



**EMM工作室项目报告**

学 院 计算机学院

专 业 计算机科学与技术

班 级 18 级 3 班

组 别 后台

姓 名 黄宗琪

学 号 3118004775

2019 年 10月

广东工业大学计算机学院制

1/40

# 目录

**项目简介\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*3**

开发工具\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*3

功能简介\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*3

**设计思路\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*4**

数据库的设计\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*4

文件目录\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*4

**详细功能设计\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*5**

注册功能\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*6

登录功能\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*7

获取主表信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

更新用户状态\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

修改密码\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*9

退出登录\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*10

刷新token\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*10

注销功能\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*10

获取个人信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*11

更新个人信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*12

更换头像/群头像/上传文件\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*13

添加好友申请\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*14

删除好友申请\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*15

更新好友申请\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*17

查询一个好友申请\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*17

查询所有申请人信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*17

查询好友信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*18

查询所有好友信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*19

删除好友\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*19

修改好友信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*20

查询用户和好友得到聊天记录\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*22

创建群聊\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*22

删除群聊\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*23

查询群聊消息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*23

删除群聊消息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*23

查询用户在的所有群\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*24

更新群信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*24

添加群成员\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*26

查询群内某个用户信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*27

查询群内所有用户信息\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*28

断网监测\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*28

断网重连\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*28

聊天过程\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*28

文件类型的聊天消息发送\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*28

# 项目简介

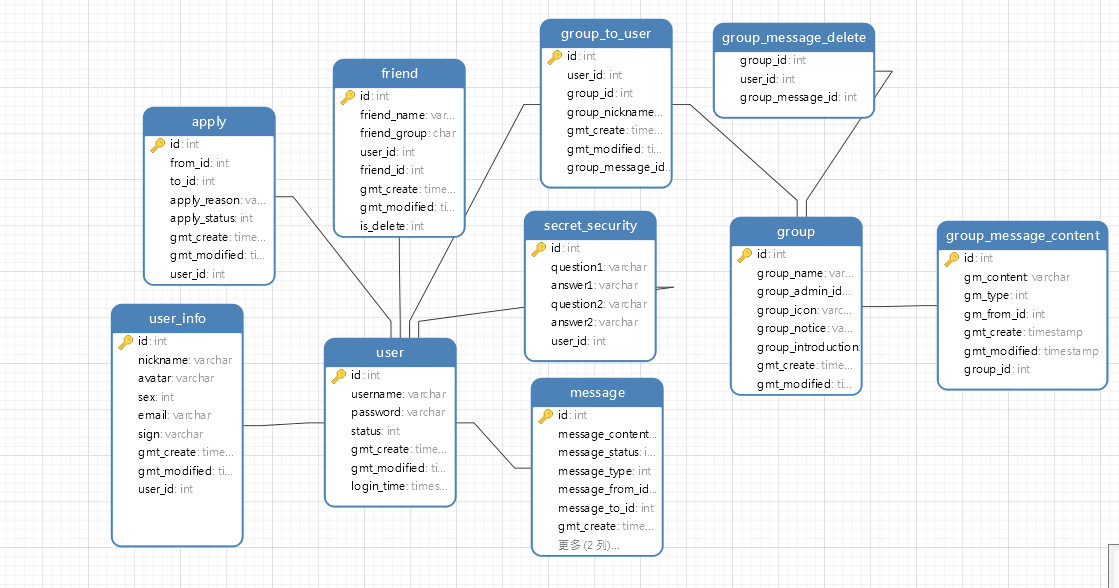
功能简介

该项目是一个基于websocket和ssm框架来实现的仿微信的聊天型项目， 基于redis增加对消息的缓存，提高了效率。用户能进行聊天和个人信息管理，好友的管理以及添加。

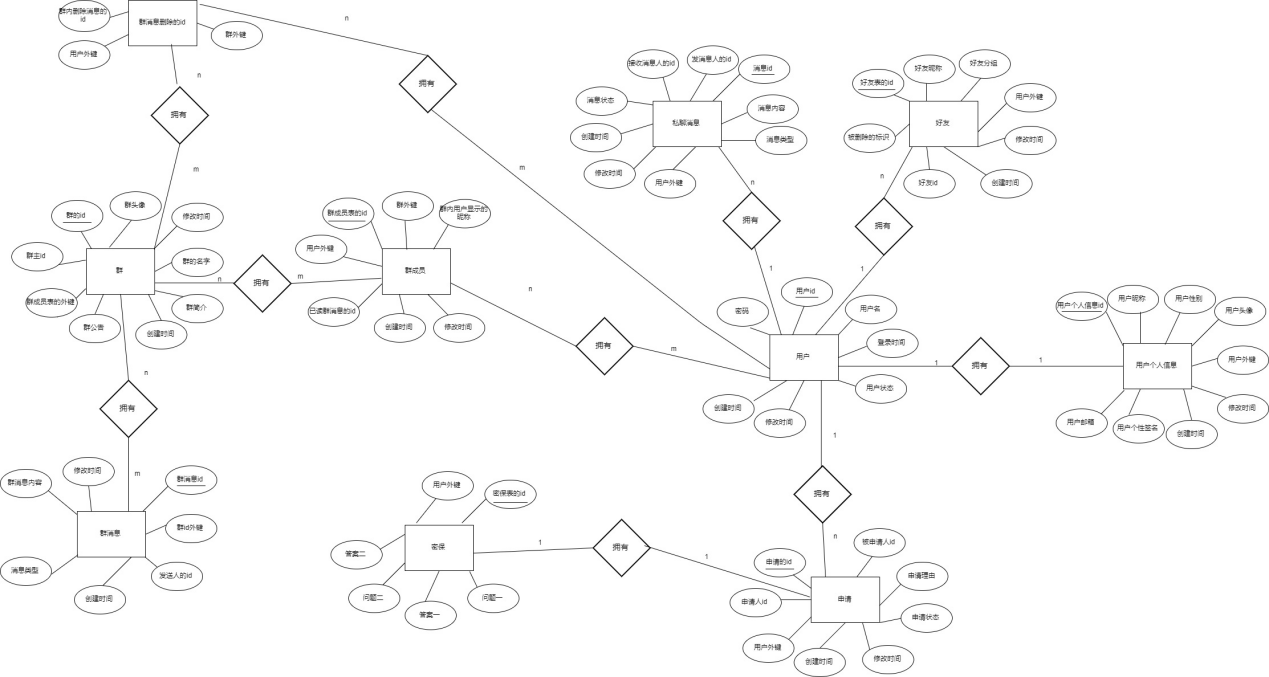
开发工具

| **工具** | **名称** |
| --- | --- |
| 开发工具 | idea 2019.1.1 |
| Web应用服务器 | tomcat 9.0.19 |
| 数据库管理系统 | mysql 7.1 |
| 使用语言：后端 | springmvc+spring+mybatis+redis。 |

数据库的设计



数据库er图



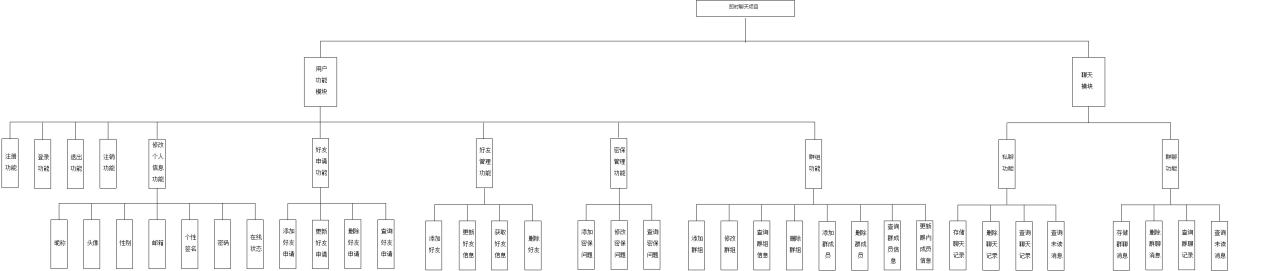
文件目录

| **文件名** | **作用** |
| --- | --- |
| controller | 控制层，分派业务逻辑 |
| service | 业务逻辑层的接口和实现类 |
| dao | 数据层的接口 |
| domain | 项目中使用的实体类（JavaBean）每一个对应数据库中的一张表。 |
| utils | 存放工具类的包，项目中封装的一些较常出现的处理方法。 |
| common | 存放常量类的包，封装了项目中经常使用的常量 |
| config | 存放配置类的包 |
| enums | 存放枚举类的包 |
| execption | 存放异常类的包 |

Github地址：<https://github.com/1092622894hzq/chat>

# 详细功能设计

功能结构图



返回值的统一处理

对于tokoen和sessionId的统一处理，虽然可以设定token和sessionId为定值，让安卓保存，但是保险点利用@ControllerAdvic和ResponseBodyAdvice统一处理返回值，在方法的返回值追加token和sessionId，避免每次重复操作。

接口的返回值都用一个自定义的统一对象进行封装便于统一管理。

统一异常的统一处理

自定义一个CustomGenericException异常，用于自己主动抛出异常和错误信息和错误码，自定义一个统一异常处理类的处理方法GlobalExceptionHandler里面有两个方法，一个捕获

CustomGenericException异常，并把错误信息打印出来的，再转换为通用对象返回给安卓，另一个捕获Execption异常，一切不可预料的异常交给它处理。返回通用对象给安卓。

注册功能

注册功能：

1. controller获取到用户名和密码，调用sevice接口
2. service验证用户名和密码格式是否正确，否，则直接返回结果
3. 验证用户名是否存在，是，则直接返回结果
4. 利用MD5加盐对密码加密，再将用户名和密码存入数据库
5. 同时利用触发器将用户名作为默认值插入到用户个人信息的默认昵称中和将自己添加为好友。



登录功能

登录功能

1 controller 获取用户名和密码，调用service层

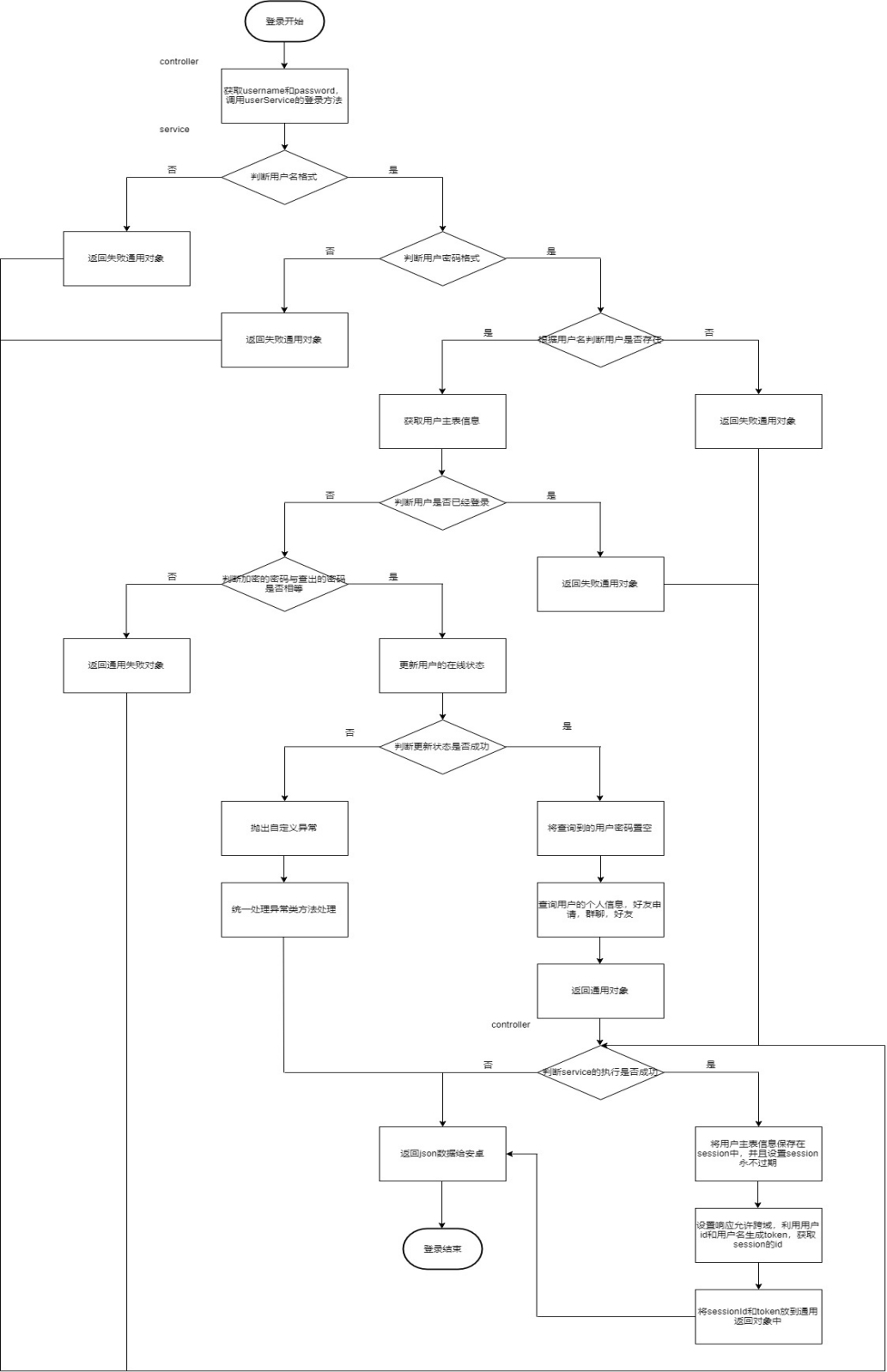
2 service 判断用户名是否存在，否，则直接返回结果

3 根据用户名获取用户信息，将密码置空，避免泄露密码

4 更新用户处于登录状态

5 查询用户所有信息，包括个人信息，好友申请，所有好友，所有群聊

6 返回结果到了controller 将用户信息存到session，同时将sessionId和accessToke存入放到通用返回对象中



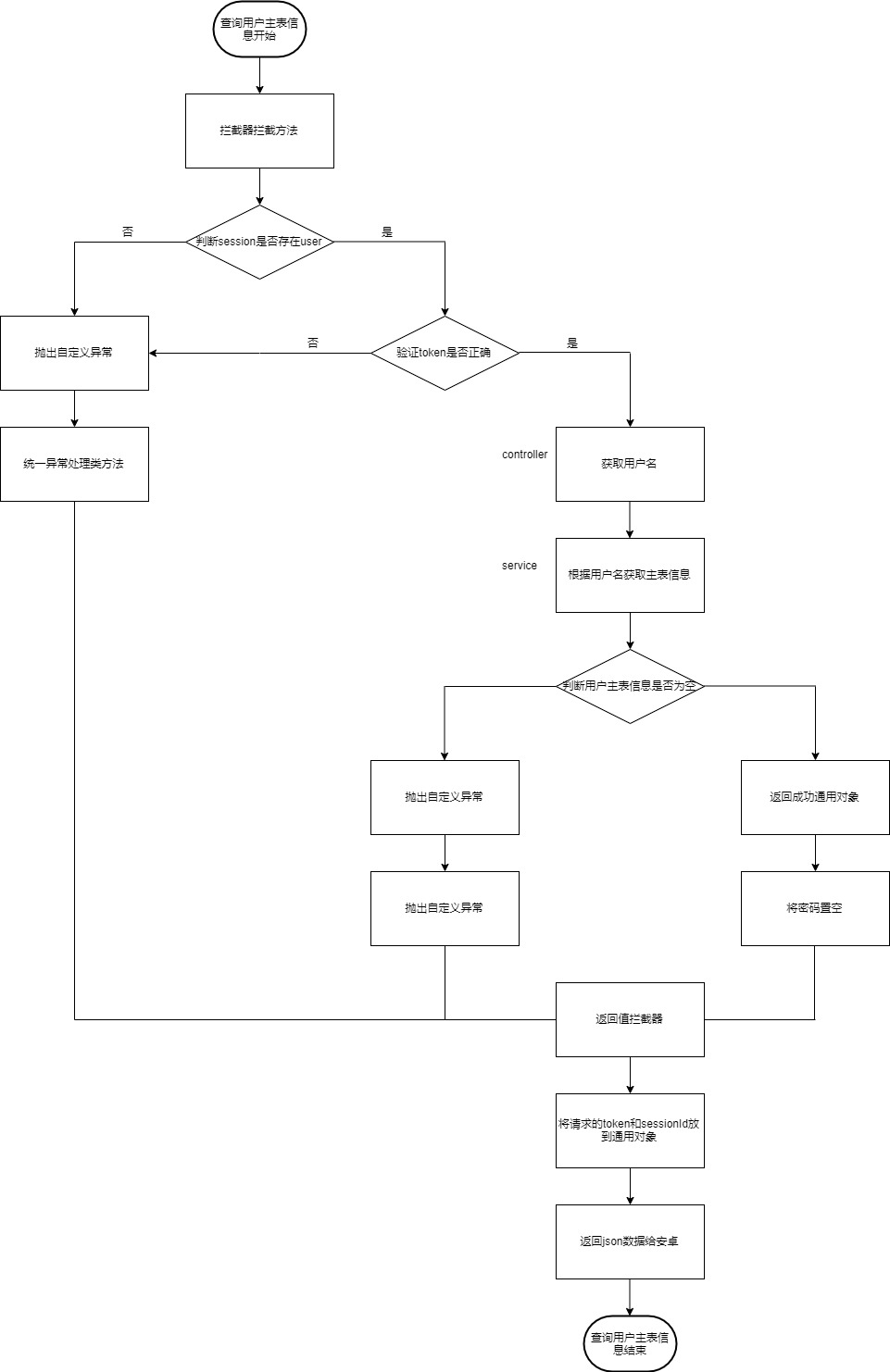
获取主表信息

获取主表信息功能

1 userController 获取用户名，调用userService的select方法

2 userService 根据用户名查询用户主表信息

3 返回结果



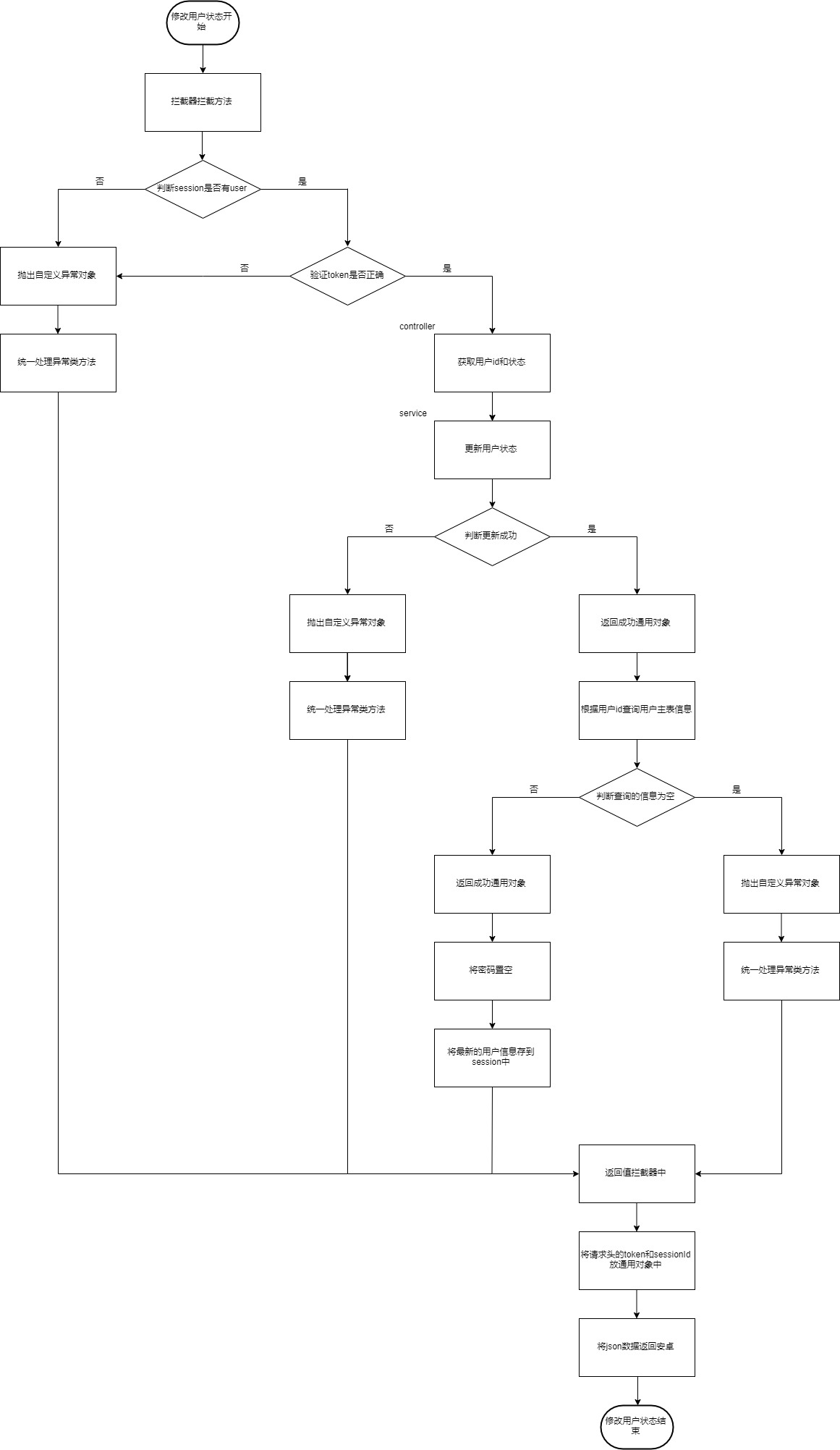
更新用户状态

更新用户状态功能

1 usercontroller 获取用户状态和id，调用userService的更新用户状态方法

2 userService 根据用户id更新用户状态

3 返回结果



修改密码

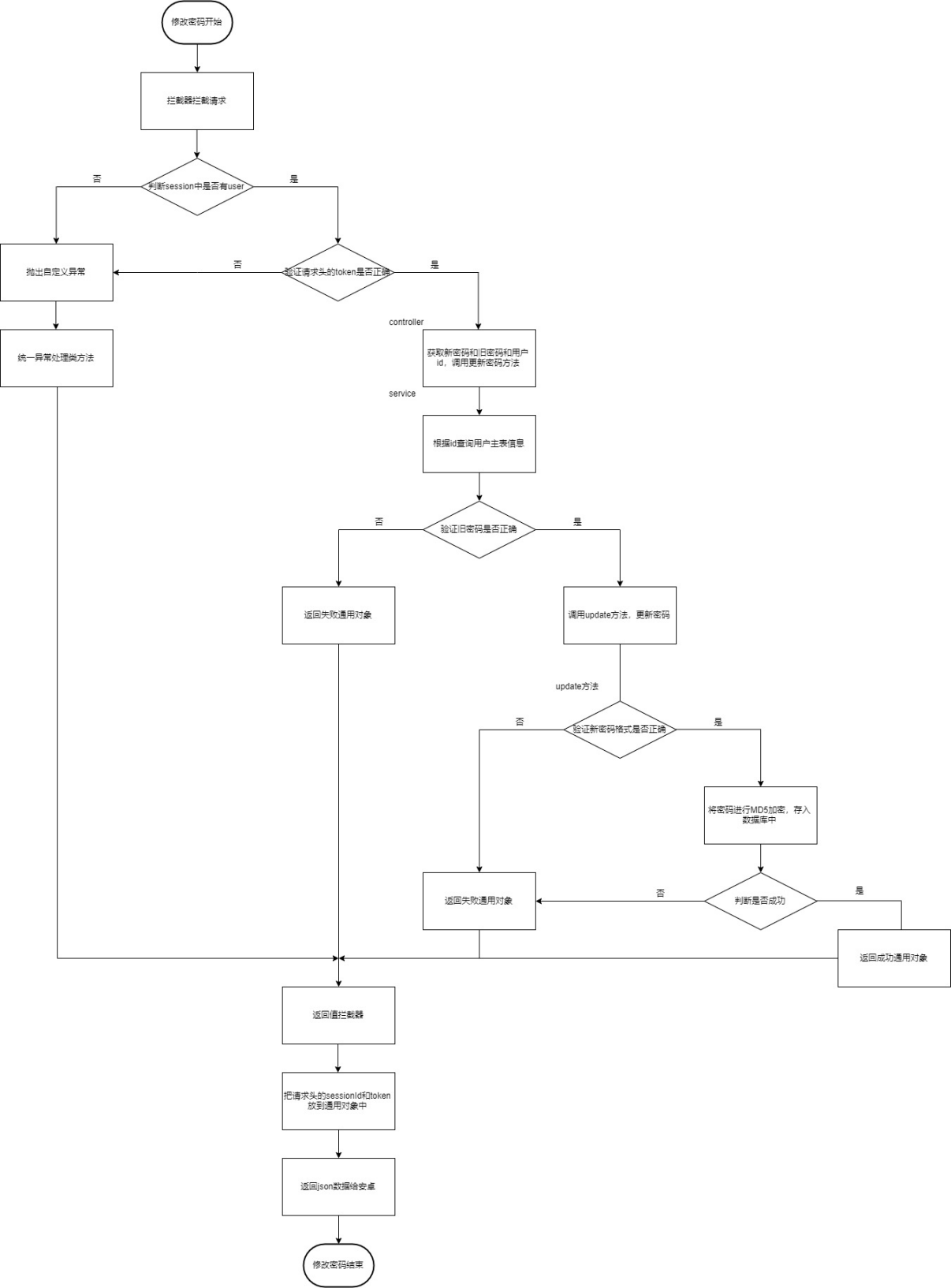
修改密码功能

1 controller 获取新密码和旧密码，在session中获取用户id，调用service层

2 service 根据用户id查询用户密码，判断加盐后的旧密码和数据库的密码是否相等，否，则直接返回结果

3 调用update方法

4 验证新密码格式，正确，将新密码加密，更新密码



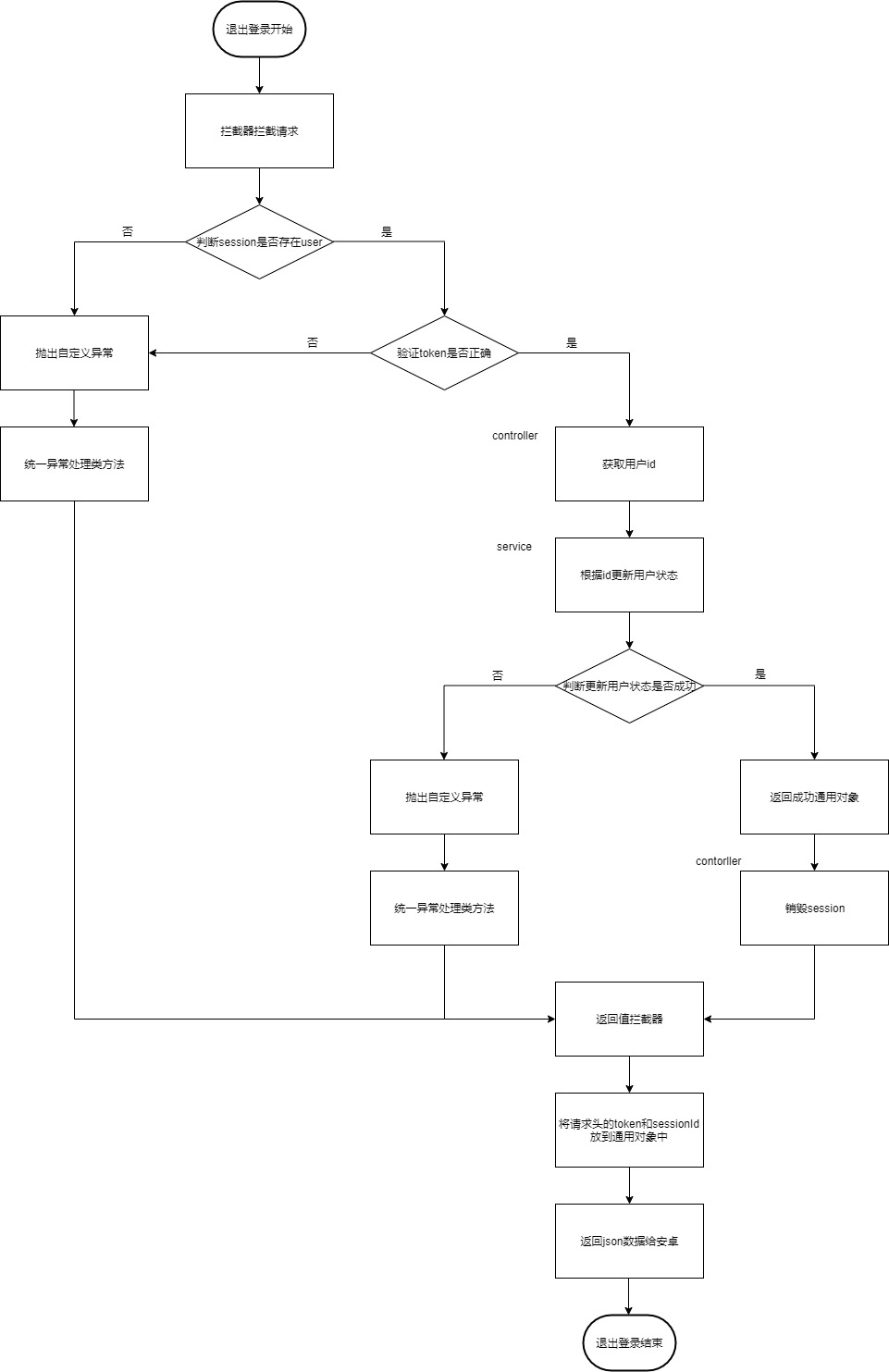
退出登录

退出登录功能

1 controller 获取新密码和旧密码，调用service层

2 service 根据用户id来更新用户状态，改为状态

3 controller 移除session中保存的用户信息，同时把session销毁



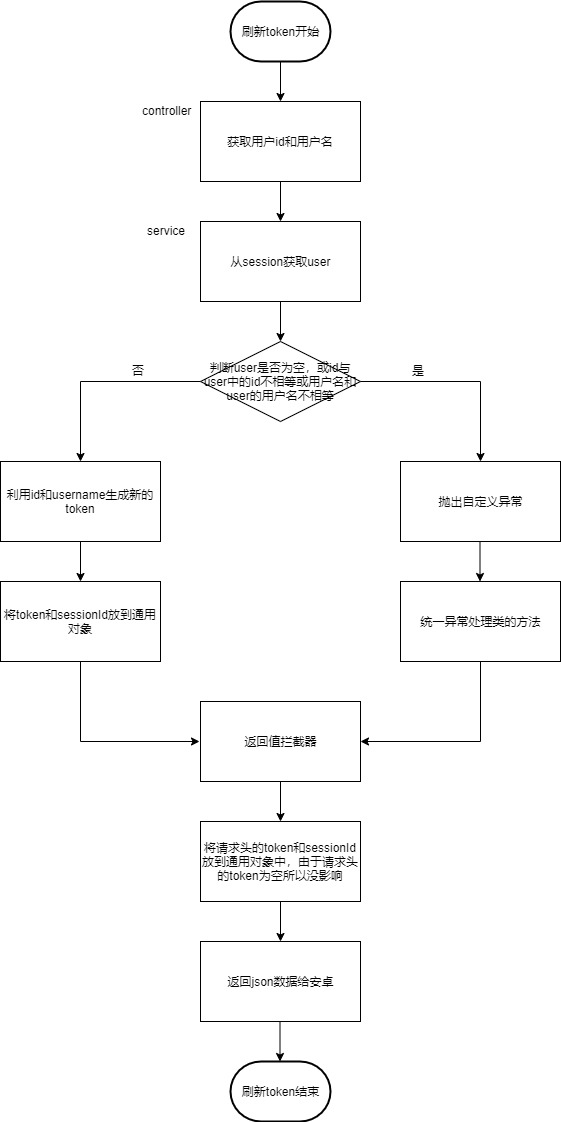
刷新token

刷新token功能

1 controller 获取用户名和用户id，调用service

2 service 获取session中的用户信息跟用户名和id比较判断用户是否在线 否，则直接返回结果

3 利用util包中的工具类和用户名，id生成新的token



注销账号

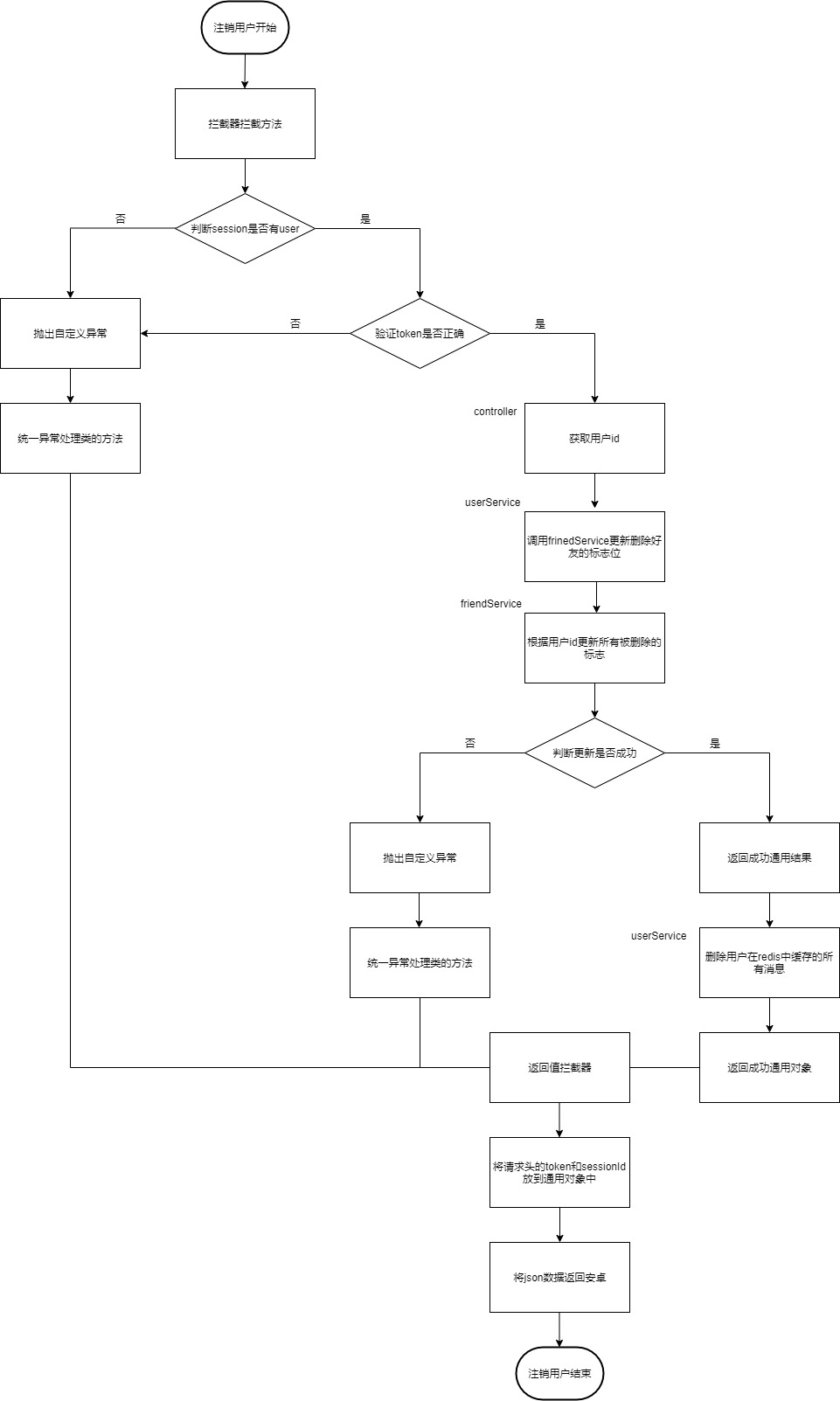
注销账号功能

1 controller 获取用户id，调用service

2 service 删除用户在redis中的缓存，删除用户

3 返回结果到controller，判断是否成功，否，直接返回结果

4 销毁session，返回结果



获取个人信息

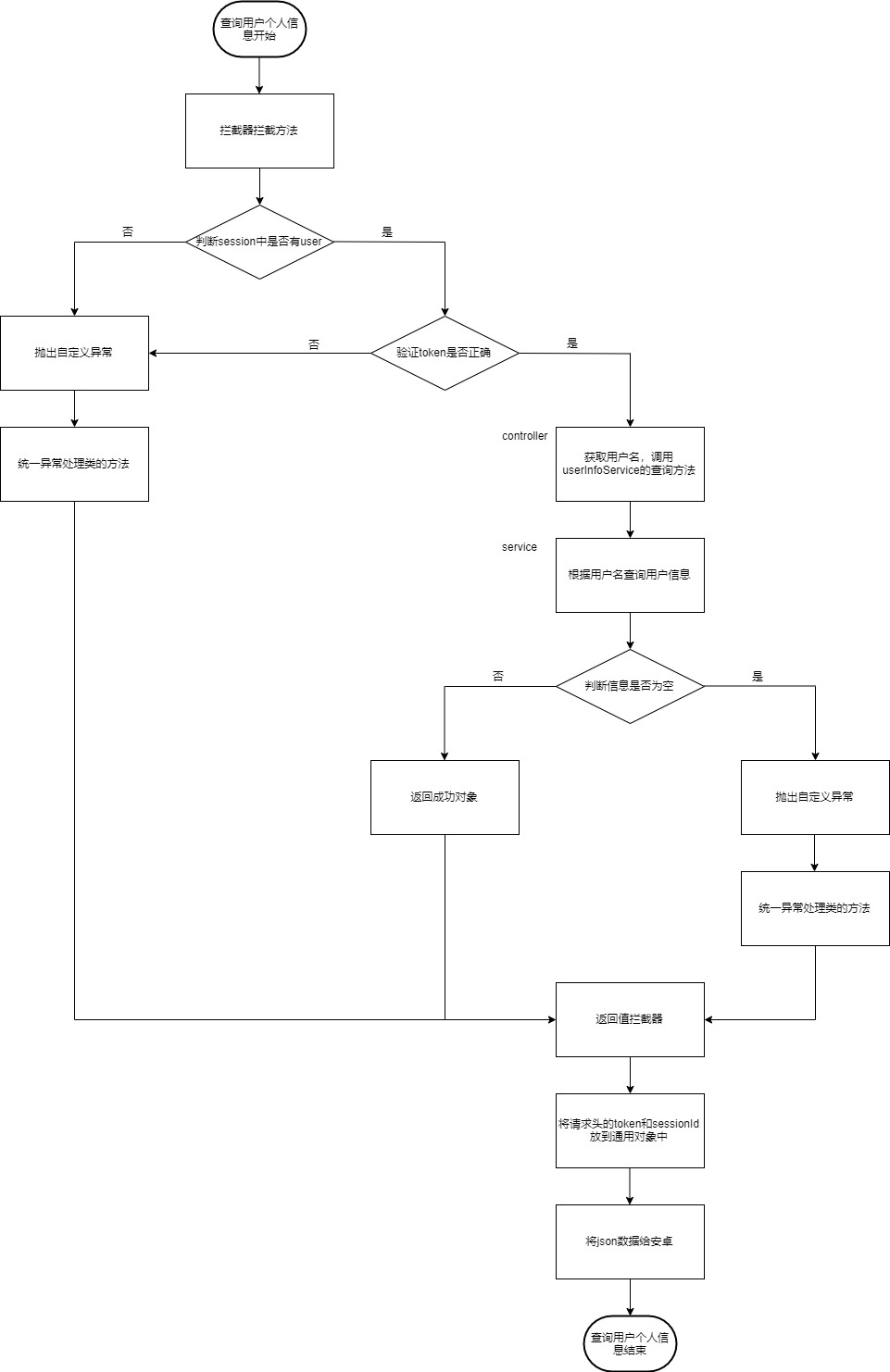
获取个人信息功能

1 controller 获取用户名，调用service

2 service 根据用户名获取用户个人信息

3 判断用户个人信息是否为空，是，抛出异常

4 返回结果



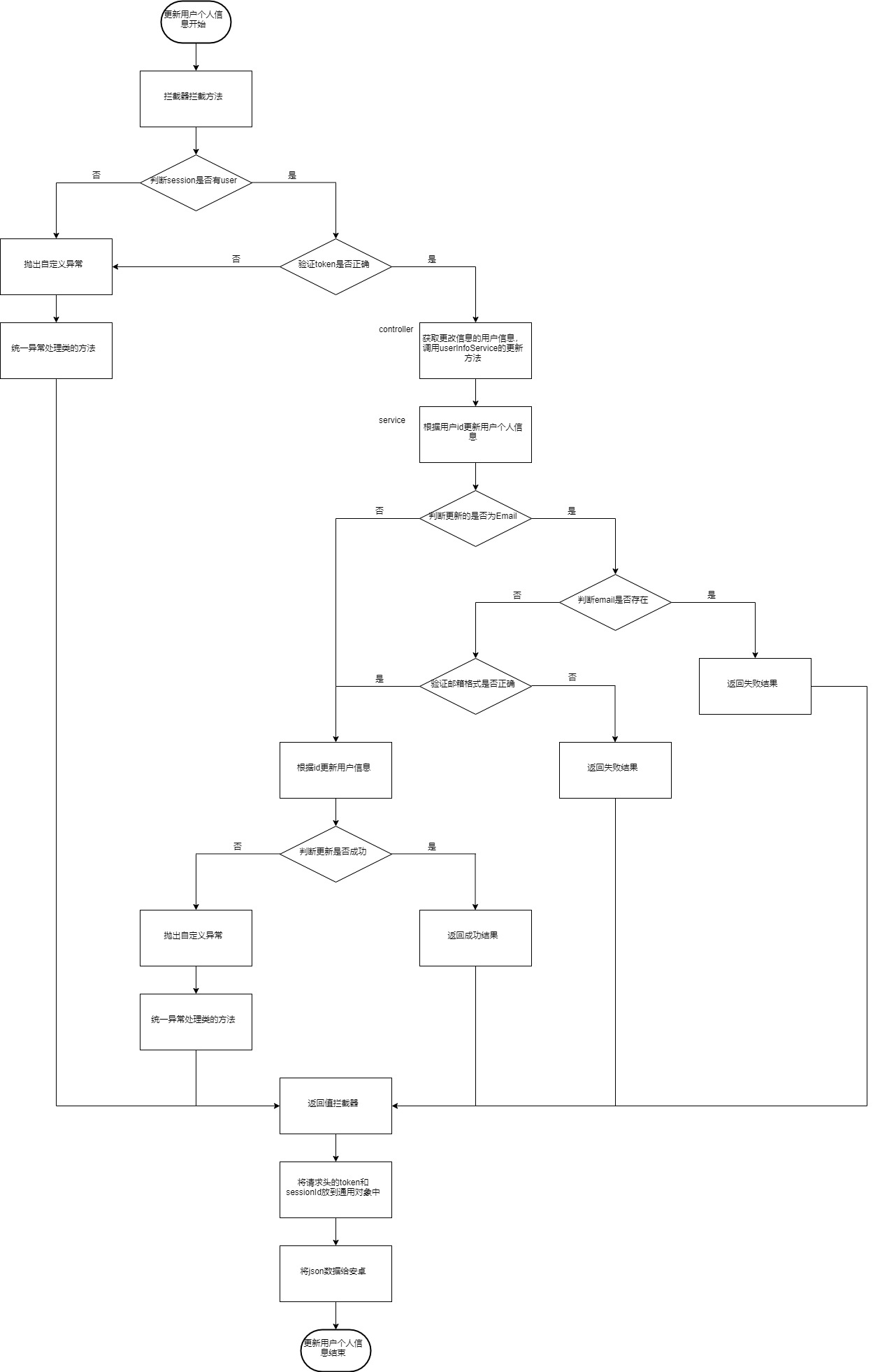
更新个人信息

更新用户个人信息

1 controller 获取存储更新用户信息的对象，调用service

2 service 判断更新得信息是否为email，是，则从数据库中查询判断email是否已经被使用，是，则返回结果

3 更新用户信息，返回结果



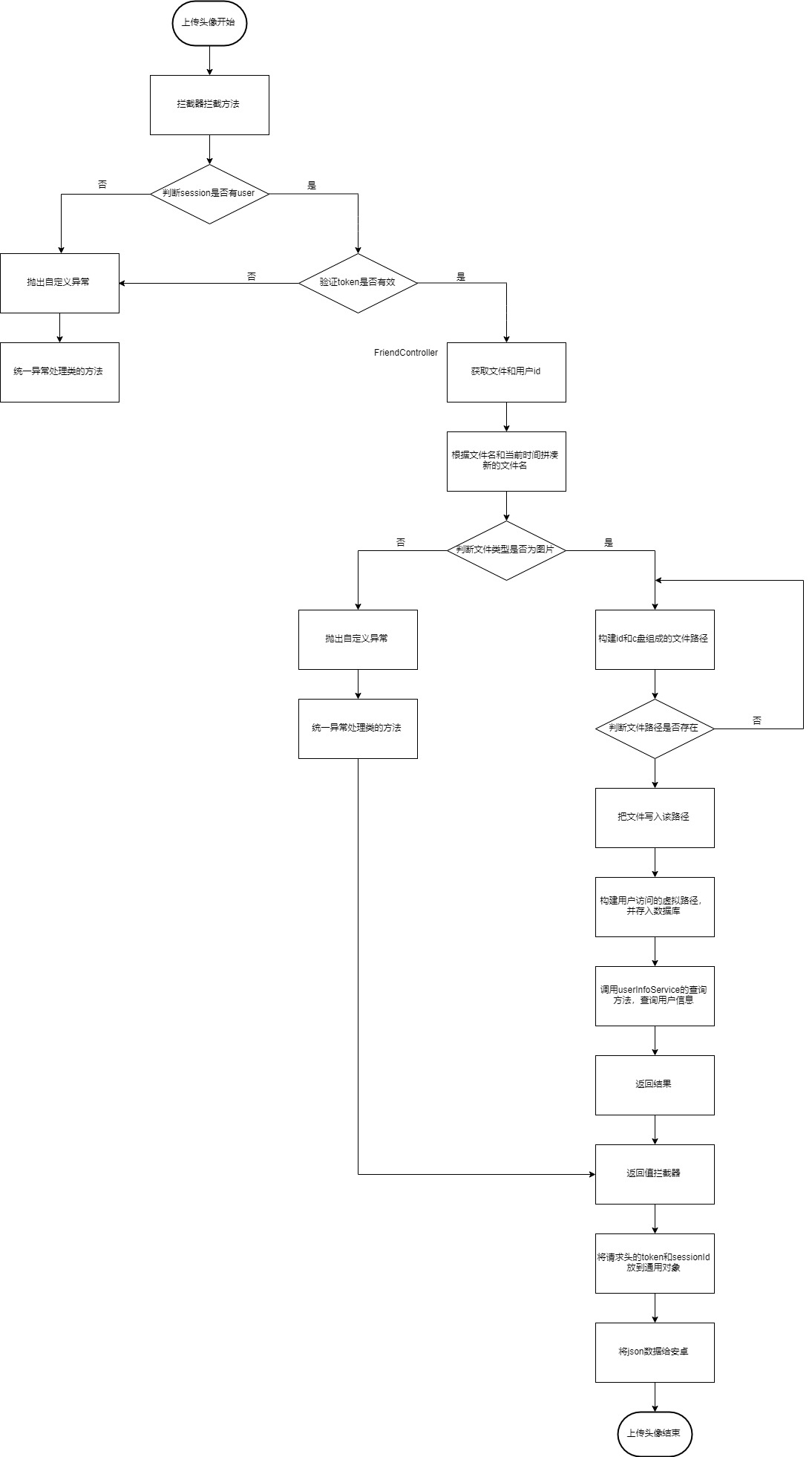
更换头像/群头像/上传文件

1 controller接收到文件，将文件名和当前时间拼接得到新文件名，避免文件名重复

2 利用i/o流写入本地磁盘，利用tomact设置文件得虚拟路径 调用service

3 service把虚拟路径存入到数据库中

3 返回结果



上传群头像和上传文件消息都是差不多操作

添加好友申请

添加好友生气功能

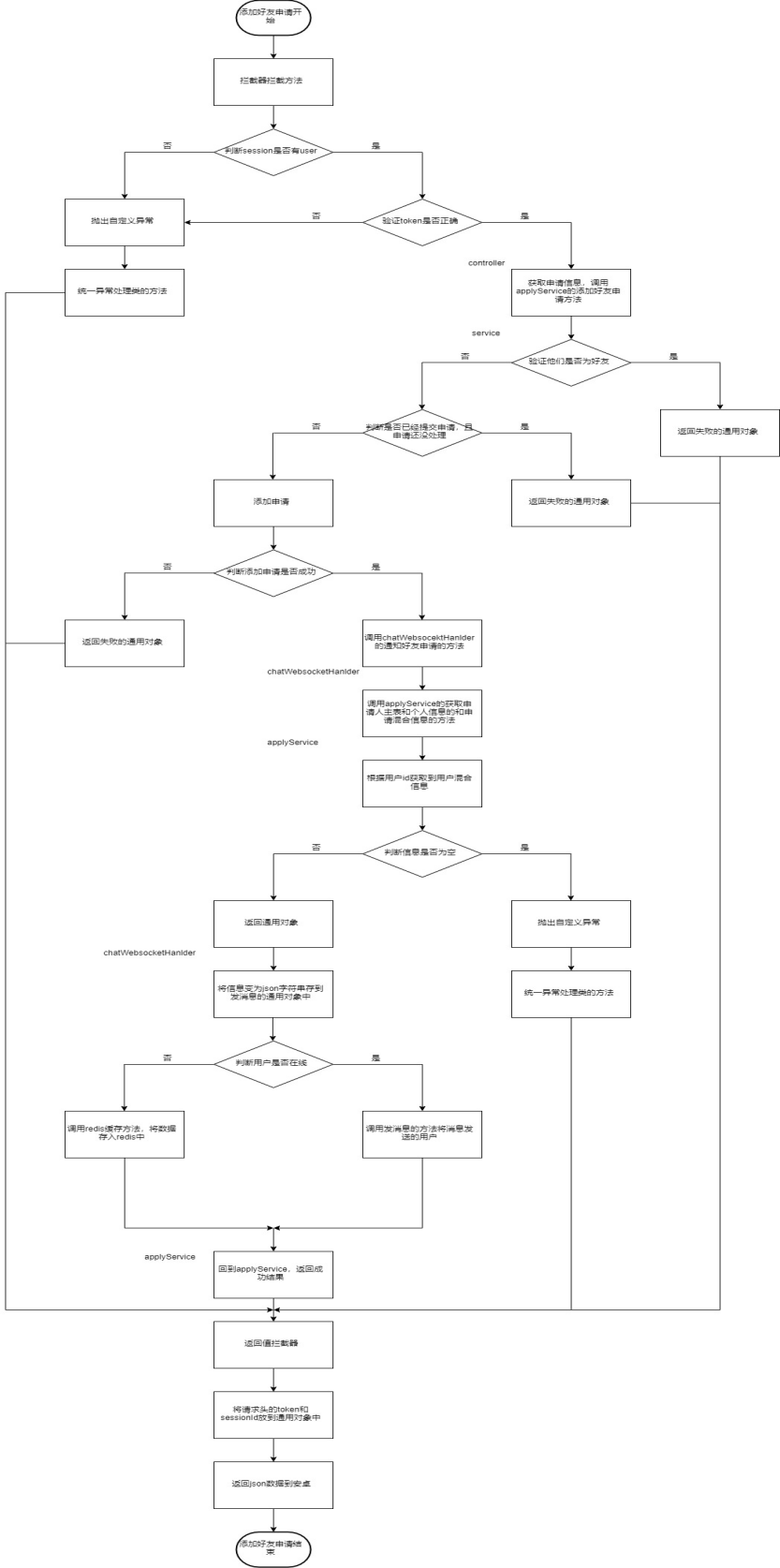
1 controller 获取到申请信息对象，调用service

2 service 根据申请人和被申请人id判断他们是否为好友，是，则直接返回结果

3 判断申请是否已经存在，是，则直接返回结果

4 将申请插入数据库中

5 利用websocket通知用户收到好友申请

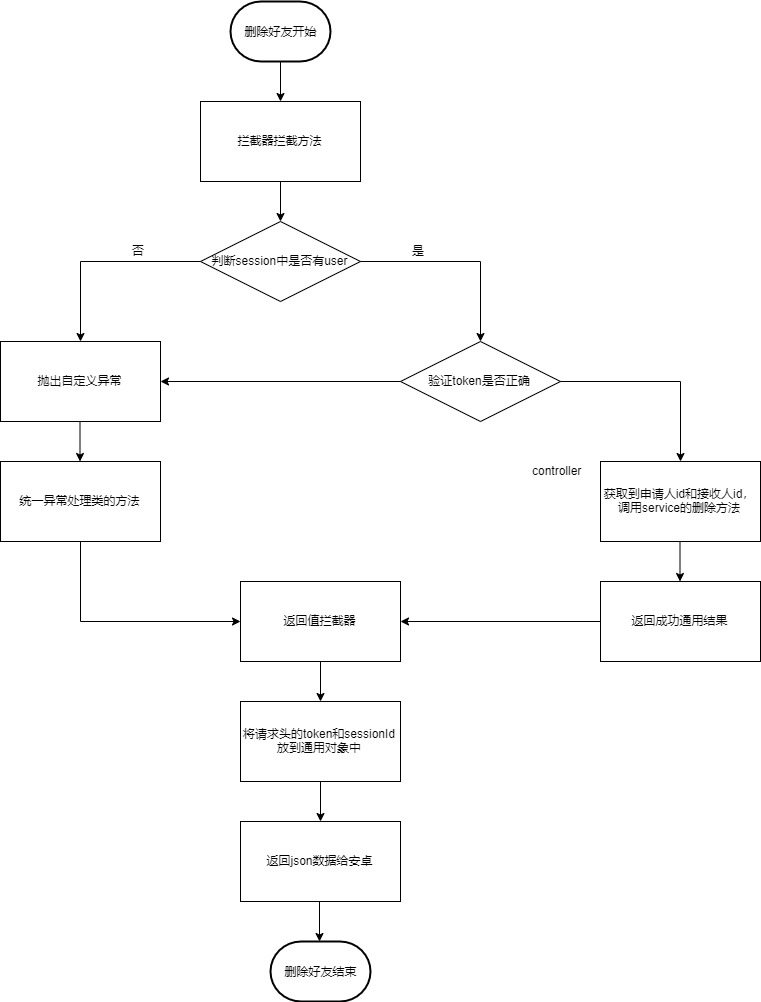


删除好友申请

删除好友功能

1 controller 获取到好友申请人id和被申请人id，调用service

2 service 根据申请人id和被申请人id删除好友申请



更新好友申请

更新好友申请功能

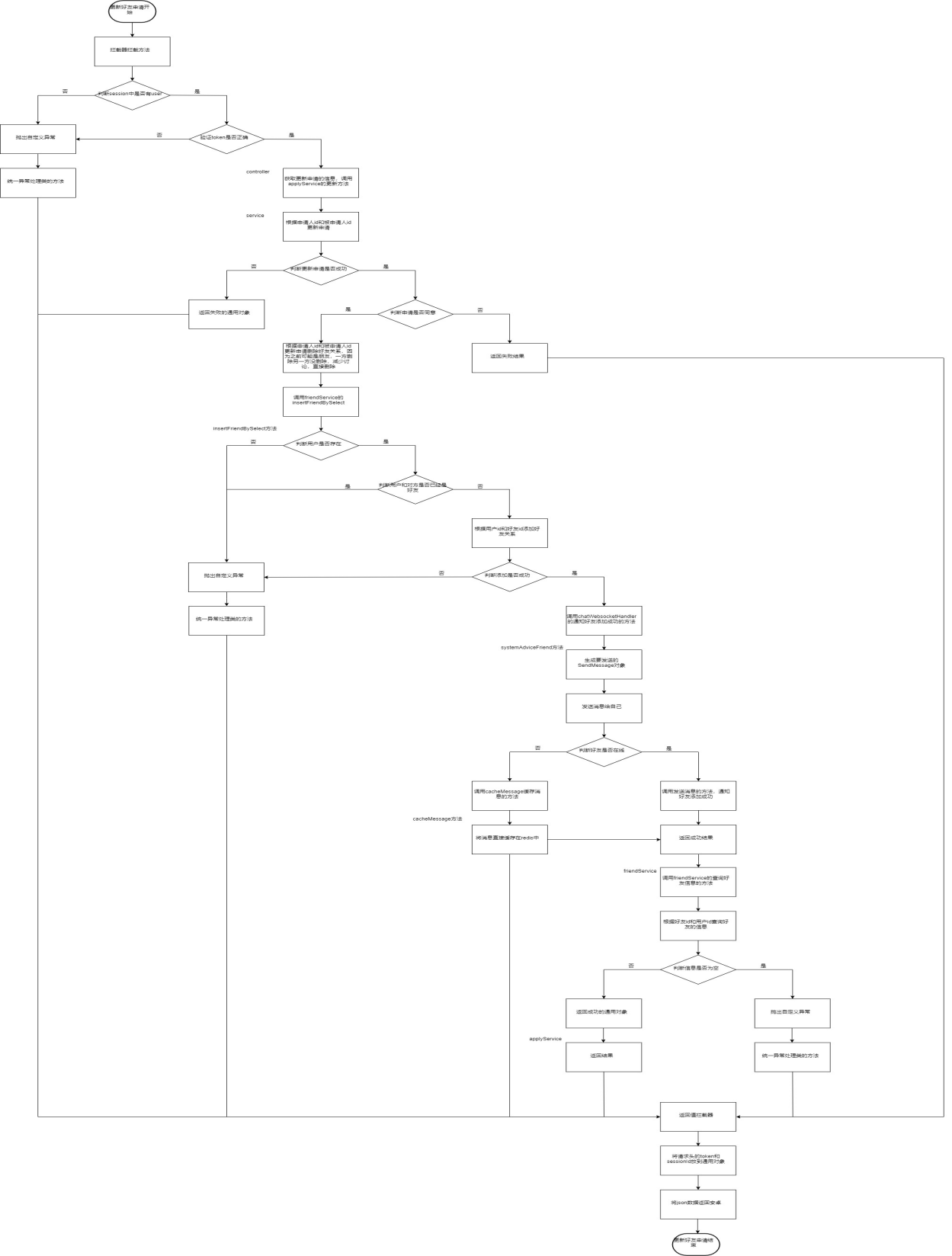
1 controller 获取更新申请的对象，调用service

2 service 更新申请状态，判断是否同意好友申请，否，直接返回结果

3 删除两个人之间的好友关系(之前可能有一方删除，另一方没删除)，重新插入两个 人的好友关系，这里利用mybatis同时执行多条sql，完成双方好友的申请

4 插入成功后，通过websocket来通知用户添加好友成功

5 查询好友信息作为返回结果



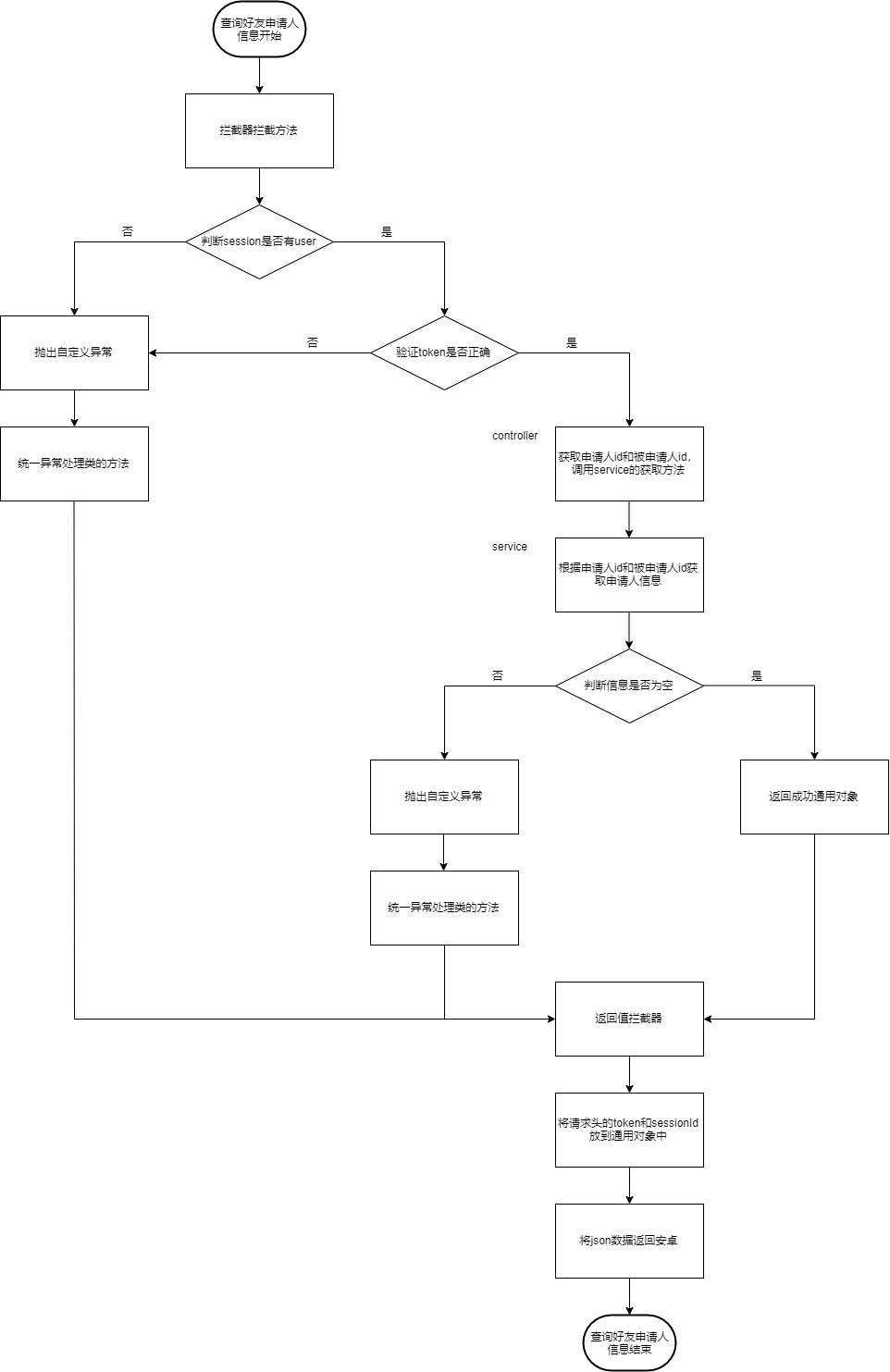
查询一个好友申请

查询好友申请功能

1 controller 获取申请人id和被申请人id，调用service

2 service 根据申请人id和被申请人id进行联合查询获取申请人信息和申请信息

3 返回结果



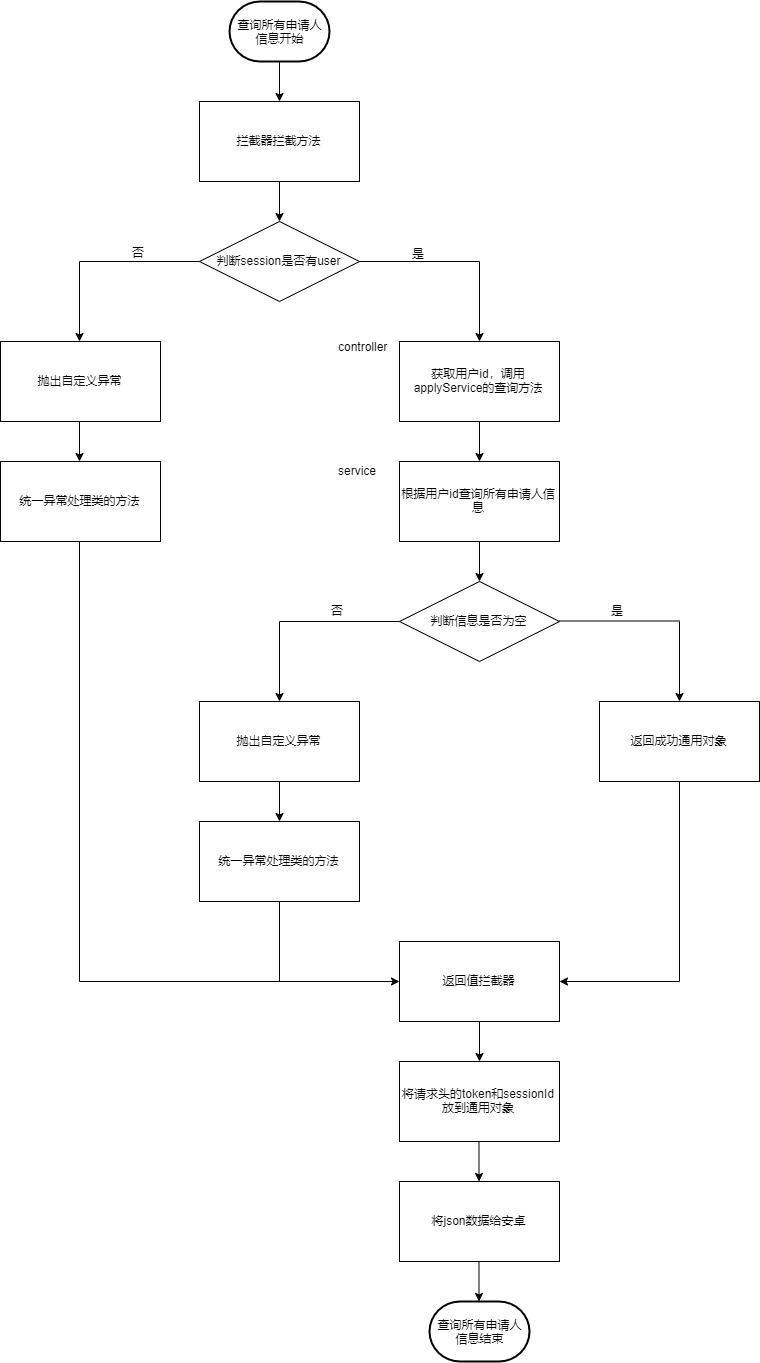
查询所有申请人信息

查询所有好友申请功能

1 controller 获取用户id，调用service

2 service 根据用户id进行联合查询获取申请人信息和申请信息

3 返回结果



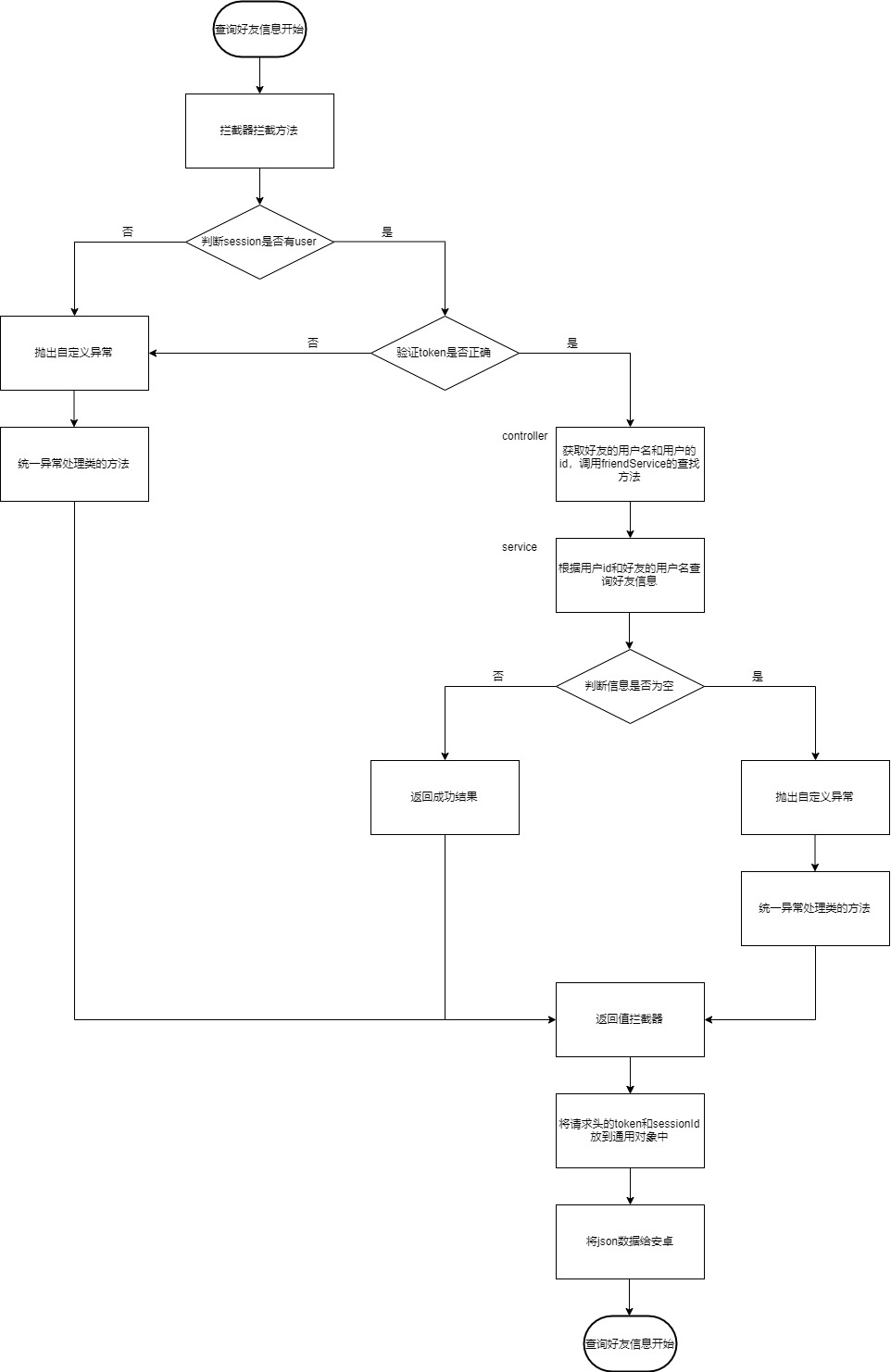
查询好友信息

查询好友信息功能

1 controller 获取好友账号，调用service

2 service 根据联合查询获取用户账号信息和个人信息

3 返回结果



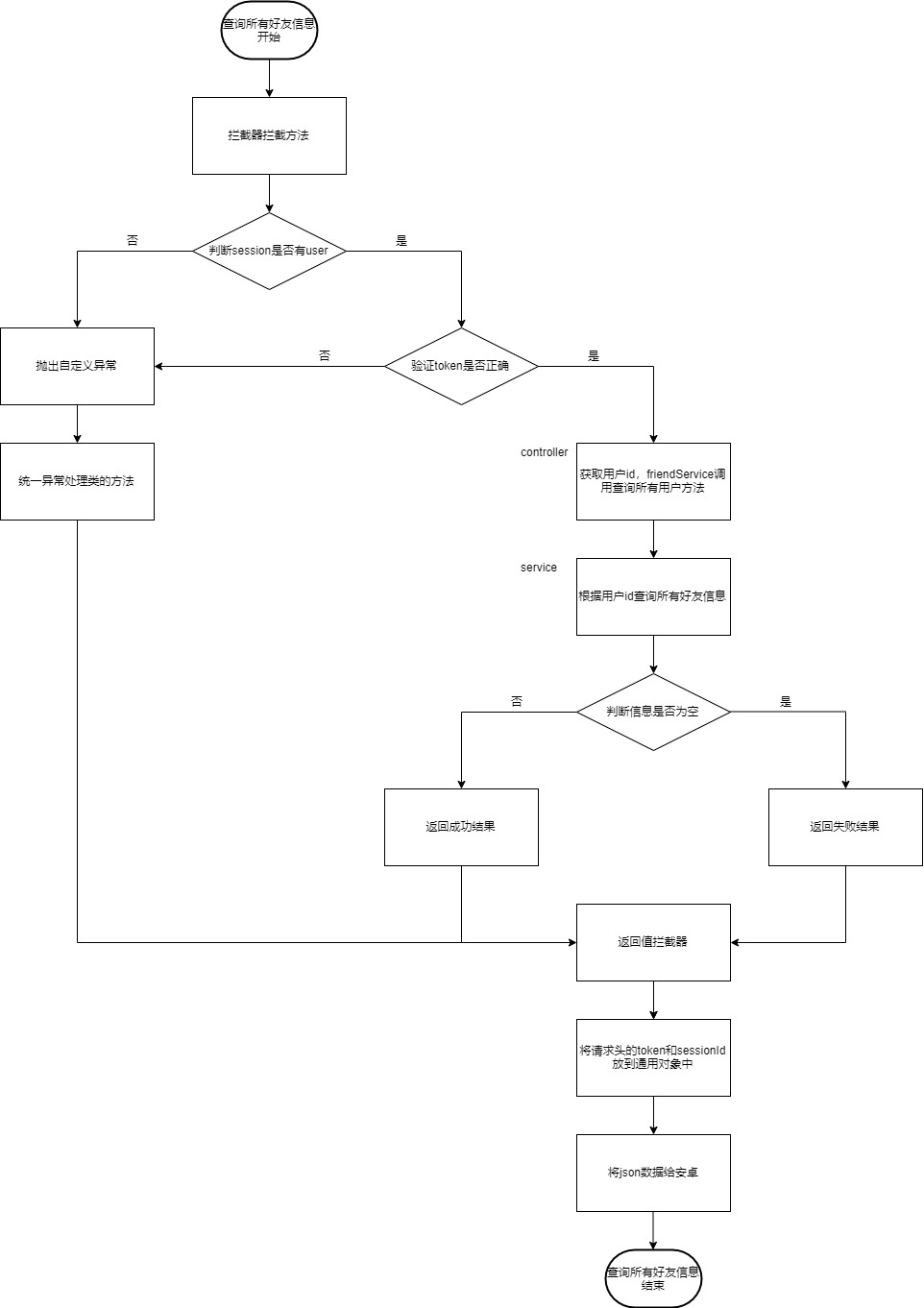
查询全部好友信息

查询全部好友信息功能

1 controller 获取用户id，调用service

2 service 通过联合查询获取所有好友账号信息和个人信息

3 返回结果



删除好友

删除好友功能

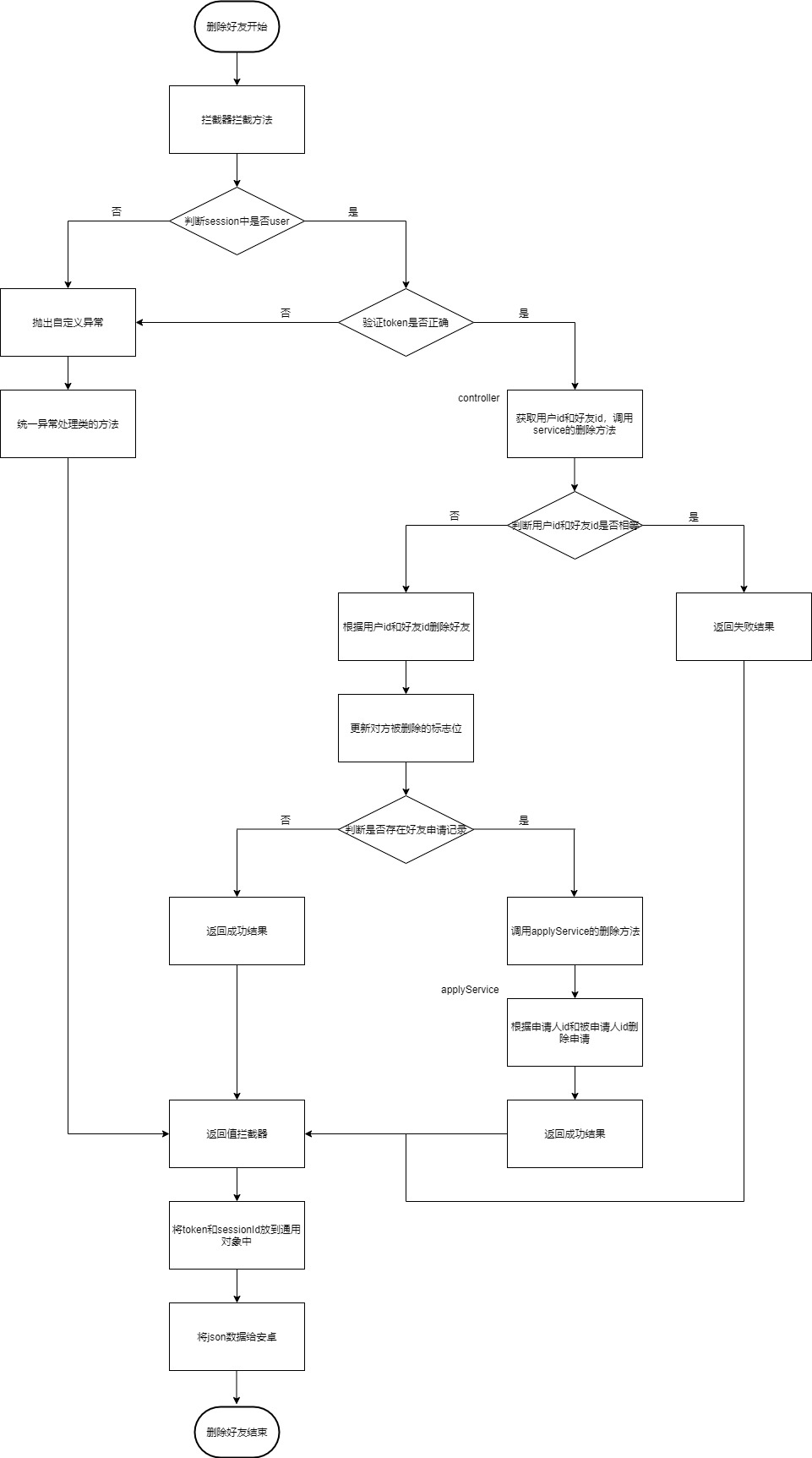
1 controller 获取用户id和好友id，调用service

2 service 判断用户id和好友id是否相同，是，则直接返回结果

3 删除好友，同时更新另一边好友被删除的标志位

4 判断好友申请是否是他提出的，是，删除好友申请

5 返回结果



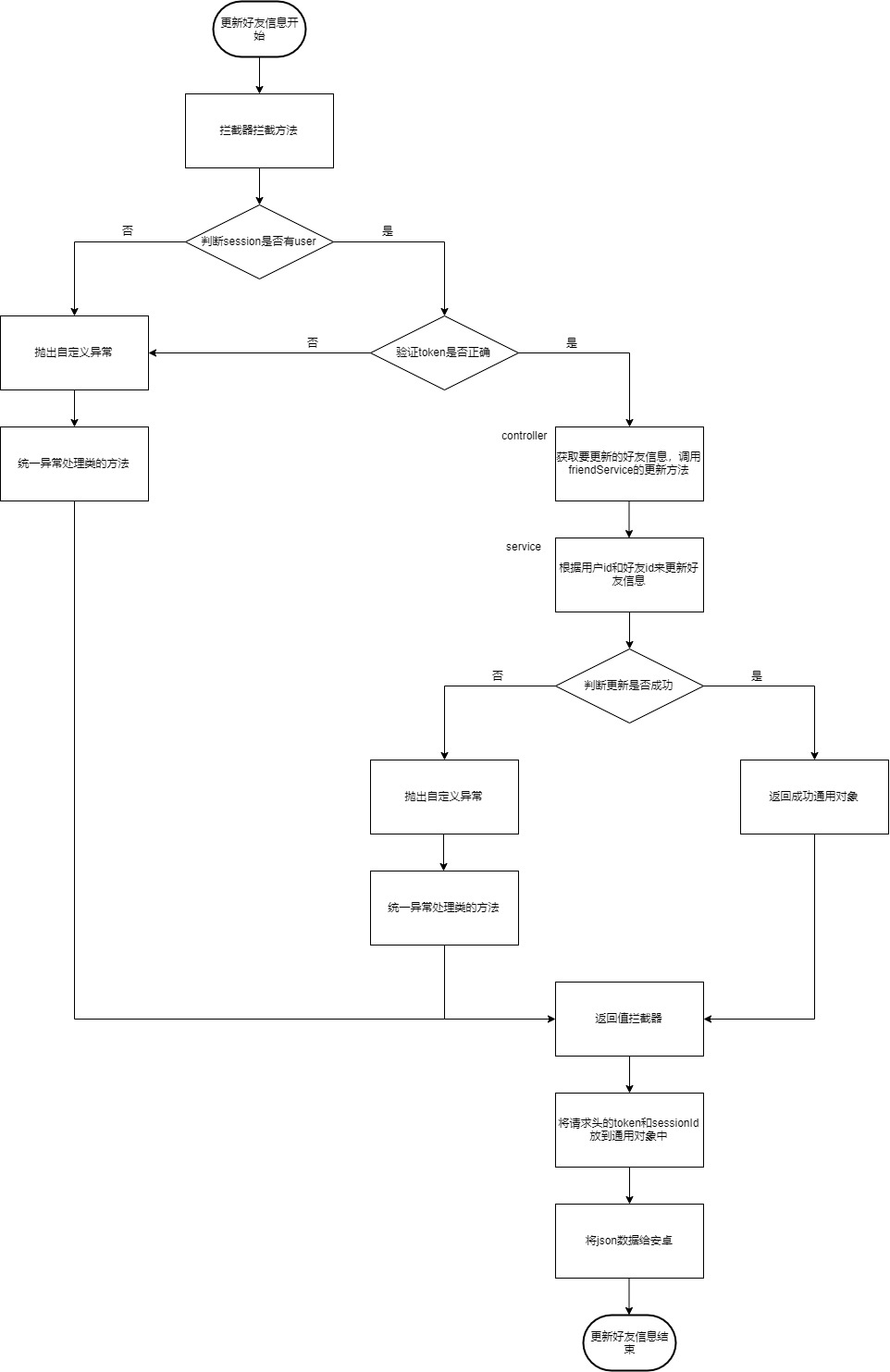
修改好友信息

修改好友信息功能

1 controller 获取要修改好友信息的对象，调用service

2 service 调用dao层更新好友信息。

3 返回结果



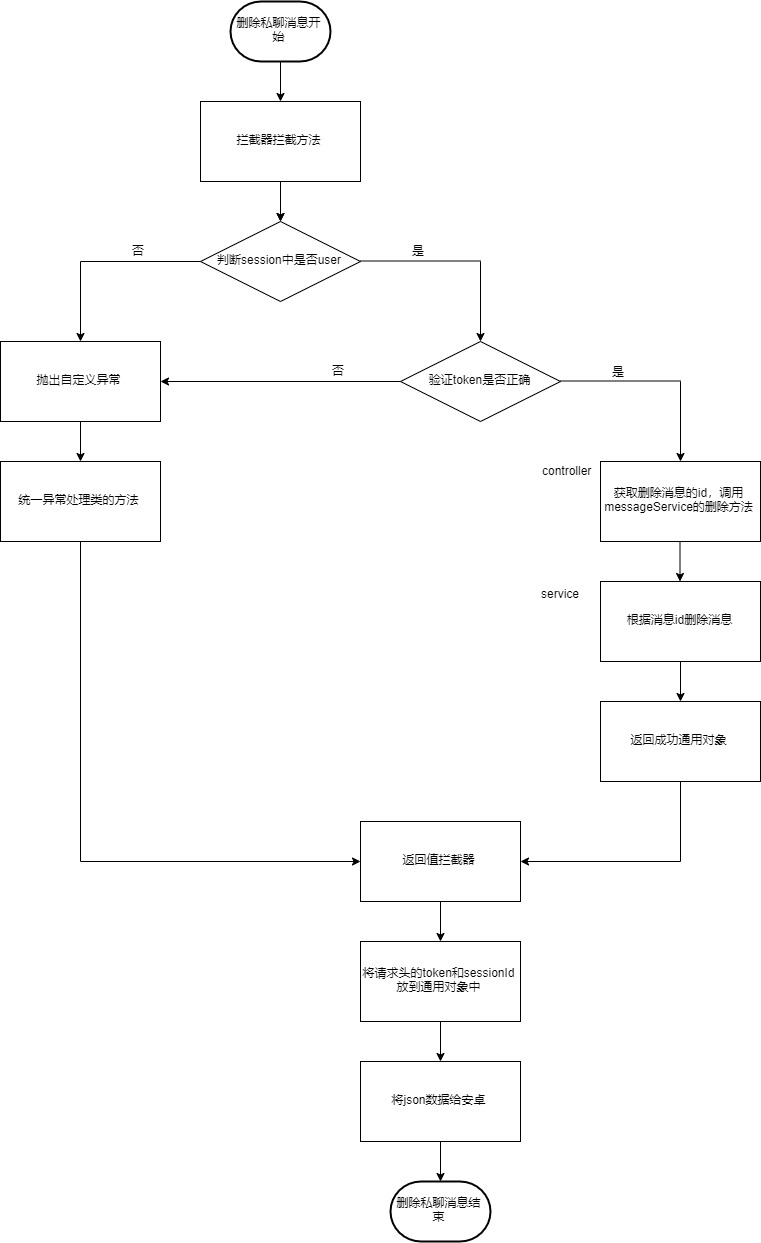
删除私聊消息

删除私聊消息功能

1 messageController 获取要删除消息的id，调用messageService的删除方法

2 service 根据id删除对应的消息

3 返回结果



查询用户和好友的聊天记录

查询用户和好友聊天记录的功能

1 groupToUsercontroller 获取用户id和好友id，调用groupToUserService的查询方法

2 groupToUserService通过多表连接，获取聊天记录和用户头像和昵称

3 返回结果

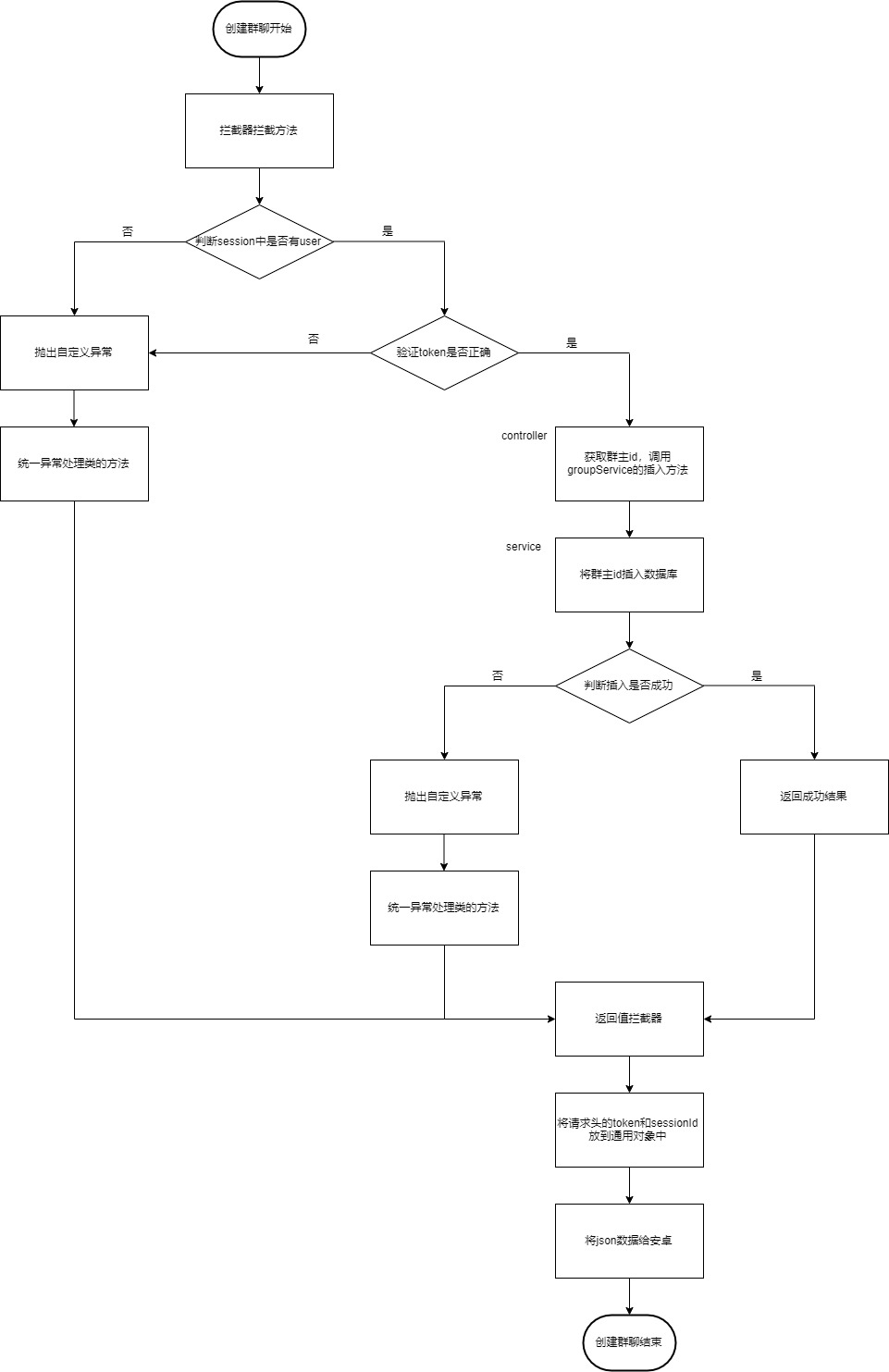


创建群聊

创建群聊功能

1 groupController 获取群主，调用groupService的insert

2 groupService根据群主id创建群聊，触发器自动把群主添加到群成员中

 3 返回结果

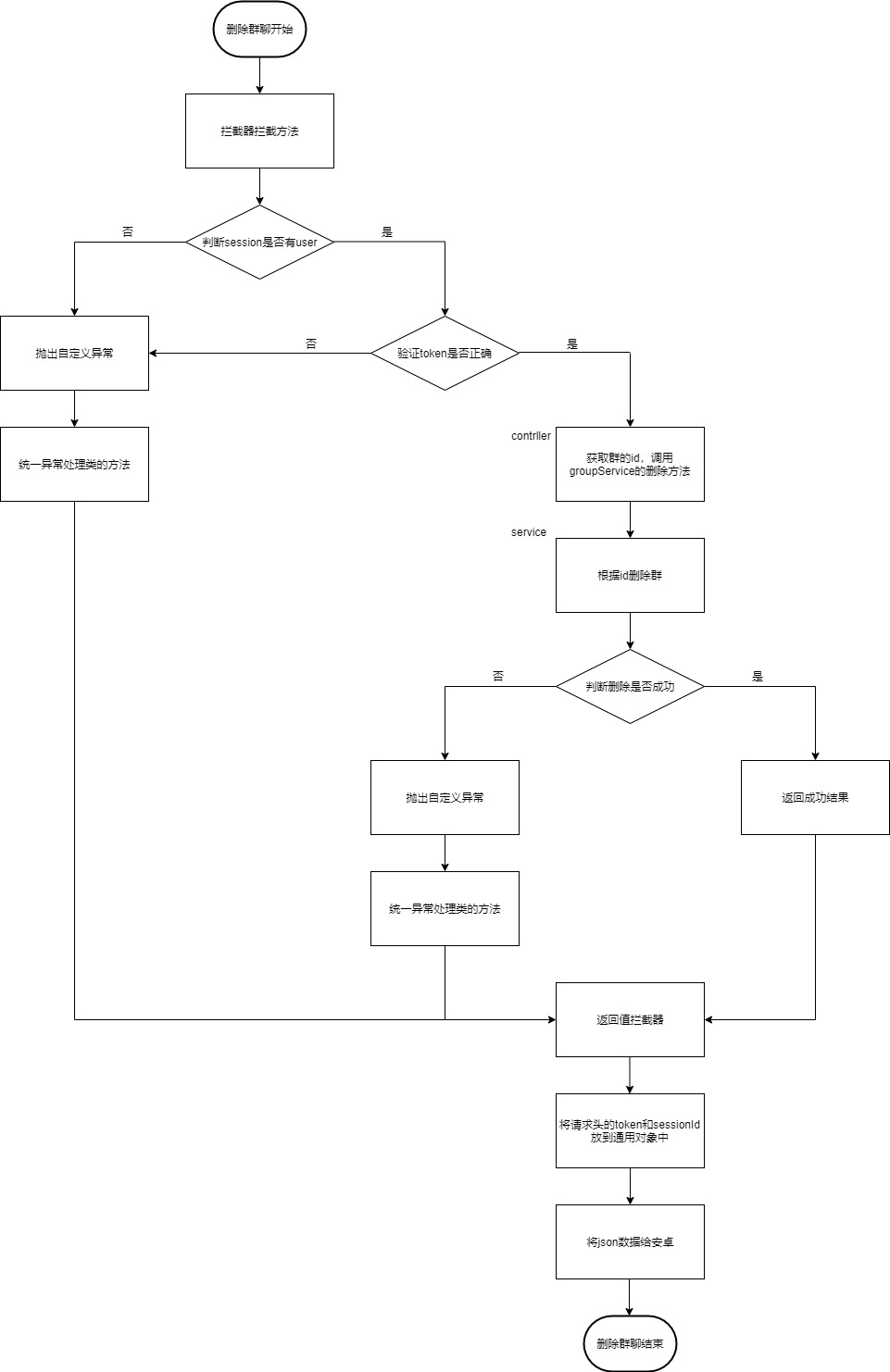
删除群聊

创建群聊功能

1 controller 获取群id，调用service

2 service 根据群id删除群聊

3 返回结果



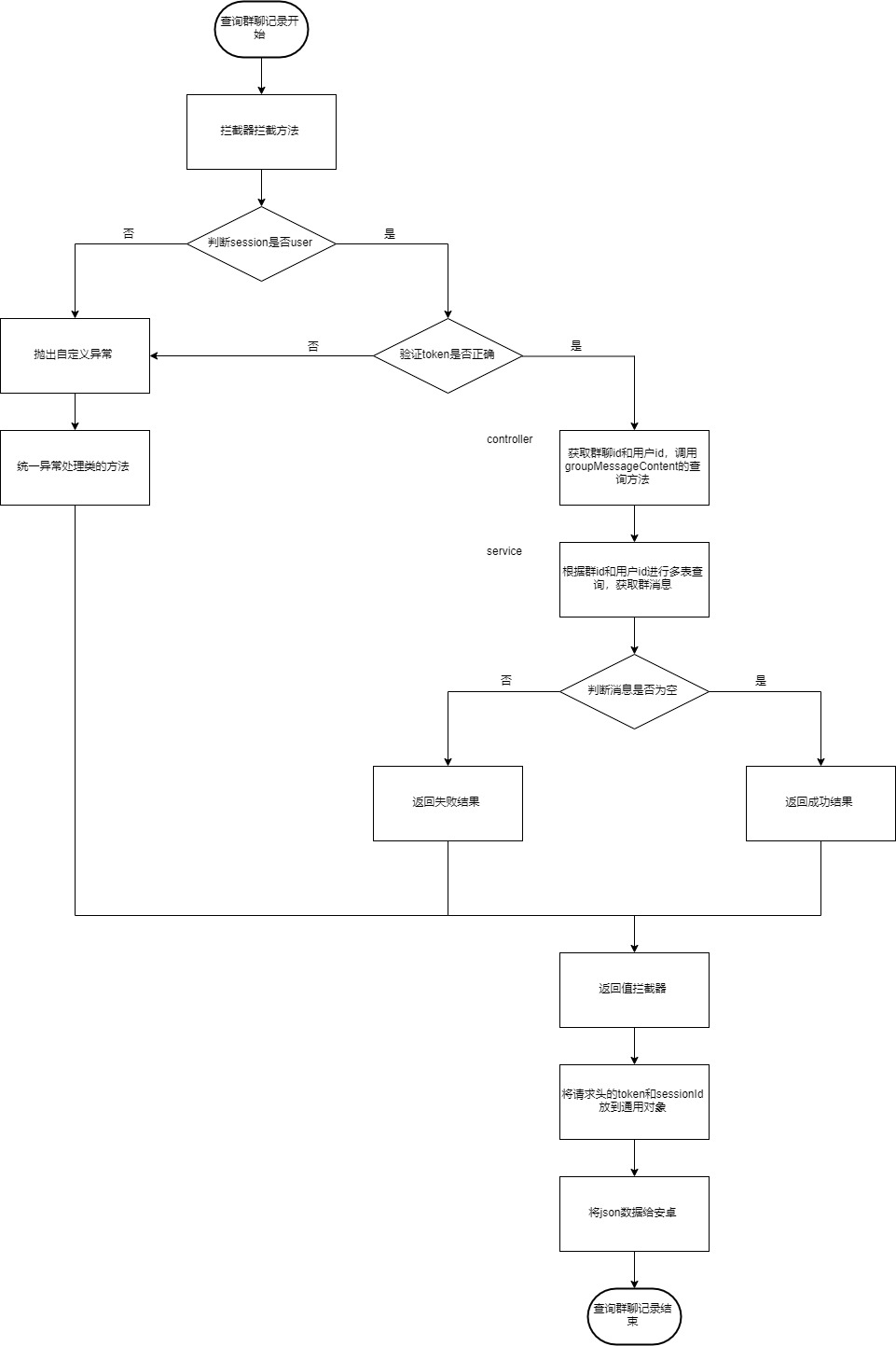
查询群聊消息

查询群聊消息功能

1 controller 获取群id和用户id，调用groupMessageContentService的查询方法

2 service 根据群id和用户id进行多表查询

3 返回结果



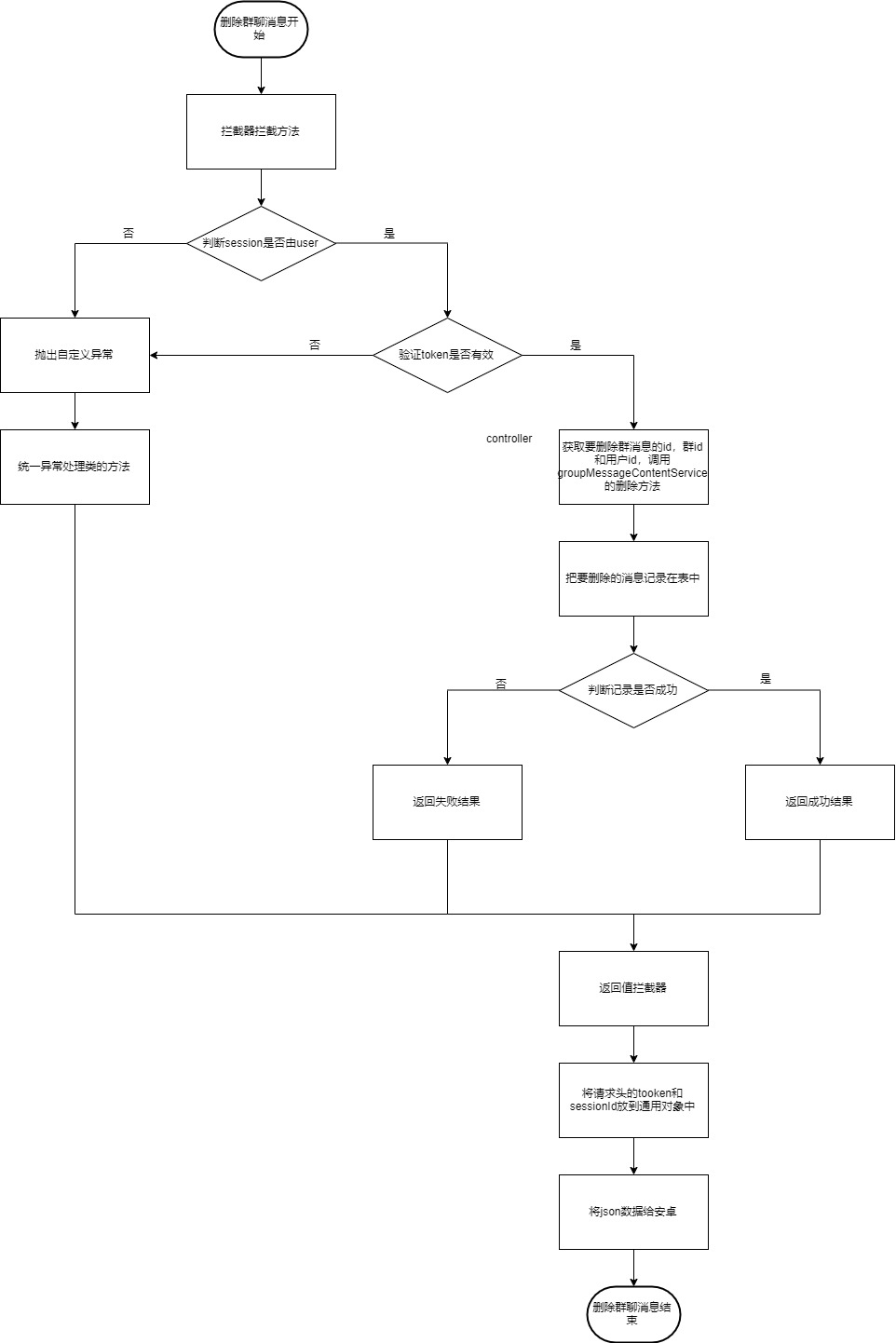
删除群聊消息

删除群聊消息功能

1 controller· 获取要删除群聊的消息id，群聊id，用户id，调用groupMessageContentService的删除方法

2 service 把消息id和群聊id，用户id记录在删除表

3 返回结果



查询用户在的所有群

查询用户在功能

1 groupController 获取用户id，调用groupService的查询方法

2 groupService 根据用户id查询所有群聊

3 返回结果



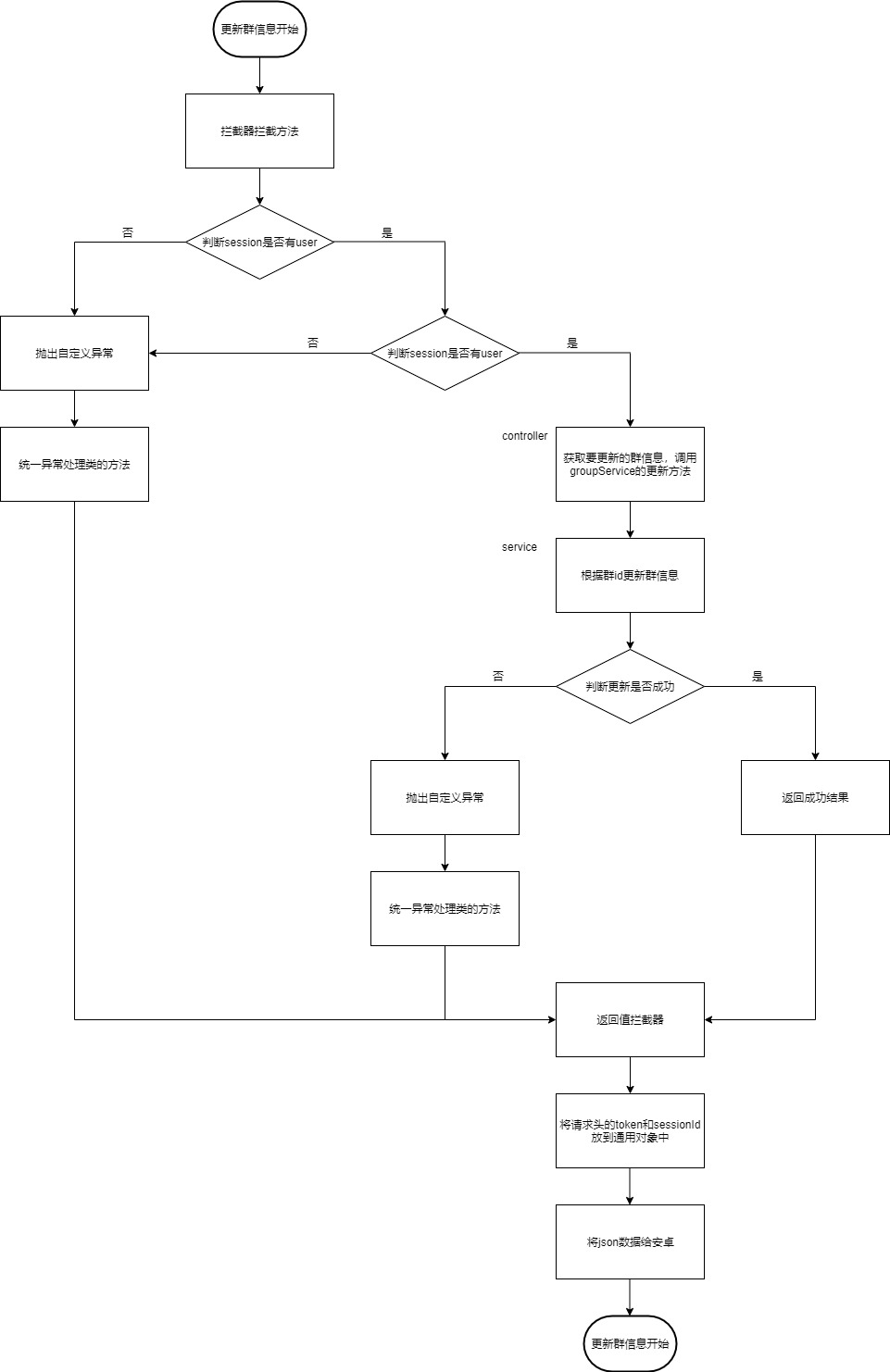
更新群信息

更新群聊功能

1 GroupControlle的update获取要更新的群信息，调用GroupService的update方法

2 GroupService根据群id删除群聊

3 返回结果



添加群用户

添加群用户功能

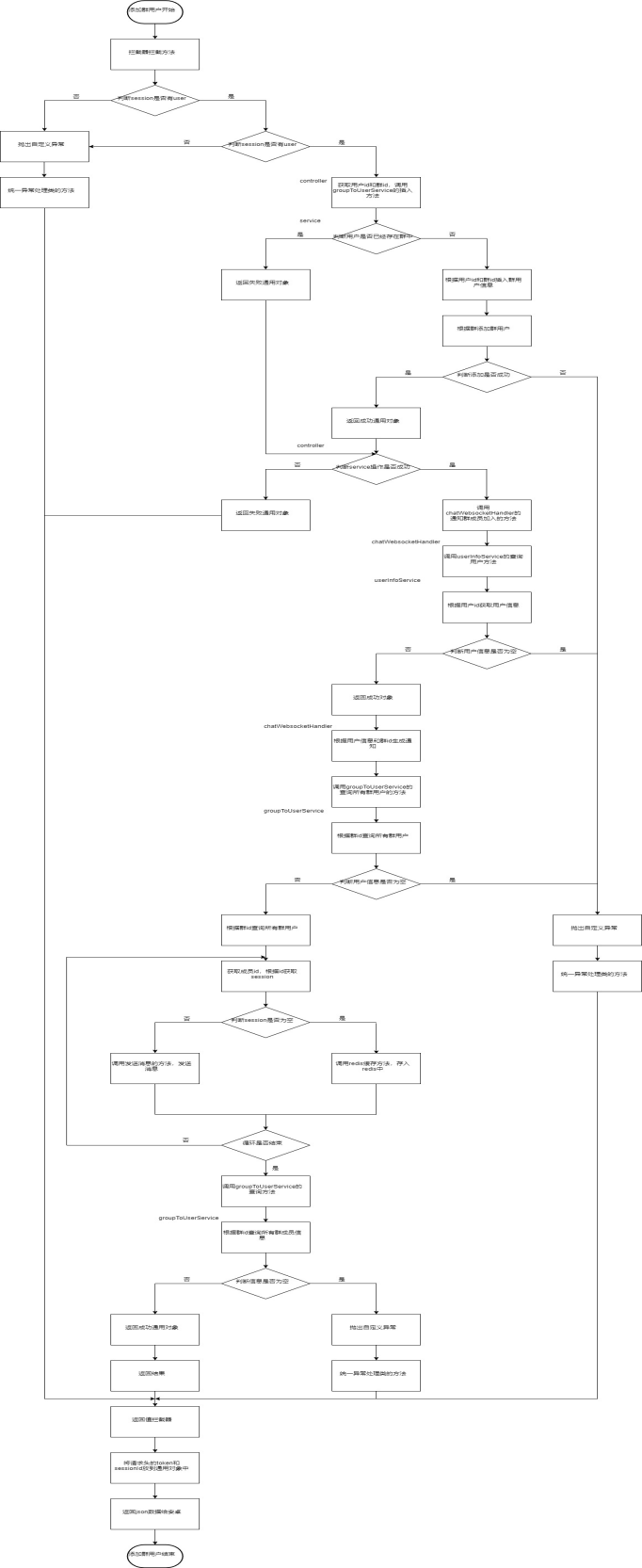
1 groupToUserController的insert来添加群用户，调用groupToUserService的insert方法

2 groupToUserService 判断群用户是否已经加入群聊，是，返回错误结果

3 添加群用户

4 调用ChatWebSocketHandler的systemAdviceGroupMember方法通知用户加入群聊

5 返回结果

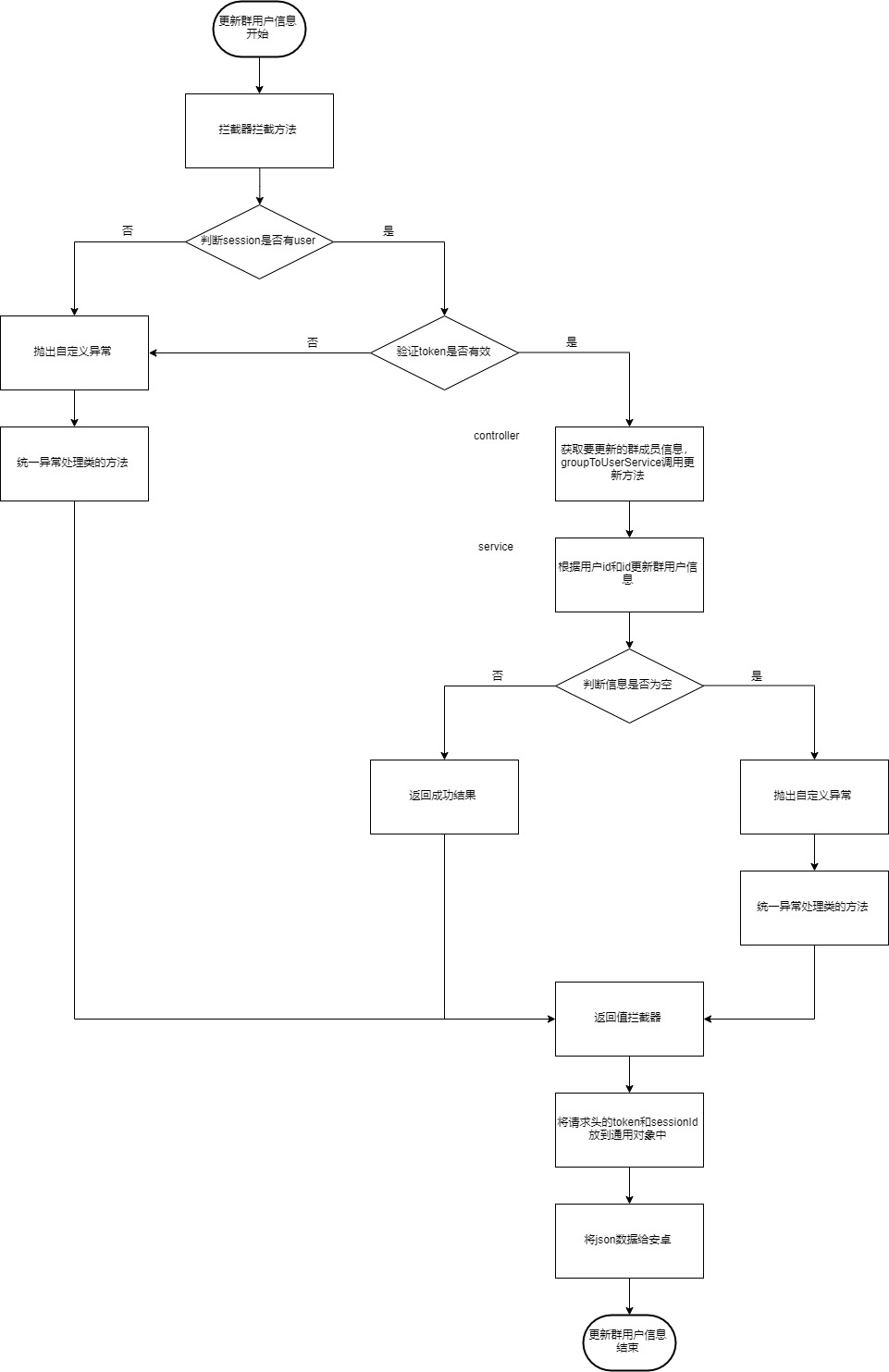


更新群用户信息

更新群用户功能

1 groupToUserController的update来更新群用户，调用groupToUserService的update方法

2 groupToUserService 更新群用户信息

 3 返回结果

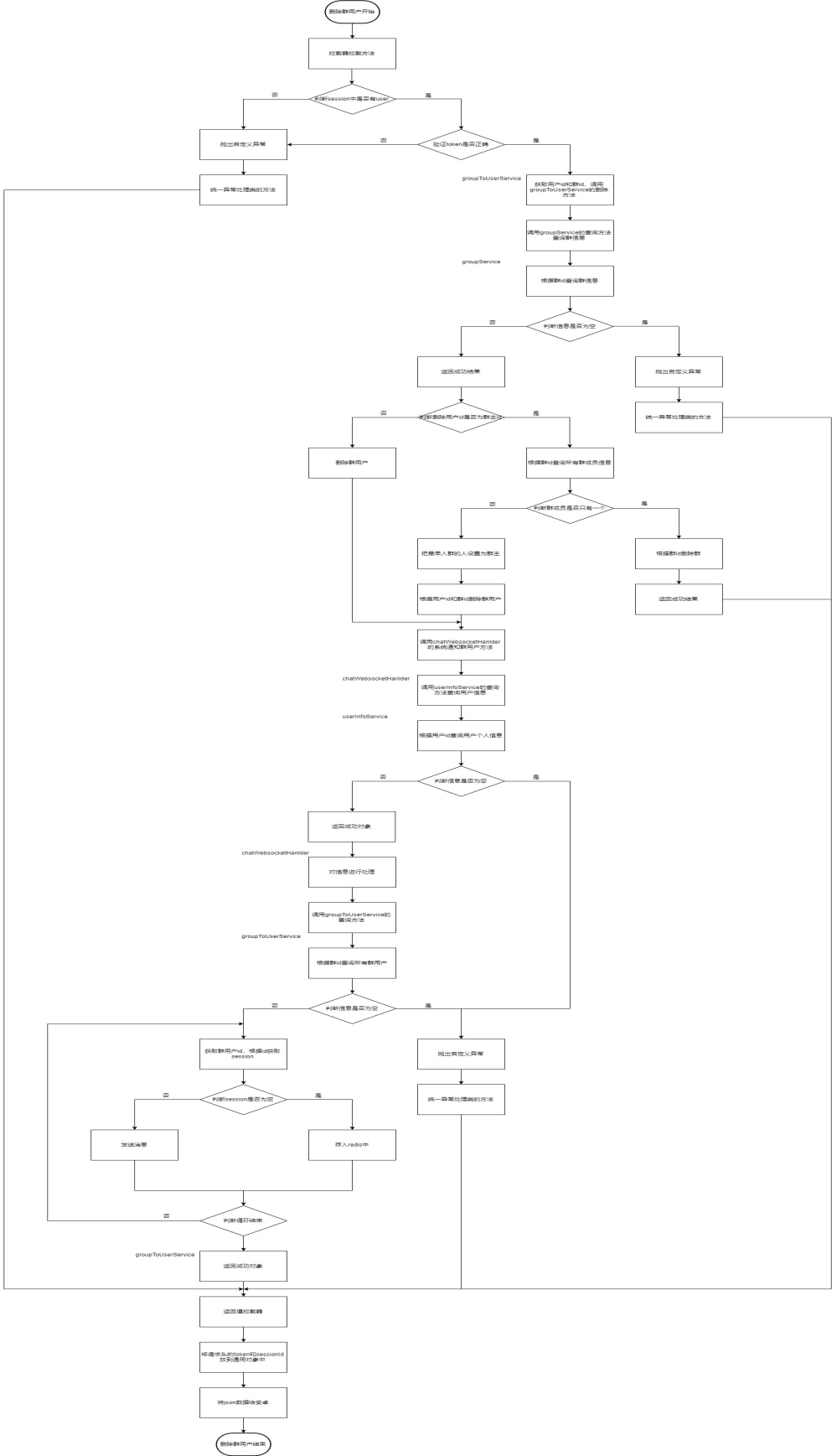
删除群用户

删除群用户功能

1 groupToUserController的delete来添加群用户，调用groupToUserService的delete方法

2 groupToUserService删除群用户

3 返回结果



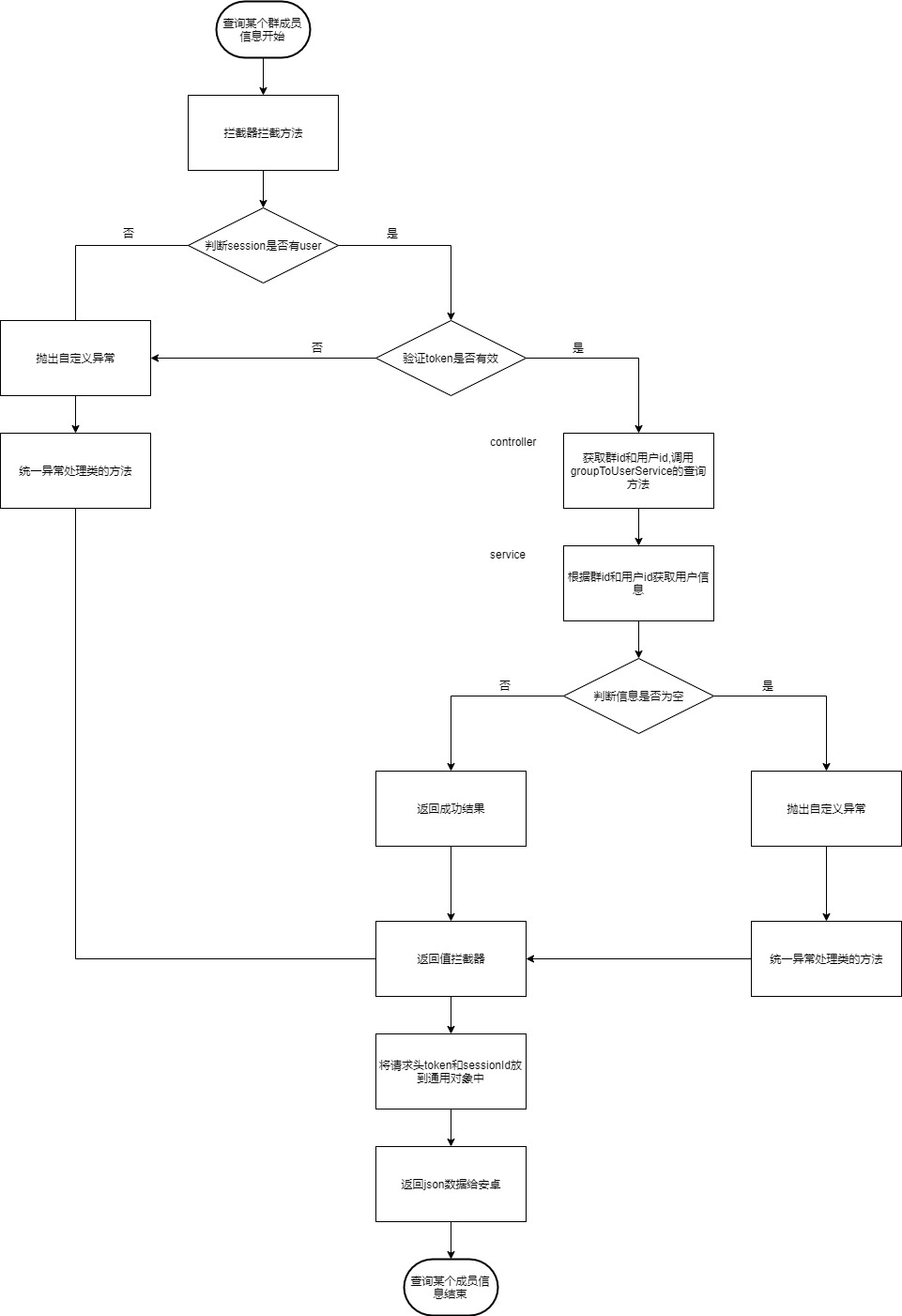
查询群内某个用户信息

查询群内某个群用户功能

1 groupToUserController的selectGroupToUser来查询群用户，调用groupToUserService的selectGroupToUser方法

2 groupToUserServiceg 根据用户id和群id查询群用户信息

3 返回结果



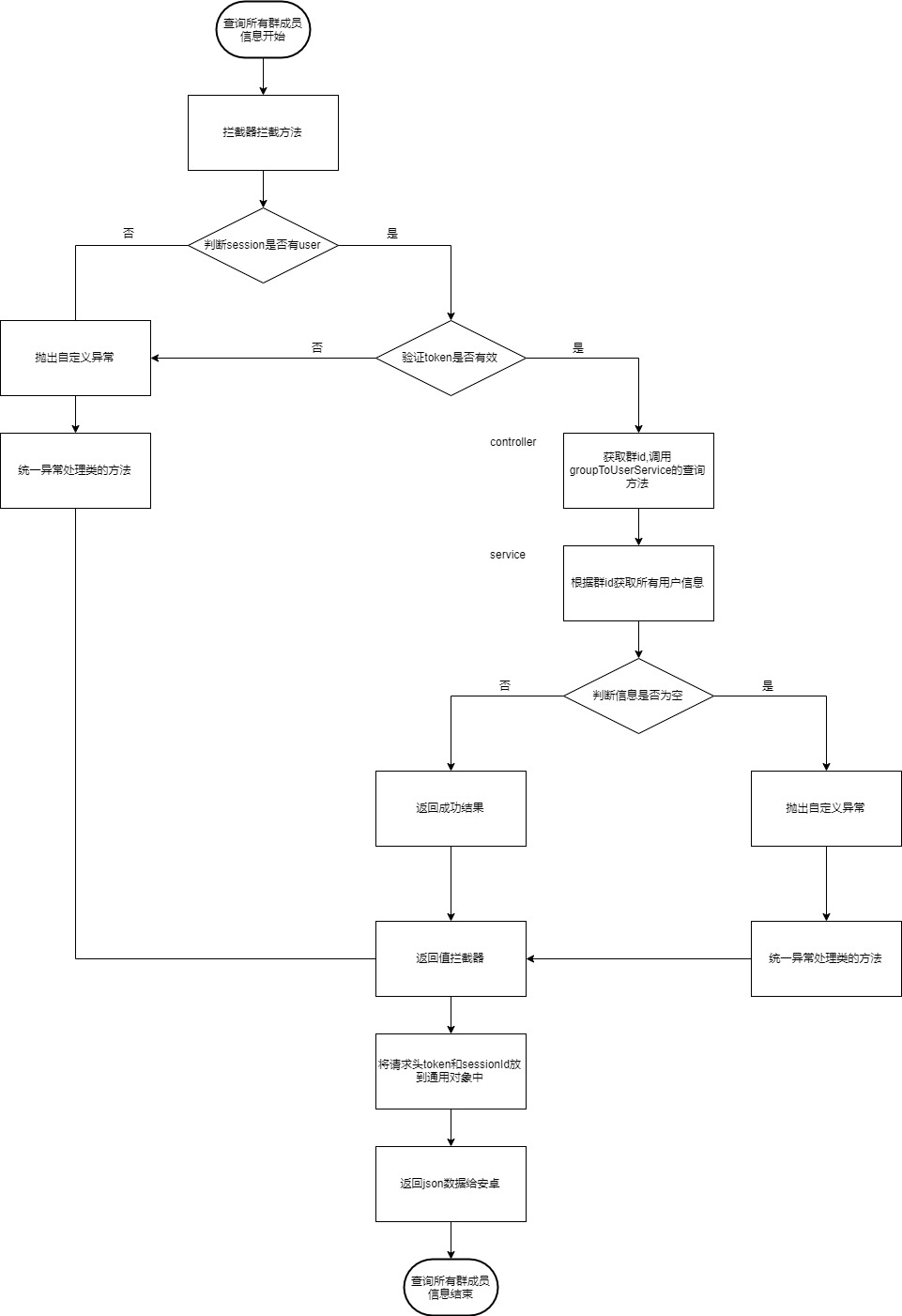
查询群内所有用户信息

查询群内所有群用户功能

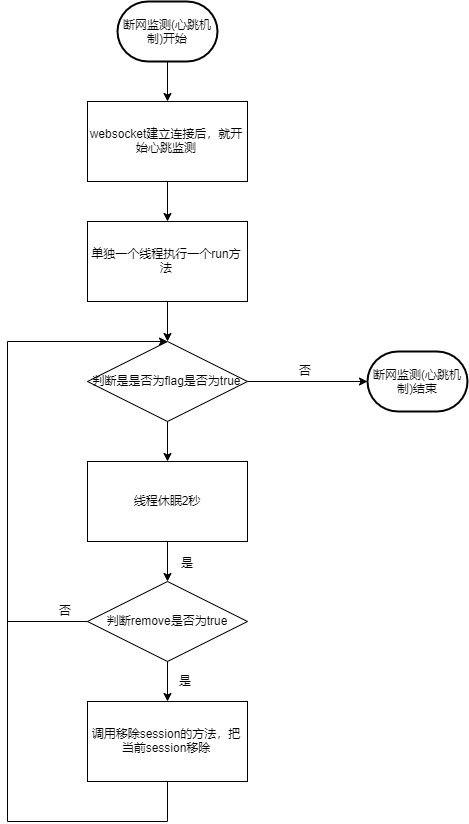
1 groupToUserController的select来查询群用户，调用groupToUserService的select方法

2 groupToUserServiceg 根据群id查询所欲群用户信息

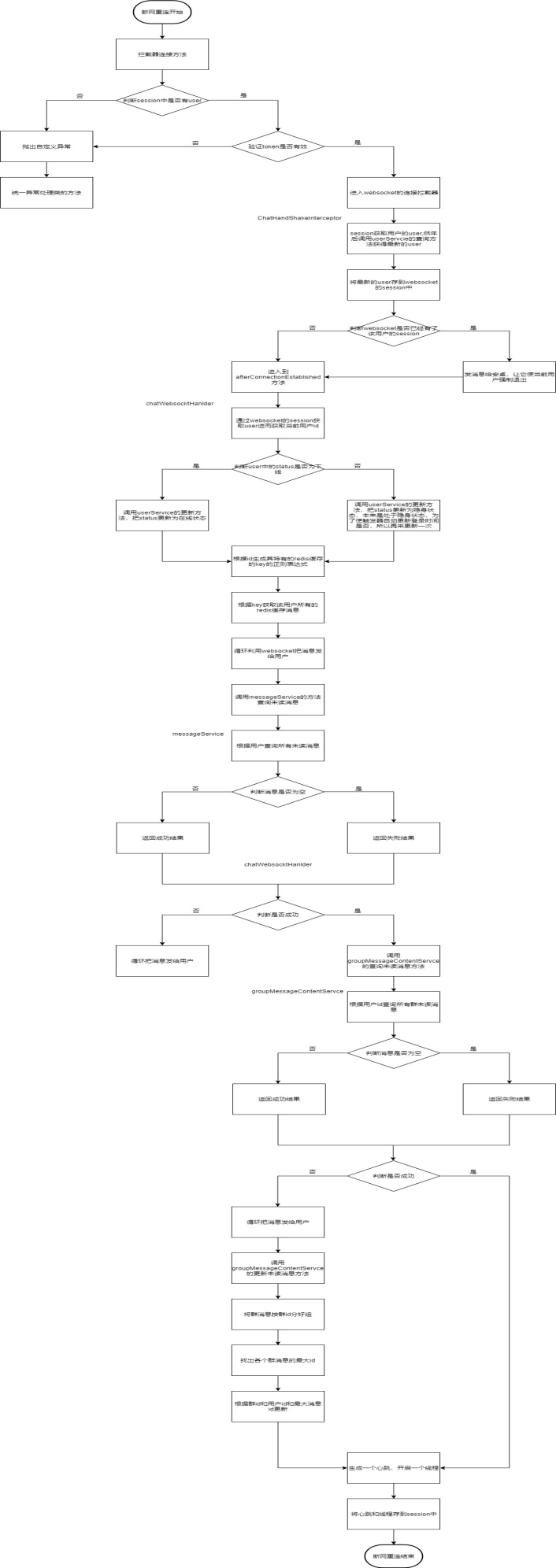
3 返回结果



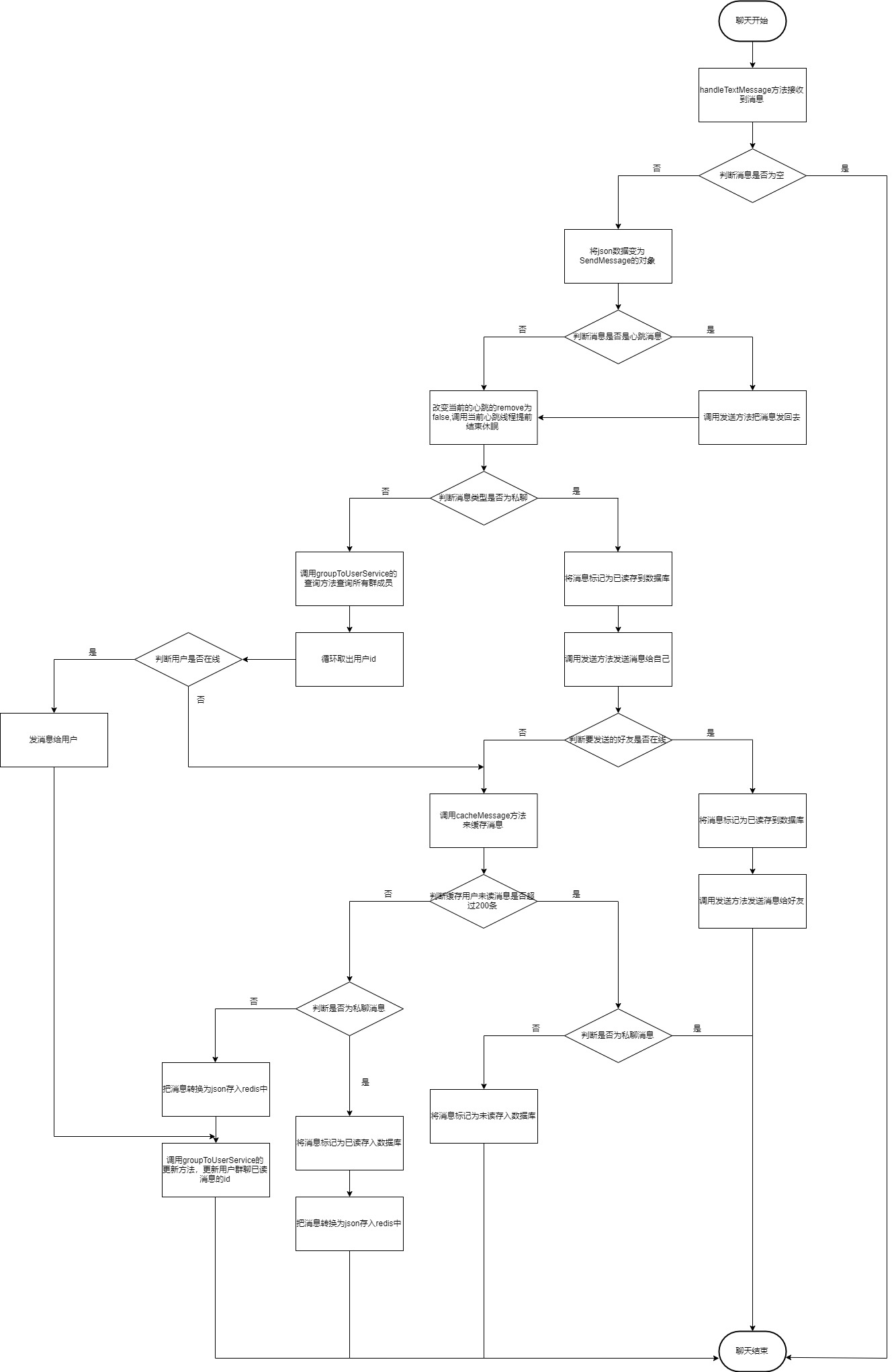
断网监测



断网重连



聊天过程



文件类型的聊天消息发送

其实是先由安卓调用上传文件的接口，把文件上传到后台服务器，我把对应的生成的虚拟访问路径给安卓，安卓直接把那个虚拟路径封装一下发送出去。

心得体会

第一次接触到安卓和后台对接工作，虽然开始都是懵懵懂懂，啥都不会，到慢慢的跌跌撞撞踩了许多坑，看了许多博客，做了许多事情，敲了许多代码之后也都慢慢懂得怎么配合，怎么合作，怎么沟通，发现问题就需要双方积极配合找出问题发生在哪里，单独一方找问题都是很困难的，而且找问题也需要双方耐心的等待，去反省过程，发现问题。

不知不觉已经过了差不多40天了，我们合作的成果也差不多形成了，虽然还有许多缺陷和小问题，但是还是做了出来，内心还是满满的成就感，不管结果如何，我们起码再这个过程中学到了许多东西，许多知识，还有考验我们具有坚持不懈的精神。