简单说一下我的计划

首先是修复迭代一和迭代二中出现的bug

后端及数据库部分：

1. 原有数据库整体修改优化
2. 增加用户表 users

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 含义 |
| id | char(19) NOT NULL | 用户id |
| mobile | varchar(11) DEFAULT '' | 手机号 |
| password | varchar(255) DEFAULT NULL | 密码 |
| nickname | varchar(50) DEFAULT NULL | 用户名 |
| avatar | varchar(255) DEFAULT NULL | 头像url |
| sign | varchar(255) DEFAULT NULL | 用户签名 |
| is\_disabled | tinyint(1) NOT NULL DEFAULT '0' | 是否禁用 |
| is\_deleted | tinyint(1) NOT NULL DEFAULT '0' | 是否删除（逻辑删除） |

最后两个是为了后面做权限管理使用的

1. 用户支持头像自定义，采用阿里云OSS，这里就会新建一个和service\_coin并列的新模块：service\_oss，端口号是8003
2. 用户登录注册模块，新建service\_users模块（手机短信验证被ban了，暂定手机号和密码进行登录注册），端口号8004
3. nginx，反向代理，在多模块的情况下，让不同的api路径分发到不同的api服务器中。就是说让代码知道，调用的api应该到哪个端口去找
4. 使用redis，缓存，讨论哪些东西需要放到缓存中？即将访问量较大且基本不改变的数据放入缓存，以减少查数据库的次数（用的话只需要加个注解就行了，不复杂，但我怀疑这样子服务器内存会爆）
5. 看情况使用nacos（服务注册）、feign（服务调用）、hystrix（熔断器），如果我们需要跨服务调用的话
6. 既然有了用户，那么就需要系统管理员的角色存在，也就是说需要一个后台系统，做权限管理。再加上Spring Security的使用。service\_acl模块，端口号8005

前端部分：

1. 视觉效果优化
2. 代码逻辑优化，迭代二里我没有解决api调用同步异步的问题
3. 登录和注册页面。填写信息时前端判断是否为空，格式是否正确（后端也要写，前端验证所谓防君子不防小人）
4. 用户中心页面
5. 后台系统，这个就可以用最开始我提供的那个前端模版