**有来商城youlai-mall**

**设计文档**

课程名：软件工程概论

一个组

一．引言

1.1编写目的

在编写本文档时，我们需要考虑开发人员的角度，希望能够为开发人员提供清晰的指导和依据，以便开发人员可以更好地理解整个项目的设计思路，规范开发流程，减少错误和漏洞，并且提高项目开发效率和质量。此外，本文档还面向项目经理和其他相关人员，以期帮助他们更好地了解项目的整体设计思路和实现过程，更好地管理项目进度和保障项目质量。本次youlai-mall后端重构项目的编写目的是为了提高项目的整体质量和商业价值，同时提供清晰的指导和依据，以实现项目开发的高效、规范和高质量。  
  
1.2读者对象

本文档主要面向开发人员、测试人员、项目经理和其他相关人员，在项目开发过程中，需要参与到youlai-mall后端重构项目中的各个环节，帮助推动项目的顺利进行。对于开发人员而言，本文档将提供详细的设计方案、技术选型、代码实现等内容，以便开发人员能够更好地理解项目的整体架构设计，并且在开发过程中遵循统一的规范进行代码编写，保证代码质量和可维护性。对于项目经理而言，本文档将提供项目的目标和要求、进度计划、质量分析工具等信息，以便项目经理能够更好地管理项目进度、掌握项目质量，优化项目开发效率和资源利用率。对于其他相关人员而言，本文档将提供项目的概述、功能介绍、技术选型等信息，以便其他相关人员能够了解项目的整体架构和实现过程，从而有效地协同工作，实现项目的共同目标。

1.3软件系统概述

youlai-mall是一个基于Spring Boot、Spring Cloud Alibaba、Vue、Element-UI、uni-app等主流技术栈构建的全栈电商平台。本次youlai-mall后端重构项目旨在优化平台后端部分，提高其性能和安全性，同时新增部分功能。

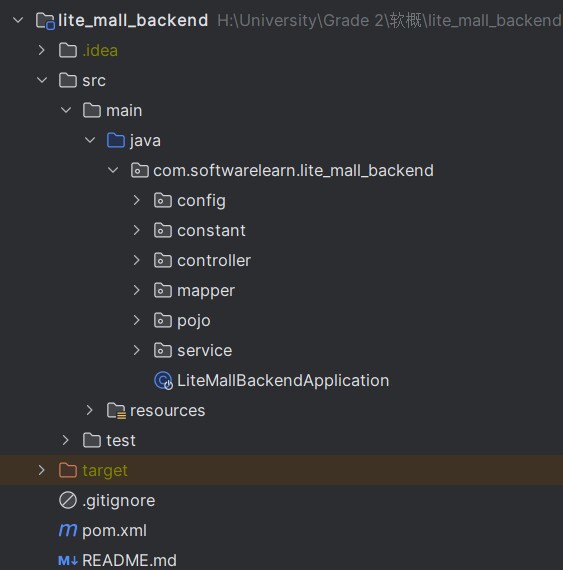
1.4参考资料

本文档参考了youlai-mall项目的相关文档、技术论坛和其他相关资料，其引用内容已在文中注明来源。

二．软件设计方案

2.1体系结构设计

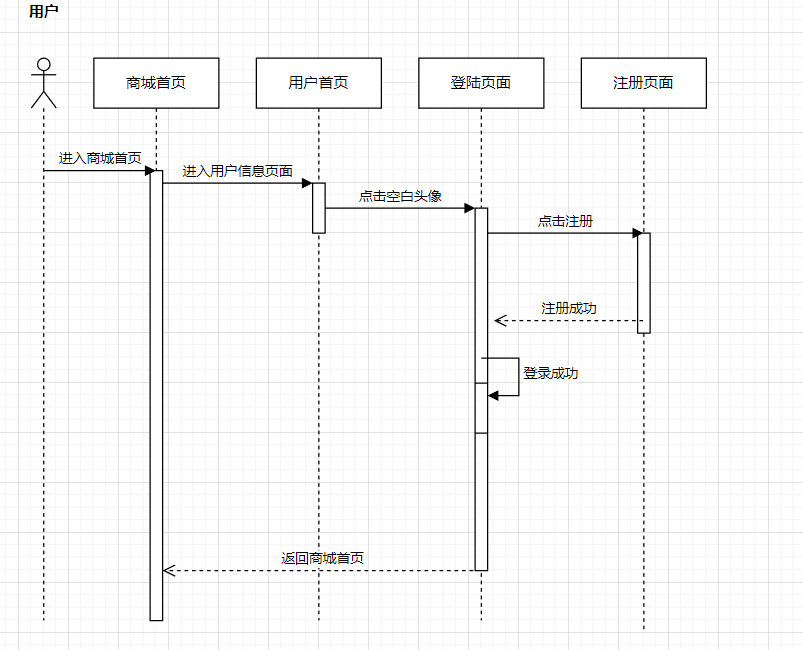
为了实现登录，查看数据等功能的重构，我们创建了新的spring boot后端项目，以下为项目目录结构：



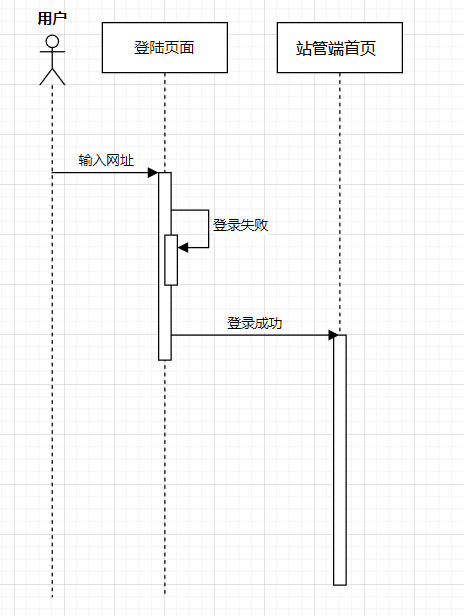
2.2用户界面设计

针对我们需要重构的功能，我们设计了APP端和站管端的用户界面交互图来描述用户与系统界面之间交互行为和流程的图形化表示，主要体现界面上的各种操作和响应。还设计了界面类图用于表示系统的整体架构和组件结构，并描述了用户界面与其他系统组件之间的关系和依赖，以及类与类之间的关联和交互。

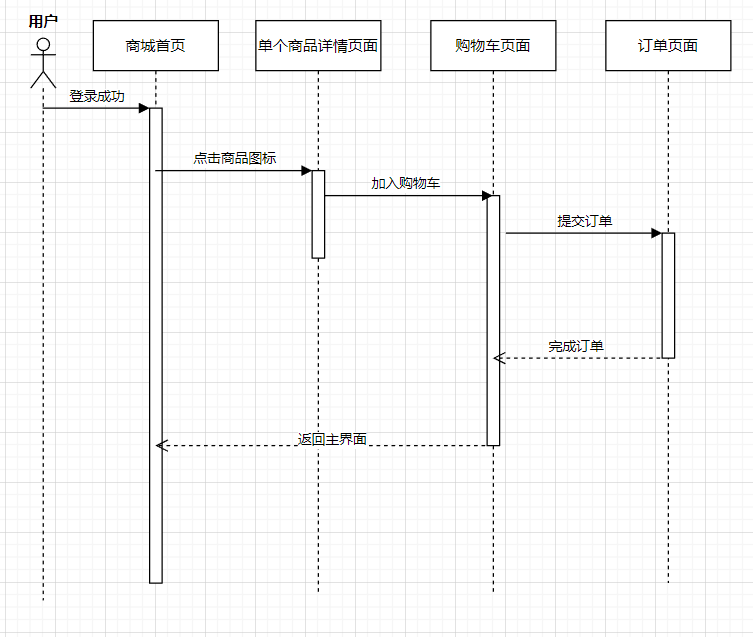
2.2.1 App端用户登录注册顺序图：



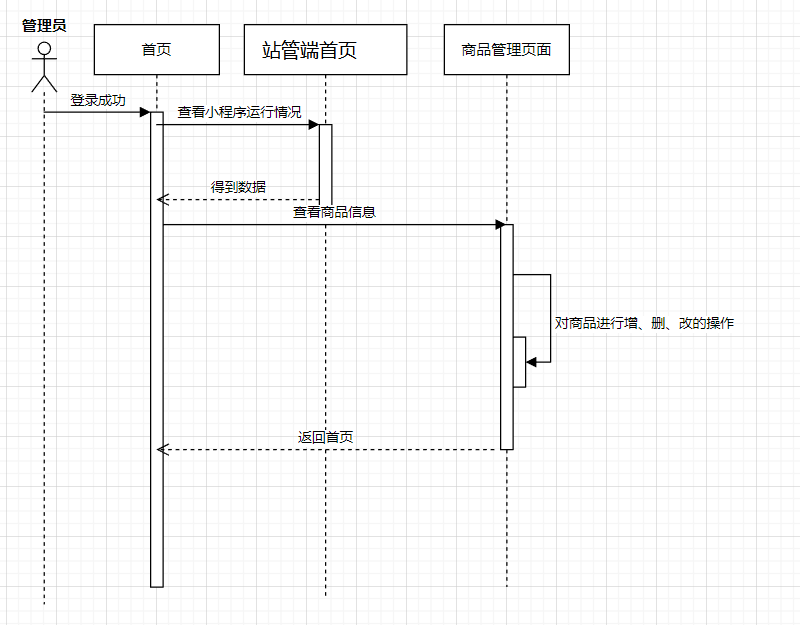
2.2.2 Web端管理员登录顺序图：



2.2.3 App端用户查看、购买商品顺序图：



2.2.4 Web端管理员查看、管理商品顺序图:



2.2.5 界面类图:

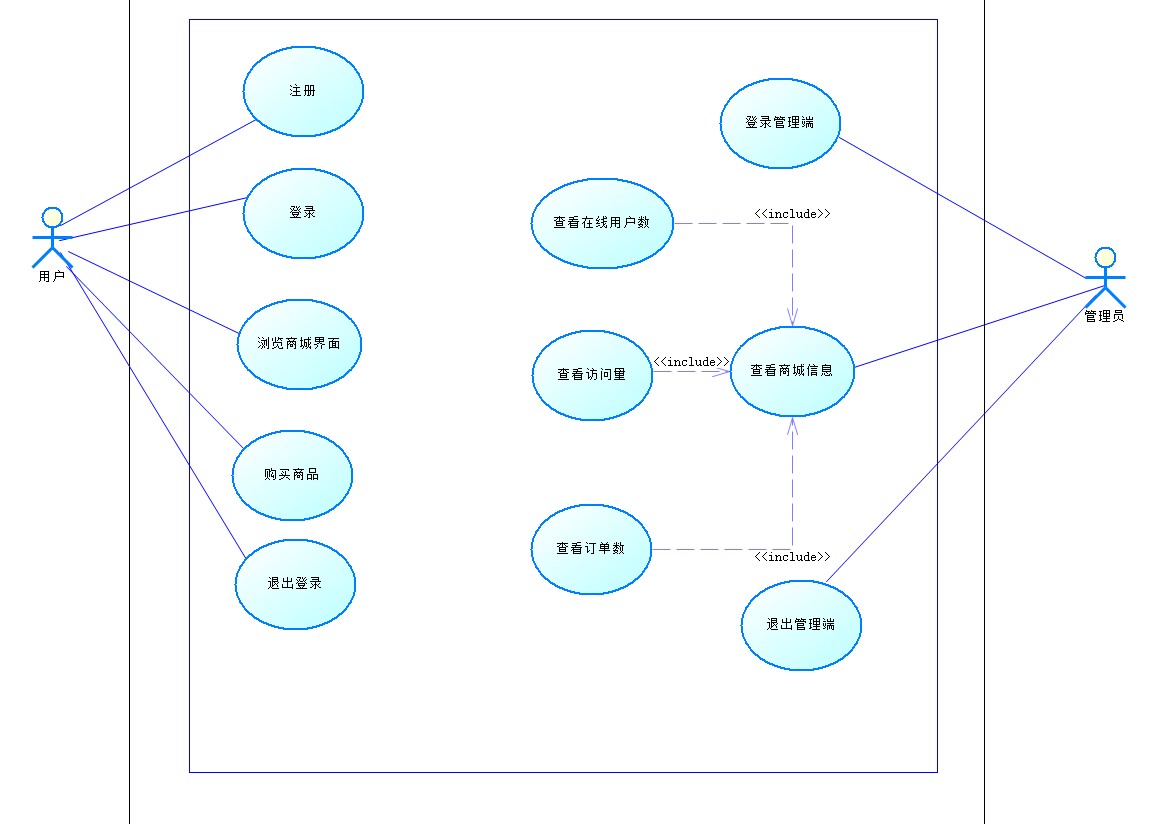
2.3用例设计

用例模型包括：用户注册并登录一个新的账号，随后进入商城界面，用户可以购买已上架的商品，管理员通过web站管端对查看APP端当前在线的用户数量，网站的总访客量以及总下单量。

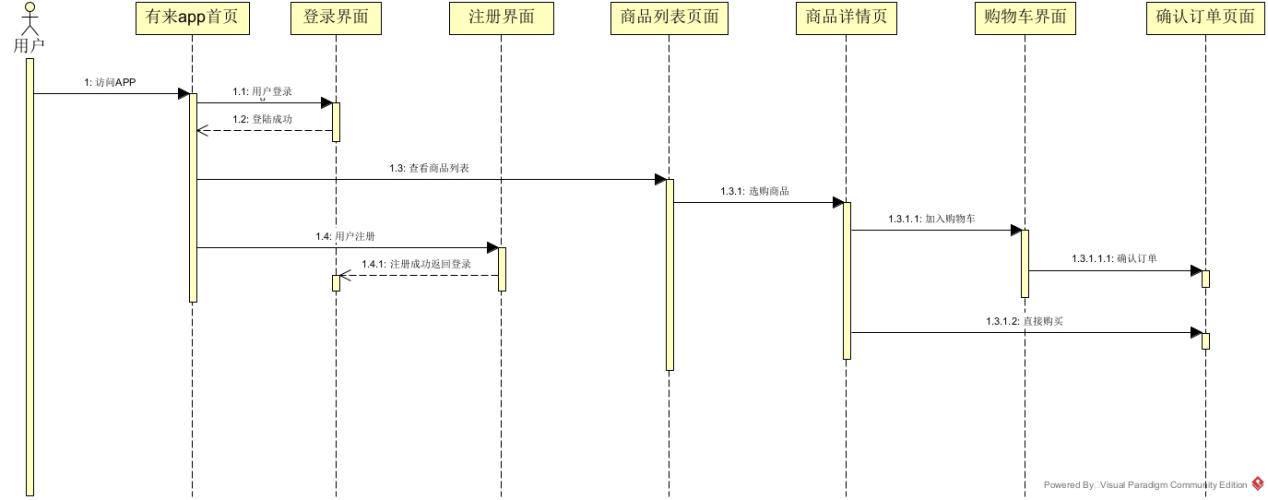
用户通过手机APP应用访问有来商城主页，通过搜索商品或是看主页的商品推荐来浏览自己所需要的商品，当用户在浏览主页时也可以看到主页各式各样的广告推荐以及各种优惠界面。当用户想要购买商品的时候可以下订单，通过填写收货地址等必要信息补充好订单后购买商品。

管理端可以查看当前APP端的用户在线总人数、APP的访客总人数、APP端的商品下单总量。

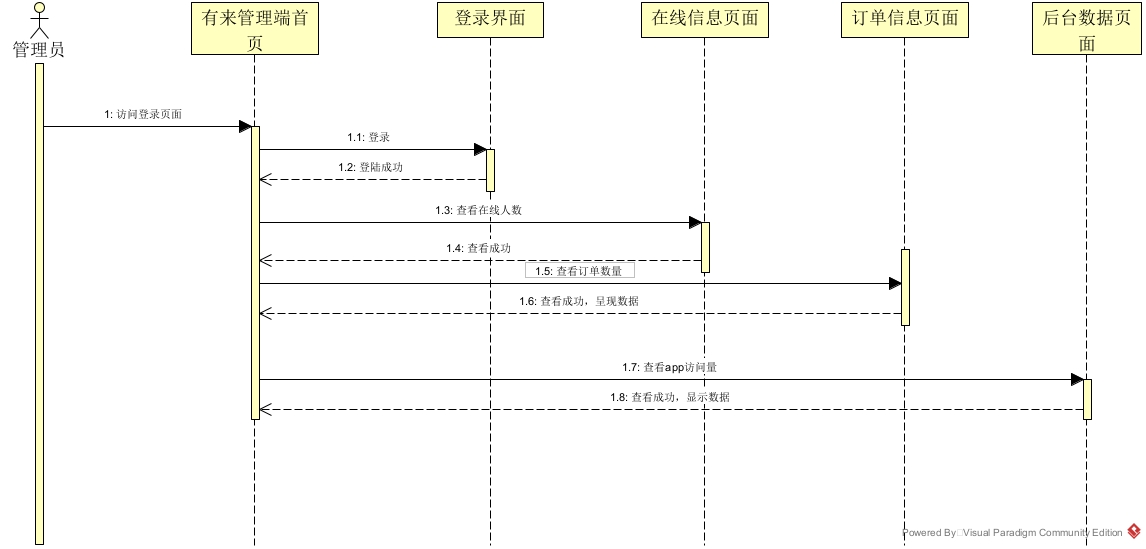
2.3.1 功能用例图：



2.3.2 APP端重构功能（登陆注册+购物）功能交互图：



2.3.3 Web站管端新增功能（查看数据）功能交互图：



2.4类设计

以下是后端部分一些核心的服务实现类每一个服务实现类都继承了ServiceImpl类，实现了对应的服务接口，并提供了一些操作。

GoodsServiceImpl类提供获取全部商品和获取指定商品的方法：

Public class GoodsServiceImpl extends ServiceImpl<GoodsInfoMapper , GoodsInfo> implements GoodService{

Private GoodsInfoMapper;

Public R getAllGoods();

Public R getGoods(GoodsInfo goodInfo);

}



OnlineUserNumServiceImpl类提供更新在线用户数、获取在线用户数、减少在线用户数和重置在线用户数等方法：

Public class OnlineUserNumServiceImpl extends ServiceImpl<OnlineUserNumMapper, OnlineUserNumInfo> implements OnlineUserNumService {

private OnlineUserNumMapper;

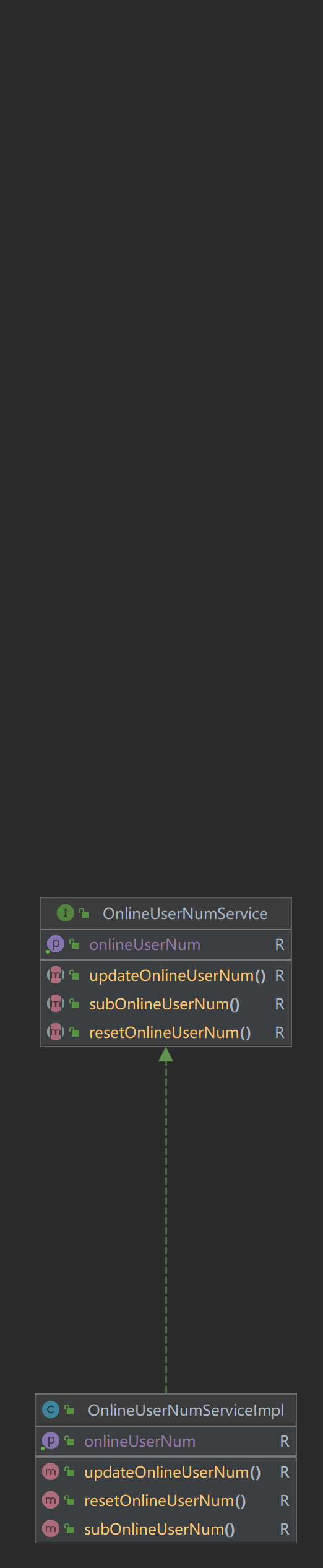
public R updateOnlineUserNum();

public R getOnlineUserNum();

public R subOnlineUserNum();

public R resetOnlineUserNum();

}



OrderNumServiceImpl类提供更新订单数、获取订单数、减少订单数和重置订单数等方法：

Public class OrderNumServiceImpl extends ServiceImpl<OrderNumInfoMapper, OrderNumInfo> implements OrderNumService {

private OrderNumInfoMapper;

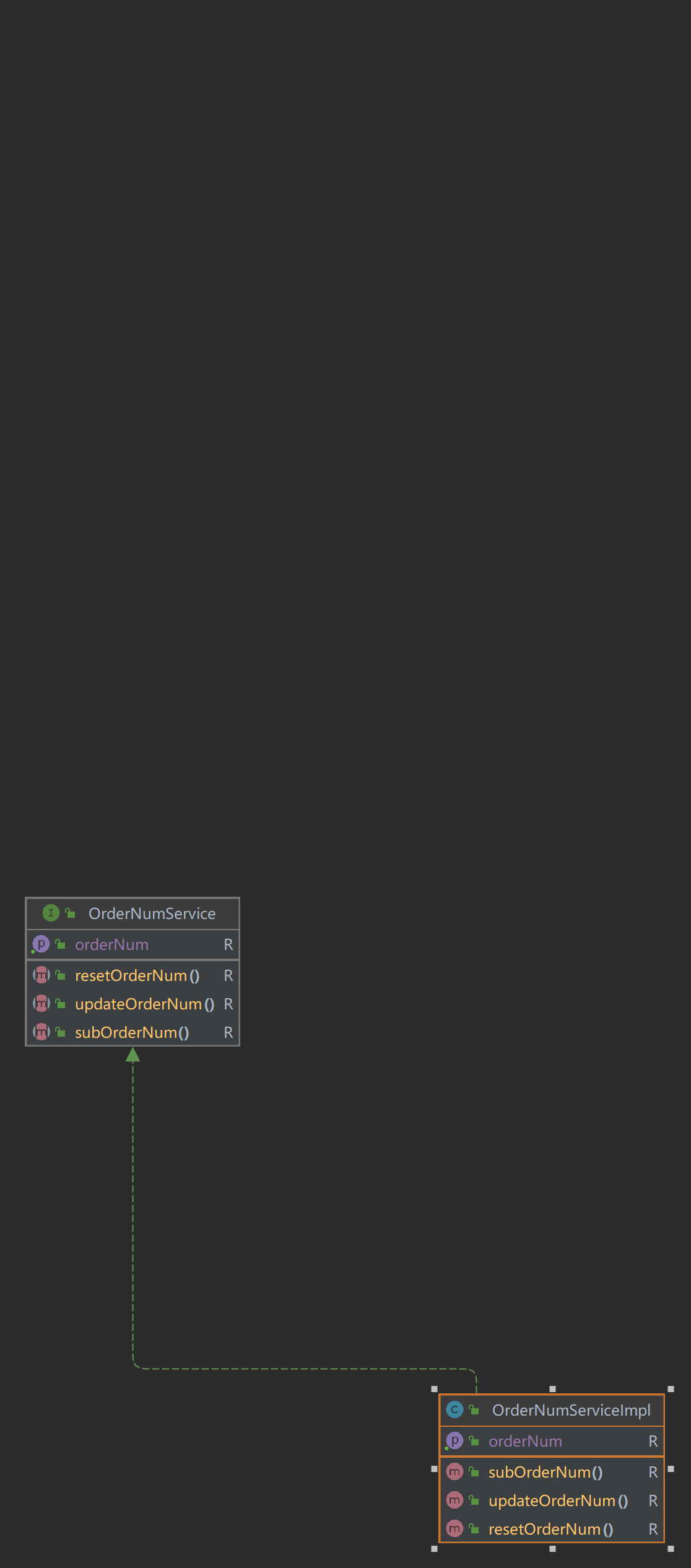
public R updateOrderNum();

public R getOrderNum();

public R resetOrderNum();

public R subOrderNum();

}



UserServiceImpl类提供注册用户、登录、获取用户信息、注销用户和修改用户信息等方法：

public class UserServiceImpl extends ServiceImpl<UserInfoMapper, UserInfo> implements UserService {

private UserInfoMapper;

public R register(UserInfo userInfo);

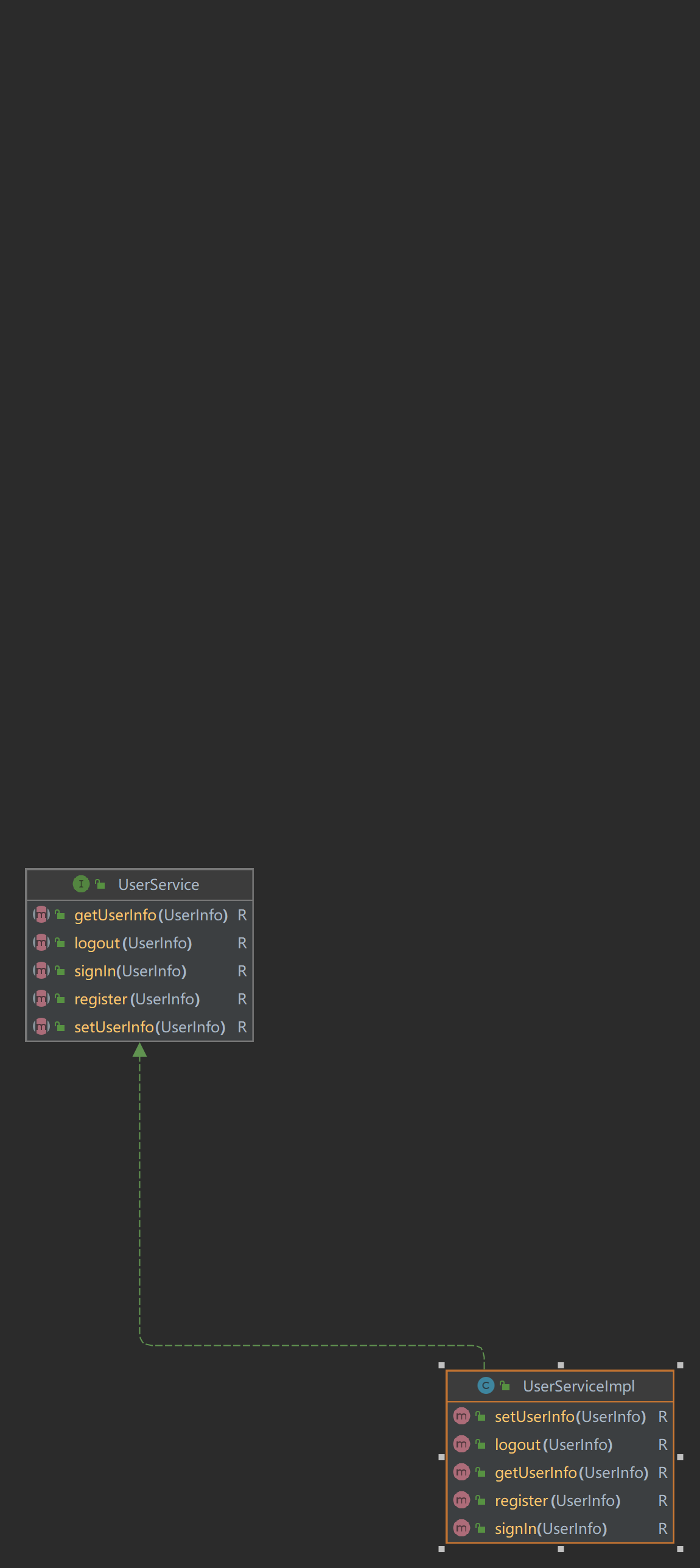
Public R signIn(UserInfo userInfo);

Public R getUserInfo(UserInfo userInfo);

Public R logout(UserInfo userInfo);

Public R setUserInfo(UserInfo userInfo);

}



VisitNumServiceImpl类提供更新访问数、获取访问数、减少访问数和重置访问数等方法：

public class VisitNumServiceImpl extends ServiceImpl<VisitNumInfoMapper, VisitNumInfo> implements VisitNumService {

private VisitNumInfoMapper;

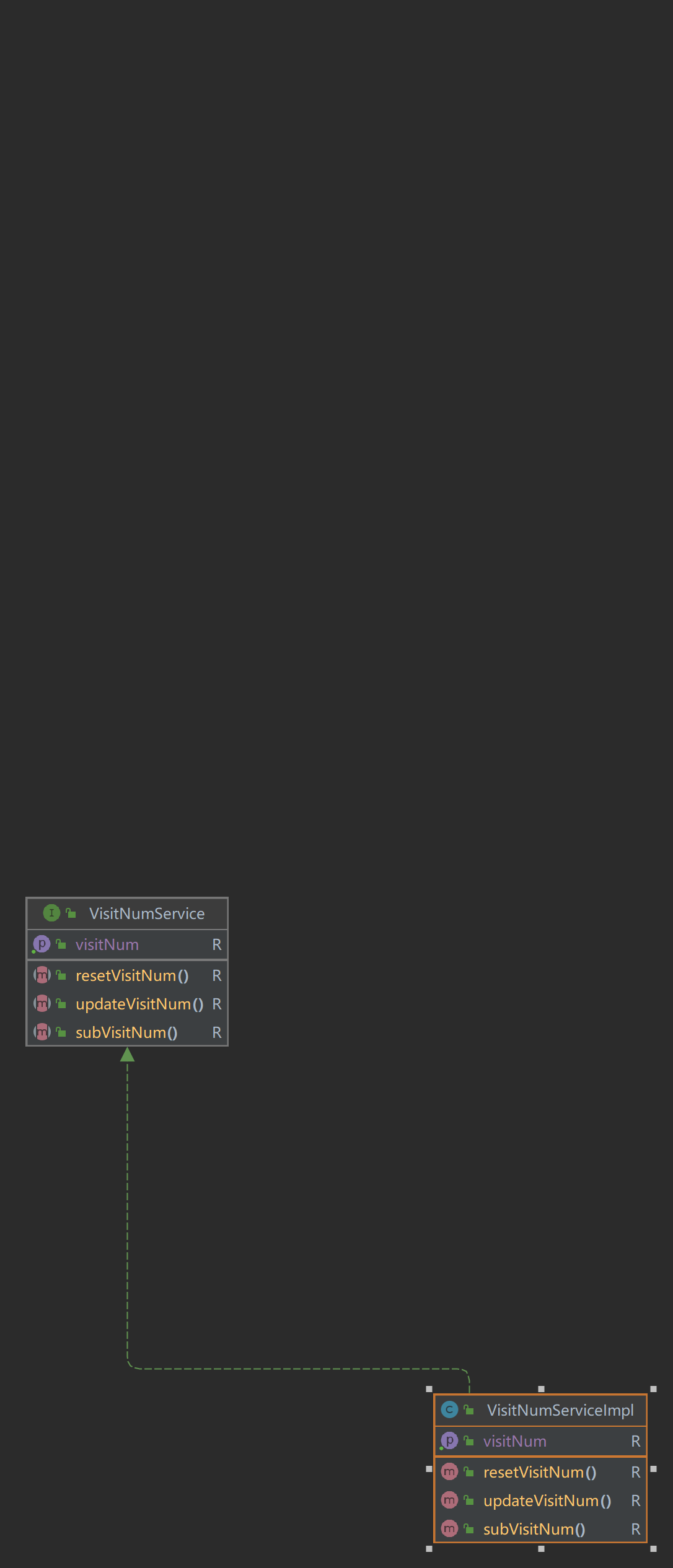
public R updateVisitNum();

public R getVisitNum();

public R subVisitNum();

public R resetVisitNum();

}



还定义了一个自定义的R类，其中包含成功、错误的标志，状态码、消息、数据以及一个用于装载任意键值对的HashMap。该类用于在服务层返回统一格式的结果对象，方便前端进行处理和解析：

Public class R{

Private int success;

Private int code;

Private String message;

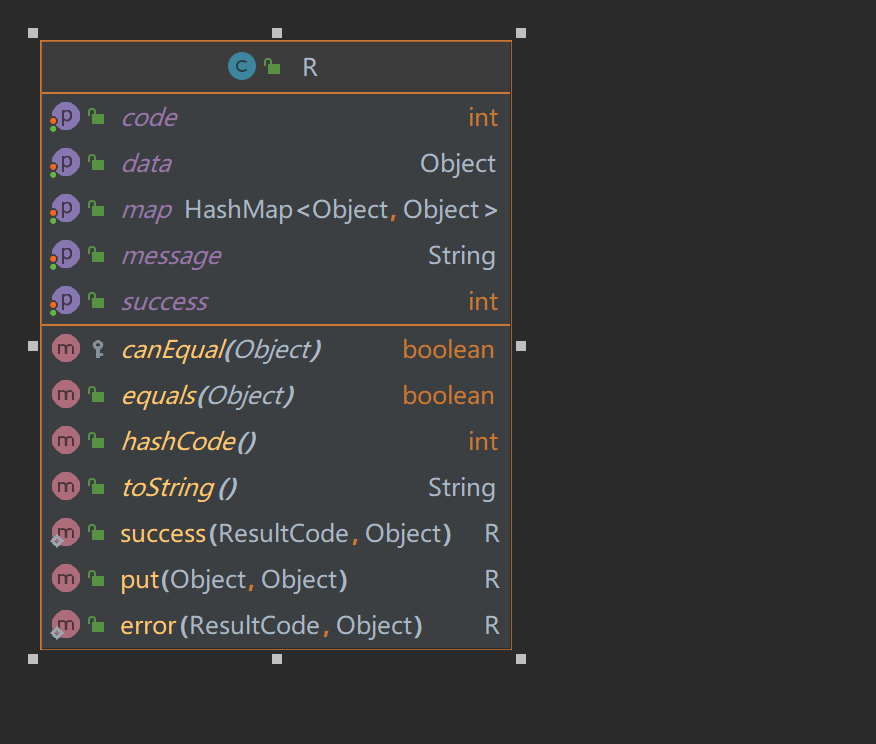
Private Object data;

Private HashMap<Object , Object> map;

Public static R success(ResultCode, Object data)

Public static R error(ResultCode, Object data)

}



2.5部署设计

2.6数据设计

**四、实施指南**