# 详细设计说明书

## 引言

### 1.1编写目的

本详细设计说明书是针对红星新闻系统而编写。目的是对该项目进行详细设计，在概要设计的基础上进一步明确系统结构，详细地介绍系统的各个模块，为进行后面的实现和测试作准备。本详细设计说明书的预期读者为本项目小组的成员以及对该系统感兴趣，在以后想对系统进行扩展和维护的人员。

### 项目开发人员

|  |  |
| --- | --- |
| 张家赫 | 3220190997 |
| 郁裕磊 | 3220190992 |
| 张博 | 3220190995 |
| 林修远 | 3220190962 |
| 栗新博 | 3220190960 |

## 系统结构

整个系统分为两部分，一部分是基于spring cloud微服务架构的后端，第二部分是基于vue框架的前端。

其中后端分为几部分：

（1）Zuul网关：如果前端要调用后端系统，统一从Zuul网关进入，由Zuul网关转发请求给对应的服务

（2）Eureka组件：各个服务启动时，Eureka Client都会将服务注册到Eureka Server，并且Eureka Client还可以反过来从Eureka Server拉取注册表，从而知道其他服务在哪里。其中client作为底层微服务，直接与数据库进行交互，然后将结果返回上一级的微服务。上一级有两个微服务，ribbon和ribbonadmin，其中ribbon是用户微服务，ribbonadmin是管理员微服务，它们之间相互不存在调用关系，通过路由处理来自前端的请求，然后进入对应的微服务，经过控制器拦截后进入服务层，服务层调用底层微服务也就是client，client向数据库发出请求，并获得数据库中的数据，返回到上一级的微服务，将得到的数据打包成json发送回前端。

## Vue前端设计说明

### **3.1开发工具**

本产品前端工作基于win10环境使用Visual Stdio Code与vue-cil进行开发

### 3.2模块设计

#### 3.2.1登录注册页面

**登录模块**

* 用户登录
* 用户输入用户名及密码，调用接口
* 用户名未找到，提示用户“用户名不存在”
* 用户名找到，但密码不匹配，提示用户“密码输入错误”
* 用户名和密码都匹配，登录成功并跳转到主页，同时将用户名存为cookie
* 加载主页获取cookie
* cookie不存在，跳转到登录页
* cookie存在，显示用户名
* 点击注销，删除cookie并跳转到登录页
* 管理员登录
* 点击管理员登录进入管理员登录界面
* 输入管理员用户名及密码，跳转到管理页

**注册模块**

* 用户输入用户名及密码，调用接口
* 注册成功直接跳转到登录页
* 注册失败刷新界面

4.2.2用户新闻浏览页面：

* 有新闻列表，包含新闻标题、新闻作者及新闻类别
* 点击查看按钮，跳转到新闻界面
* 点击今日新闻logo返回新闻列表界面

4.2.3管理员新闻管理页面：

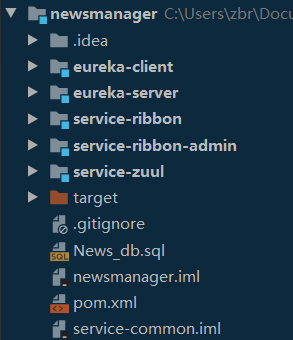
* 管理员用户
* 点击注销返回管理员登录页面
* 点击左侧编辑条目下的新闻管理展开管理页面
* 点击新增增添一条新闻到数据库
* 点击删除从数据库删除指定的新闻
* 点击编辑修改数据库中新闻编号、作者、内容等信息
* 所有内容在修改后实时更新，与数据库间以axios请求进行数据交互

## Springcloud后端设计说明

### 4.1 开发工具

IntelliJ IDEA

### 项目路径说明



后端项目目录下有五个模块，eureka-client、eureka-server、service-ribbon、service-ribbon-admin及service-zuul。News\_db.sql为项目数据库文件，pom.xml是maven配置文件。

### eureka- server模块说明

eureka- server模块作为服务注册中心，提供服务注册和发现。

### 4.4 eureka-client模块说明

eureka-client模块由三部分组成，数据库接口mapper、控制器controller及服务层services，其中mapper包含userMapper和adminMapper两个文件，分别用来处理用户和管理员的数据库相关操作。

userMapper中的函数列表：

Integer login(User user);

User findUserByName(@Param("user\_name") String user\_name);

void regist(User user);

List<News> getnews(@Param("news\_id") int news\_id);

adminMapper中的函数列表：

Integer login(Admin admin);

List<News> getnews();

void insertnews(News news);

int delete(@Param("news\_id") int news\_id);

int update(News news);

控制器controller中包含ReAdminController和ReUserController处理对应的请求，并交给services处理，services调用mapper接口

### service-ribbon、service-ribbon-admin模块说明

service-ribbon模块包含三部分，控制器controller、服务层services及bean类。控制器拦截来自zuul分配的请求，然后交给服务层处理，服务层调用eureka-client模块来进行具体实现。service-ribbon-admin模块与service-ribbon模块流程上基本一致，只是功能上是两个互不关联的微服务。

Bean类中包含四个文件，News，User，Admin，Result，其余三个都是从数据库表中逆向生成的类，Result较为特殊，是将从数据库中获取的数据封装为json后添加到网络请求的返回中。

### 功能模块设计

主要分为两个模块，用户模块为service-ribbon，端口为8764，管理员模块为service-ribbon-admin，端口为8765.

**用户端模块功能及实现流程：**

* 用户注册

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Register.vue | handleSubmit2(ev) |
| 2 | service-ribbon下userController.java | public String regist(User user,HttpServletResponse response) |
| 3 | service-ribbon下userService.java | public Result regist(User user) |
| 4 | eureka-client下ReuserController.java | public Result regist(@RequestBody User user) |
| 5 | eureka-client下ReuserService.java | public Result regist( User user) |
| 6 | eureka-client下userMapper.java | void regist(User user); |

* 用户登录

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Login.vue | handleSubmit2(ev) |
| 2 | service-ribbon下userController.java | public Result login(User user, HttpServletResponse response) |
| 3 | service-ribbon下userService.java | public Result login(User user) |
| 4 | eureka-client下ReuserController.java | public Result login(@RequestBody User user) |
| 5 | eureka-client下ReuserService.java | public Result login(User user) |
| 6 | eureka-client下userMapper.java | Integer login(User user); |

* 用户获取新闻列表

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中View.vue | getUsers() |
| 2 | service-ribbon下userController.java | public Result news\_get(HttpServletResponse response) |
| 3 | service-ribbon下userService.java | public Result getnews() |
| 4 | eureka-client下ReuserController.java | public Result getnews() |
| 5 | eureka-client下ReuserService.java | public Result getnews() |
| 6 | eureka-client下adminMapper.java | List<News> getnews(); |

* 用户查看新闻

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Point.vue | getUsers() |
| 2 | service-ribbon下userController.java | public Result news\_get(int news\_id ,HttpServletResponse response) |
| 3 | service-ribbon下userService.java | public Result getnewsbyid(int news\_id) |
| 4 | eureka-client下ReuserController.java | public Result getnews(@RequestBody int news\_id) |
| 5 | eureka-client下ReuserService.java | public Result getnewsbyid(int news\_id) |
| 6 | eureka-client下adminMapper.java | List<News> getnews(@Param("news\_id") int news\_id); |

**管理员端模块功能及实现流程：**

* 管理员登陆

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中AdminLogin.vue | handleSubmit2(ev) |
| 2 | service-ribbon-admin下adminController.java | public Result login(Admin admin, HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result login(Admin admin) |
| 4 | eureka-client下ReAdminController.java | public Result login(@RequestBody Admin admin) |
| 5 | eureka-client下ReAdminService.java | public Result login(Admin admin) |
| 6 | eureka-client下adminMapper.java | Integer login(Admin admin); |

* 管理员获取新闻列表

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Table.vue | getUsers() |
| 2 | service-ribbon-admin下adminController.java | public Result news\_get(HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result getnews() |
| 4 | eureka-client下ReAdminController.java | public Result getnews() |
| 5 | eureka-client下ReAdminService.java | public Result getnews() |
| 6 | eureka-client下adminMapper.java | List<News> getnews(); |

* 管理员修改新闻

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Table.vue | editSubmit: function () |
| 2 | service-ribbon-admin下adminController.java | public Result news\_update(News news,HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result updatenews(News news) |
| 4 | eureka-client下ReAdminController.java | public Result updatenews(@RequestBody News news) |
| 5 | eureka-client下ReAdminService.java | public Result updatenews(News news) |
| 6 | eureka-client下adminMapper.java | int update(News news); |

* 管理员删除新闻

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Table.vue | handleDel: function ( row) |
| 2 | service-ribbon-admin下adminController.java | public Result news\_delete(int news\_id,HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result deletenews(int news\_id) |
| 4 | eureka-client下ReAdminController.java | public Result deletenews(@RequestBody int news\_id) |
| 5 | eureka-client下ReAdminService.java | public Result deletenews(int news\_id) |
| 6 | eureka-client下adminMapper.java | int delete(@Param("news\_id") int news\_id); |

* 用户及管理员按标题模糊搜索新闻

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Table.vue和View.vue | getUsersbyTitle() |
| 2 | service-ribbon-admin下adminController.java | public Result news\_getbytitle(String news\_title,HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result getnewsbytitle(News news ) |
| 4 | eureka-client下ReAdminController.java | public Result getnewsbytitle(@RequestBody News news) |
| 5 | eureka-client下ReAdminService.java | public Result getnewsbytitle(String news\_title) |
| 6 | eureka-client下adminMapper.java | List<News> getnewsbytitle(String news\_title); |

* 管理员新增新闻

|  |  |  |
| --- | --- | --- |
| 执行步骤 | 文件 | 函数 |
| 1 | Vue中Table.vue | addSubmit: function () |
| 2 | service-ribbon-admin下adminController.java | public Result news\_insert(News news,HttpServletResponse response) |
| 3 | service-ribbon-admin下adminService.java | public Result insertnews(News news) |
| 4 | eureka-client下ReAdminController.java | public Result insertnews(@RequestBody News news) |
| 5 | eureka-client下ReAdminService.java | public Result insertnews(News news) |
| 6 | eureka-client下adminMapper.java | void insertnews(News news); |