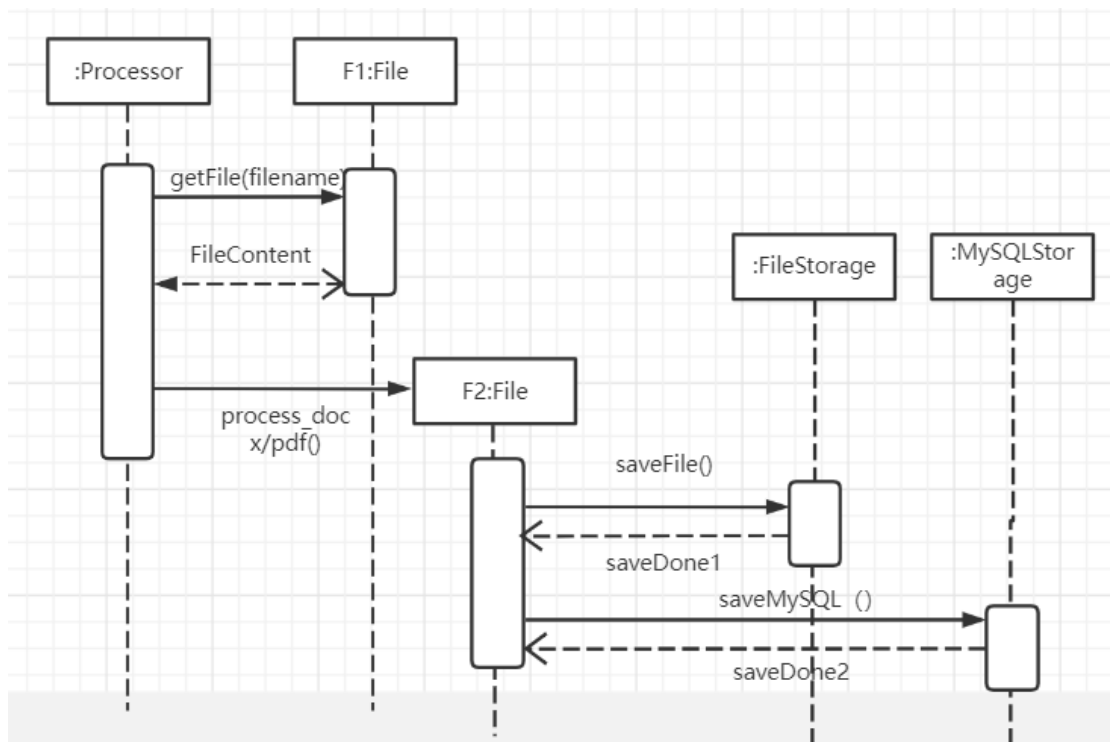
处理文件：（同 OOA 部分）



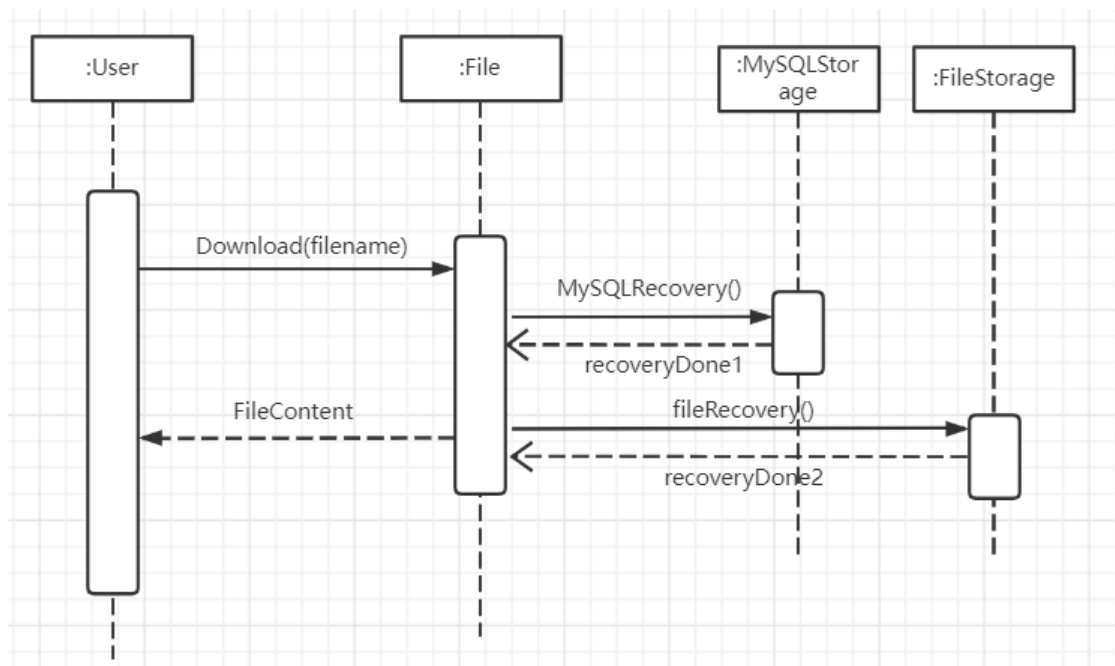
上传文件：



处理后并生成新文件：



下载文件：



2.2 顺序图中的对象与参与者描述（新增）

对象类型	对象名称	是否为主动对象	其他参与对象或参与者有关信息
类	MySQLStorage	否	saveMySQL() 、 MySQLRecovery() 、 saveDone2()、 recoveryDone1()
类	FileStorage	否	saveFile()、 fileRecovery()、 saveDone1()、 recoveryDone2()

2.3 对象接收发送消息的描述（新增）

MySQLStorage:

消息名称	是发送还是接收	消息类型	直接触发的消息的名称列表	是否为自接收消息	消息的发送对象名称	消息的接收对象名称
saveMySQL	接收	同步	[]	否	File	MySQLStorage
MySQLRecovery	接收	同步	[]	否	File	MySQLStorage
saveDone2	发送	异步	无	否	MySQLStorage	File
recoveryDone1	发送	异步	无	否	MySQLStorage	File

FileStorage:

消息名称	是发送还是接收	消息类型	直接触发的消息的名称列表	是否为自接收消息	消息的发送对象名称	消息的接收对象名称
saveMySQL	接收	同步	[]	否	File	FileStorage

	接收	同步	□	否	File	FileStorage
saveDone1	发送	异步	无	否	FileStorage	File
recoveryDone2	发送	异步	无	否	FileStorage	File

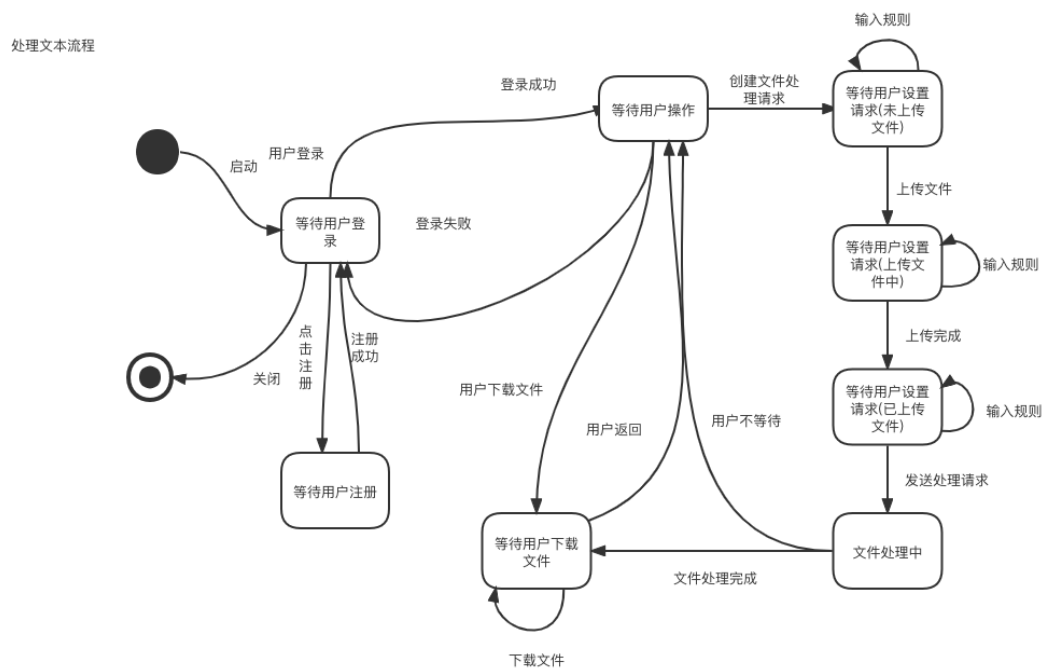
2.4 设计说明

以上顺序图中，上传文件、处理并生成文件、下载文件顺序图是在数据管理部分设计中被添加或修改，其具体原因为：数据管理部分设计中增加了两个存储器类——FileStorage 类与 MySQLStorage 类，并相应地为需要存储的对象类（User 类、Administrator 类、File 类）增加了相应的存取请求操作。因此顺序图中相应的对象、文件存取顺序执行部分需要做出相应的更改。

具体更改细节见顺序图综述中给出的上传文件、处理并生成文件、下载文件三个部分的顺序图。

3 状态图文档

面向对象设计不改变 OOA 部分的状态图，整体系统的状态图如下：



3.1 状态图综述

同 OOA 部分状态图综述。

3.2 状态图的状态描述

同 OOA 部分状态图的状态描述。

3.3 状态图的转换描述

同 OOA 部分状态图的转换描述。

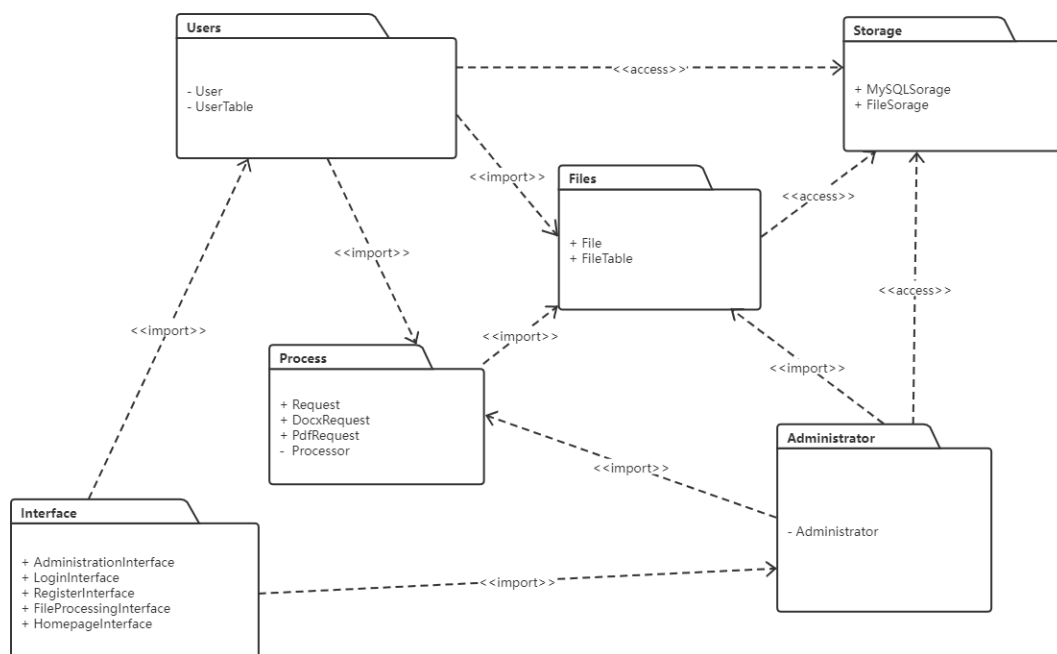
3.4 设计说明

在问题域、人机交互、控制驱动、数据管理这四部分设计中均对该状态图无影响。

4 包图文档

4.1 包图综述

如下图：



4.2 包图中的新增包描述（对比 OOA）

包名称	该包所包含的建模元	相关的其他包
-----	-----------	--------

	素所在的文档	包名称	该包的关系角色
Interface	EasyFile 面向对象设计文档的类图文档	Users	<<import>>目标包
		Administrator	<<import>>目标包
Storage	EasyFile 面向对象设计文档的类图文档	Users	<<access>>源包
		Administrator	<<access>>源包
		Files	<<access>>源包