PassThemOn

软件架构文档

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 21/7/2021 | 1.0 | 初步完成软件架构文档 | 王浩天 陶昱丞 徐惠东 周昱宏 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 参考资料 4

2. 用例视图 4

3. 逻辑视图 4

3.1 概述 4

3.2 在构架方面具有重要意义的设计包 4

4. 进程视图 4

5. 部署视图 4

6. 实现视图 5

7. 技术视图 5

8. 数据视图（可选） 5

9. 核心算法设计（可选） 5

10. 质量属性的设计 5

软件架构文档

# 简介

## 目的

本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

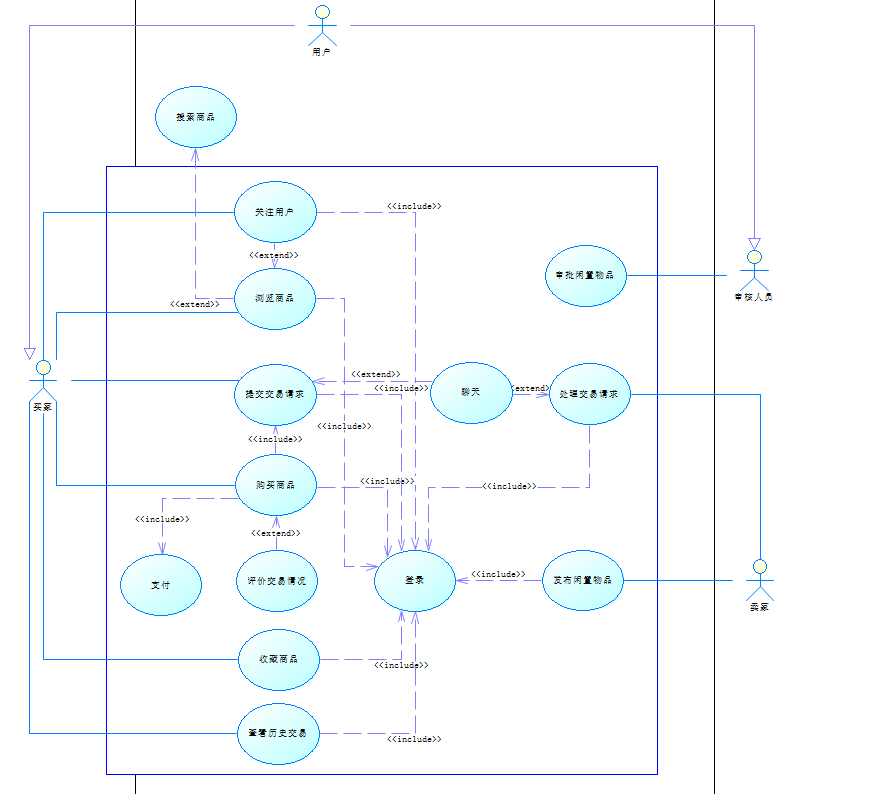
## 参考资料

《软件架构文档 template》

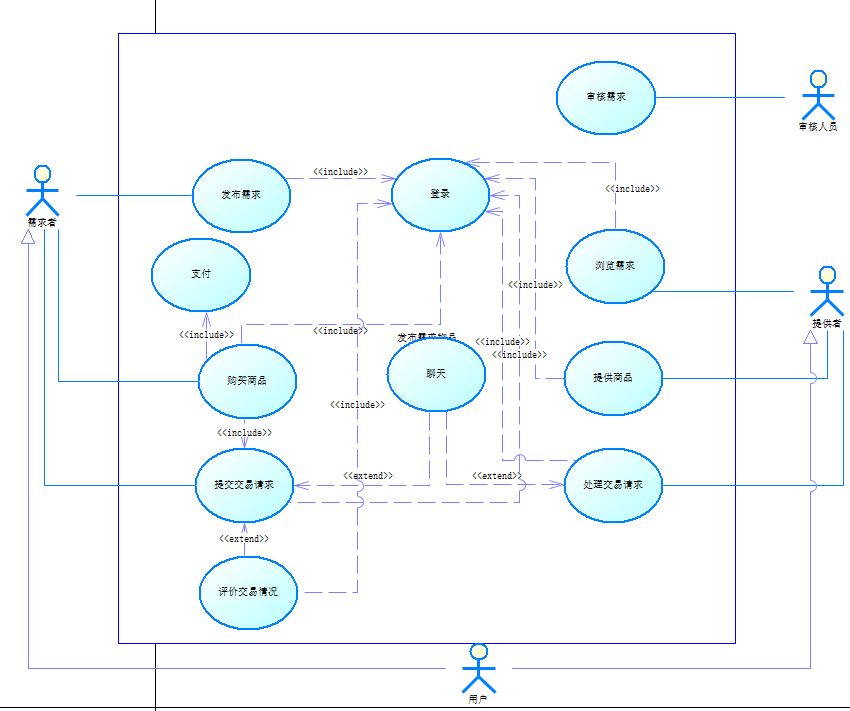
《软件工程原理》

# 用例视图

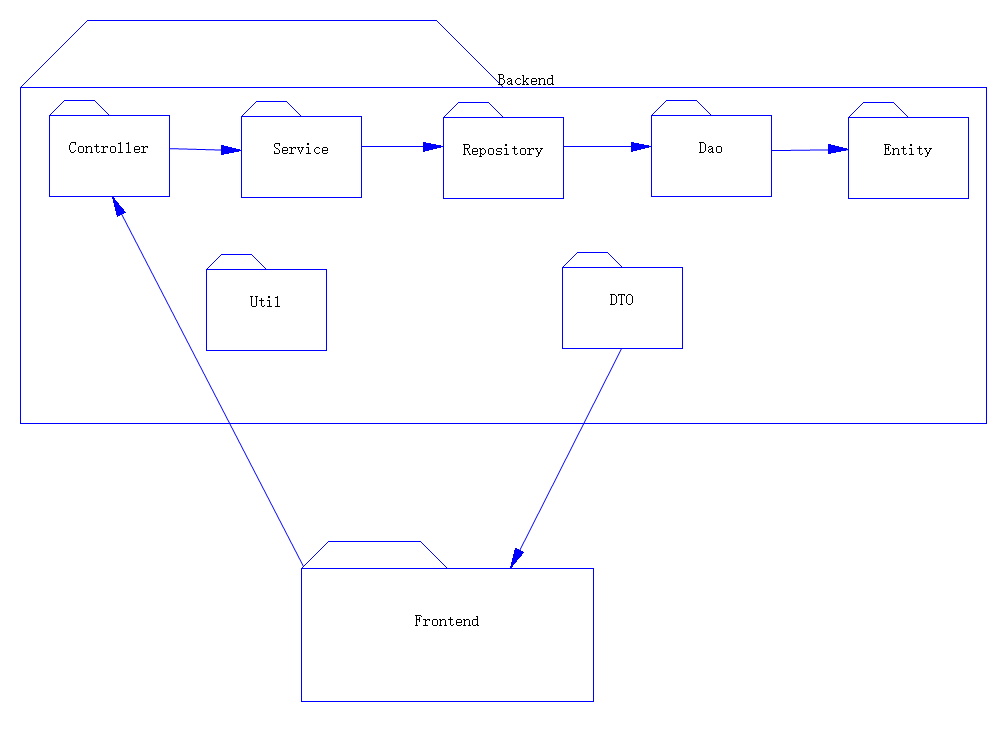
Use-case1二手物品交易：



Use-case2发布需求：



# 逻辑视图



## 概述

Backend:

1、Controller：连接前端和后端，接受前端发来的请求，同时向后端发送请求，负责业务调度。

2、Service：服务层，编写具体业务逻辑，更贴近于具体业务功能的要求，将对于数据访问的操作封装起来。

3、Repository:仓库，向上层屏蔽了数据来源和内部实现细节。

4、Dao：数据访问层，全称为data access object，属于一种比较底层，比较基础的操作，具体到对于某个表、某个实体的增删改查。

5、Entity：实体bean，用于ORM对象关系映射，一个实体映射成一张表，一般无业务逻辑代码，修改entity可以同步到数据库。

6、Util：功能包，针对本项目的工具类。主要用于存在项目各层都有可能出现、不好划分到某层中、出现频率较高的功能（类）。

7、DTO：数据传输对象（Data Transfer Object），是一种设计模式之间传输数据的软件应用系统。

Frontend：前端用户访问界面。

## 在构架方面具有重要意义的设计包

无

# 进程视图

由于采用框架处理，因而没有进程视图

# 部署视图

# 

# 实现视图

采取垂直分割方案，将业务大致分为发布闲置物品、购买闲置物品、聊天、支付、首页推荐5个子系统，将其分配给几个不同的开发人员，每个开发人员负责实现对应模块中从Web层到数据库层的所有开发，且每天进行集成。

# 技术视图

编程语言：后端使用JAVA,前端使用Javascript

开发工具：Intellij IDEA

框架：后端使用Spring boot/Spring Cloud框架，前端使用React Native框架。

数据库：mysql数据库

中间件：无

# 数据视图（可选）

无

# 核心算法设计（可选）

暂无

# 质量属性的设计

[说明软件架构如何促成诸如性能、可扩展性、可靠性、易用性、可移植性等所有系统能力（而非功能）的实现。如果这些特征具有特殊的意义（例如在安全性或保密性方面的意义），则应该对它们进行详细的说明。]