**基于区块链的安全数据共享系统**

系统设计

（V1.0.0）

瀚文哥哥说：太男了

2019 年 8月

**基于区块链的安全数据共享系统**

系统设计

（V1.0.0）

编 写：侯添久 日期：2019.8.30啊

校 对：侯添久 日期： 啊

标准化： 日期： 啊

审 核： 日期： 啊

批 准： 日期： 啊

**文档修改记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修订人 | 修订说明 | 批准人 | 发布日期 |
| V1.0.0 | 侯添久 | 初稿 | 侯添久 | 2019.8.30 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 引言 5](#_Toc19175673)

[1.1 编写目的 5](#_Toc19175674)

[1.2 背景 5](#_Toc19175675)

[1.3 定义 5](#_Toc19175676)

[1.4 参考资料 5](#_Toc19175677)

[2 总体设计 6](#_Toc19175678)

[2.1 需求规定 6](#_Toc19175679)

[2.2 运行环境 6](#_Toc19175680)

[2.3 基本设计概念和处理流程 6](#_Toc19175681)

[2.3.1 基本概念设计 6](#_Toc19175682)

[2.3.2 处理流程 6](#_Toc19175683)

[2.4 结构 7](#_Toc19175684)

[2.5 功能需求与程序的关系 7](#_Toc19175685)

[3 接口设计 8](#_Toc19175686)

[3.1 用户接口 8](#_Toc19175687)

[3.1.1 用户管理 8](#_Toc19175688)

[3.1.2 文件服务 8](#_Toc19175689)

[3.2 内部接口 8](#_Toc19175690)

[4 运行设计 8](#_Toc19175691)

[4.1 运行模块组合 8](#_Toc19175692)

[4.2 运行控制 9](#_Toc19175693)

[5 系统数据结构设计 9](#_Toc19175694)

[5.1 逻辑结构设计要点 9](#_Toc19175695)

[5.2 物理结构设计要点 10](#_Toc19175696)

# 引言

## 编写目的

对系统整体，从角色角度进行划分，定义每个角色所拥有的基本功能。再对功能进行模块化的划分，让小组成员对于系统有一个宏观上的认识，同时，可以作为系统的开发的指导。定义好系统中需要用到的数据结构，系统架构设计，系统的运行环境，接口的设计，模块间的调用关系等，极大程度上帮助了我们的开发。

## 背景

云平台作为数据外包服务的常见实现手段，可以有效降低企业管理和分析数据的成本，但安全与隐私问题已成为制约其发展的重要因素。用户的敏感数据被存储在云端服务器后会超出云服务使用者的掌控范围，安全性完全依赖云服务提供商，数据安全问题是云服务使用者密切关注的问题。同时，传统的云存储模型以集中存储的方式运行，因此单点故障可能导致崩溃的系统。对于数据安全问题，基于属性的加密(ABE)被认为是一种解决数据隐私和细粒度访问控制问题的重要技术。区块链技术作为一种分散式存储模式，可以解决单点故障问题，具有价格低、吞吐量高等优点。

## 定义

本项目要求实现一个分布式数据存储与共享方案，将IPFS（星际文件系统）、区块链和ABE技术结合起来。数据所有者能够向数据使用者分发密钥，通过指定访问策略对共享数据进行加密，实现细粒度访问控制。基于智能合约的关键字搜索实现对分散存储在IPFS中的密文建立索引，数据使用者能够对加密后的数据进行检索。本课题的创新点在于智能合约的设计。

系统主要包括如下四个功能：用户注册、文件保存、文件上传和文件下载。数据所有者和数据使用者想使用系统功能都需要注册账户。文件拥有者分享文件分为文件上传和文件保存两步。文件上传过程记录文件的基本信息，文件保存包括对文件的加密、将密文上传至区块链等功能。数据使用者申请使用文件需要进行文件下载。

## 参考资料

[1] Shangping W, Yinglong Z, Yaling Z . A Blockchain-Based Framework for Data Sharing with Fine-grained Access Control in Decentralized Storage Systems[J]. IEEE Access, 2018:1-1.

[2] 包括各种ABE相关方案的charm库：<https://github.com/JHUISI/charm>，建议使用其中最简单的方案。

[3] 区块链也可以使用超级账本。

# 总体设计

## 需求规定

提供用户登录入口，需要对用户进行简单身份校验。同时该入口作为系统的首页入口，所有未在登录授权下进行的操作都需强制跳转到登录页面进行用户登录操作。

提供用户注册功能，用户通过填写用户名等必填信息完成账号注册，从而具有登录系统使用系统相关文件操作的基础功能。

通过用户注册时填写的用户名与预留电子邮箱信息，进行密码重置。

用户的工作台界面按照名称字典序展示用户的文件列表，用户能进行文件的上传保存，删除指定文件，指定文件共享到指定用户，下载指定文件到本地。

## 运行环境

开发环境：windows+mac

系统运行环境：Linux

## 基本设计概念和处理流程

### 基本概念设计

系统按功能模块主要划分为三个模块：用户管理模块，文件服务模块，加密检索模块。

1）用户管理：进行用户的注册，登录，密码找回，用户检索功能。

2）文件服务：主要用户对文件的上传和下载等动作。

3）加密检索：对用户上传的文件进行加密，建立索引，解密，分布式存储等功能。

其中用户管理和文件服务模块都是对外提供的服务，而加密检索模块是对文件加密，写入ipfs，解密，以及检索而等在内部进行调用的模块。

### 处理流程

1）用户管理：用户点击相应的登录，注册，找回密码等按钮，前端向后端请求特定的服务，后端接受到请求，通过controller将请求转发给service，service调用持久层对相应的数据表进行增删改查操作。

2）文件服务：用户点击相应的上传和下载按钮请求相应的服务。

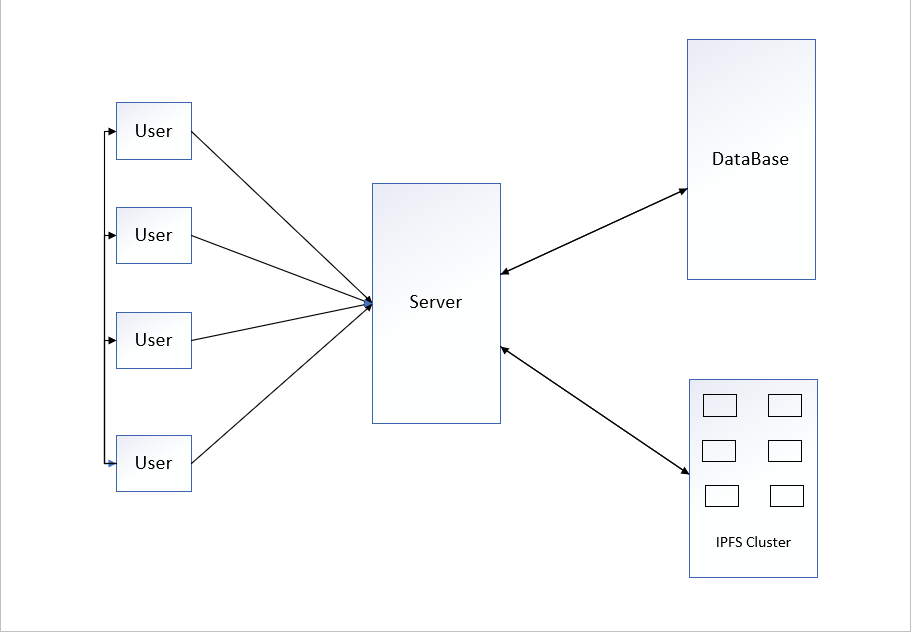
文件上传：用户发送请求，后端响应请求，并接收前端传来的字节流，之后对字节流进行加密，放入ipfs中。

文件下载：用户发送请求，后端响应请求，并去相应的ipfs上拉取数据，对数据进行解密，传送给相应的用户。

3）加密检索：在上传文件到达服务器时，调用该模块进行文件的加密操作，建立索引，并放入ipfs中。在用户下载文件时，拉去ipfs上相应的文件，解密后传送给用户。通时，支持用户对文件进行检索功能。

## 结构

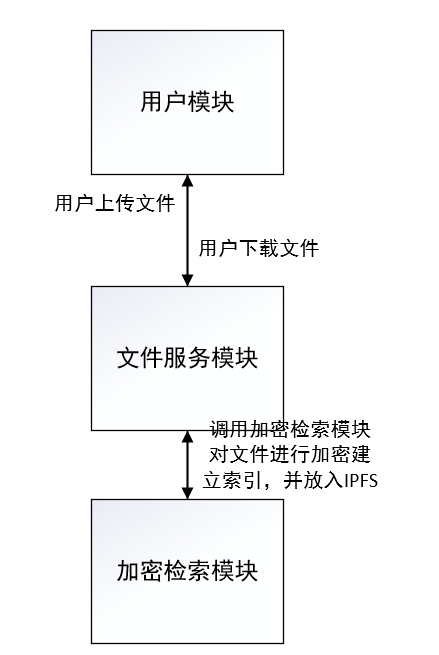
后端整体架构设计：采用springboot+mybatis+mysql的设计，对于加密检索服务另起模块，在service层进行集成调用即可。放置一台服务器作为我们的中转服务器，不存放数据，只是接受用户请求，之后进行相应处理。



## 功能需求与程序的关系

用户管理：用户直接与中转服务器进行交互，中转服务器进行数据库查询等操作。

文件上传：用户直接与中转服务器进行交互，中转服务器不进行文件的存储，只是对文件进行处理操作，放入ipfs。



# 接口设计

## 用户接口

用户接口主要是用户可以与服务器直接进行交互的部分，接口主要涵盖了两大功能模块，一类是用户管理，另一类是文件服务。

### 用户管理

用户登录接口：playcall.cn:8687/qingqingshare/user/user/login

用户注册接口：playcall.cn:8687/qingqingshare/user/regist

用户找回密码接口：playcall.cn:8687/qingqingshare/user/reset

用户检索接口：playcall.cn:8687/qingqingshare/user/query

### 文件服务

文件上传接口：playcall.cn:8687/qingqingshare/file/upload

文件下载接口：playcall.cn:8687/qingqingshare/file/download

文件检索接口：playcall.cn:8687/qingqingshare/file/showfile

文件删除接口：playcall.cn:8687/qingqingshare/file/delete

具体参数参考接口文档。

## 内部接口

内部接口主要是对后端提供服务，整体包括：MD5加密，ABE加密，IPFS，以太坊与智能合约。

MD5加密：通过MD5Util类对密码进行加密，再存入数据库。

ABE加密：对文件进行加密，对hash数值进行加密，对文件和hash数值解密。

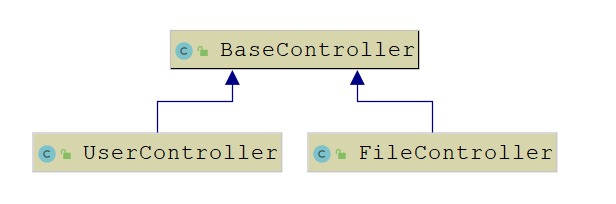
IPFS：通过IPFSUtil将文件从后端写入ipfs集群或从ipfs集群读取文件。

以太坊与智能合约：存储文件的关键字信息，同时对加密后的文件关键字进行检索。

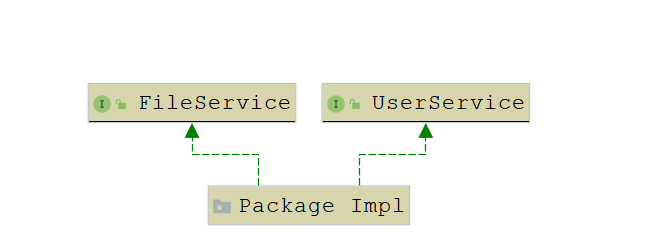
# 运行设计

## 运行模块组合

Controller：



Service：



## 运行控制

对整体BS架构的应用，为了防止使用url越过登录进行访问，我们使用了拦截器，对于越过登录访问的用户会跳转到登陆界面，先进行注册登录。

对于用户的登录，注册，找回密码等功能，用户的请求是先与中转服务器进行交互，再与数据库进行交互存取数据。

对于文件上传，下载等功能，用户请求先是与中转服务器交互，用户上传的文件暂存于中转服务器上，中转服务器再对文件进行加密解密，上传到ipfs等操作。与用户直接进行交互的是中转服务器，而不是后面的集群或者说以太坊等。

# 系统数据结构设计

## 逻辑结构设计要点

整体上是一个三层架构的设计，分别为Controller，Service，Dao，在Service层对文件的操作进行了集成。

在智能合约的设计上，使用了两个Hahmap架构，一个存储文件名与加密后的hash数值的映射，另一个存储文件标签与文件名的映射关系。通过文件名，我们拿到加密后的hash数值，进而从IPFS集群上拿到文件，在对文件进行搜索时，输入标签，通过hashmap拿到对应的该标签对应的文件列表，类似于一个倒排索引的结构。

## 物理结构设计要点

对于区块链中的存储，只是简单的存储字符串，我们使用序列化的方式将hashmap保存为一个文件，读入文件，此时单纯的将文件内容当作字符串进行操作。在进行查找文件时，从区块链上拿到对应的文件，进行反序列化的操作即可还原hashmap，进行拿到hash数值，拿到对应的文件。所以，这样对于区块链上的设计，就极大的简化了。