茶行天下

软件架构文档

版本 1.1

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2016/10/22 | 1.0 | 软件架构文档初稿 | 李桐宇、路丽菲、叶聪聪、孙晏、金初阳 |
| 2016/10/24 | 1.1 | 软件架构文档第一次修改稿 | 李桐宇、路丽菲、叶聪聪、孙晏、金初阳 |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 5

1.1 目的 5

1.2 范围 5

1.3 定义、首字母缩写词和缩略语 5

1.4 概述 5

2. 构架表示方式 5

3. 构架目标和约束 5

3.1 易用性 5

3.1.1 茶农 5

3.1.2 消费者 6

3.1.3 系统管理员 6

3.2 可靠性 6

3.2.1 平均故障间隔时间 6

3.2.2 可用时间百分比 6

3.2.3 平均修复时间 6

3.3 性能 6

3.3.1 部署环境约束 6

3.3.2 对事务的响应时间 6

3.3.3 并发数 6

3.4 可支持性 6

3.5 设计约束 6

3.5.1 软件语言 6

4. 用例视图 7

4.1 面向茶农的茶叶售卖子系统 7

4.1.1 管理茶产品 7

4.1.2 订单管理 7

4.1.3 发起众筹（预售） 7

4.1.4 发起众筹（现货） 7

4.1.5 参与众包 7

4.2 面向消费者的茶叶购买子系统 8

4.2.1 消费者查看茶产品 8

4.2.2 立即购买 8

4.2.3 购物车购买 8

4.2.4 加入购物车 8

4.2.5 发起众包 8

4.2.6 参与众筹（现货） 8

4.2.7 参与众筹（预售） 9

4.2.8 消费订单管理 9

4.3 面向系统管理员的管理运营子系统 9

4.3.1 管理用户 9

4.3.2 管理茶产品类型 9

4.3.3 查看数据分析 9

4.3.4 订单管理 9

5. 逻辑视图 10

5.1 概述 10

5.1.1 面向茶农的茶叶售卖子系统 11

5.1.2 面向消费者的茶叶购买子系统 11

5.1.3 茶产品交易运营管理子系统 12

6. 进程视图 13

7. 部署视图 13

8. 实施视图 14

9. 数据视图 15

10. 大小和性能 16

11. 质量 16

**软件架构文档**

# 简介

## 目的

本文档将从架构方面对茶行天下系统进行综合概述，其中会使用多种不同的架构视图来描述系统的各个方面。为系统的整体实现提供指导和依据。

## 范围

本文档适用于软件工程课程第三组、IST实验室的六名学生开发的茶行天下系统，将影响和指导下一迭代中该系统的开发过程。

## 定义、首字母缩写词和缩略语

|  |  |
| --- | --- |
| 定义及缩略语 | 含义 |
| MVC | Model，View和Control分离的三层架构 |
| B/S、C/S | 浏览器/服务器模式、客户端/服务器模式 |
| Ionic | 基于HTML和Angular JS的移动APP开发框架 |
| 众筹 | 由单个茶农发起的，多个消费者共同购买的茶叶交易；茶农可以对积压现货发起众筹，也可以预计产出的茶叶发起众筹。 |
| 众包 | 由单个消费者发起的，多个茶农共同供货的茶叶交易 |

## 概述

本文档将明确茶行天下系统的架构表示方式、架构的目标和约束、性能和质量等，并通过以下一系列视图来表示茶行天下系统的软件架构：用例视图、逻辑视图、进程视图、部署视图、实施视图和数据视图。

# 构架表示方式

本文档将通过以下一系列视图来表示茶行天下系统的软件架构：用例视图、逻辑视图、进程视图、部署视图、实施视图和数据视图。其中数据视图是由MySQL根据数据库表自动生成的，其他的视图都是通过PowerDesigner工具建立的UML模型。

# 构架目标和约束

## 易用性

### 茶农

考虑到众筹众包的流程相对复杂，茶农掌握茶叶售卖子系统的培训时间不应超过5h。

### 消费者

消费者掌握茶叶购买子系统的培训时间不应超过2h。

### 系统管理员

系统管理员掌握运营管理子系统的培训时间不应超过2h，同时运营管理子系统会将数据以图表的形式展示给管理员，界面友好。

## 可靠性

### 平均故障间隔时间

系统的平均故障间隔时间应大于7\*24h。

### 可用时间百分比

系统的可用时间百分比应大于99.9%。

### 平均修复时间

系统的平均修复时间应小于2h。

## 性能

### 部署环境约束

单节点部署、CPU个数为1个、内存大小为2G。

### 对事务的响应时间

在部署环境约束的前提下，平均响应时间不超过1s，最长响应时间不超过7s。

### 并发数

在部署环境约束的前提下，系统应支持100的并发数。

## 可支持性

茶叶售卖子系统应该支持在Android 4.0版本及以上运行；

茶叶购买子系统以及运营管理子系统能够支持各个PC操作系统(OS X，Window7+)通过Chrome浏览器进行访问。

## 设计约束

### 软件语言

本系统后台服务端采用Spring Framework（Java语言），MySQL数据库，网页端采用AngularJS作为框架，手机端采用Ionic框架。

# 用例视图

## 面向茶农的茶叶售卖子系统

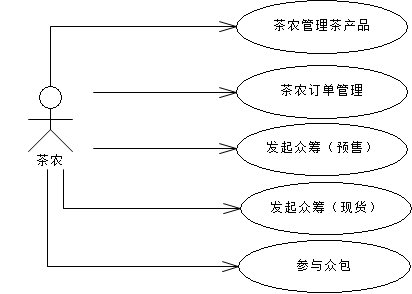


Figure ‑面向茶农的茶叶售卖子系统用例图

### 管理茶产品

茶农登录后，可以根据系统提供的模版生成或者（个性化）增加茶产品，或对已添加的茶产品进行修改上架下架等操作。然后将修改后的产品信息进行展示和条件搜索。

### 订单管理

茶农登录到系统后，对未发货，已发货，已完成等订单进行查看和条件筛选。或对已完成的订单进行删除，则往后将无法再看到该订单。

### 发起众筹（预售）

茶农登录到系统后，可以发起众筹，可以选择预售模式，填入产品信息，售价，众筹结束时间，定金，每份的数量等信息。到规定时间后，统计众筹是否成功，成功后则等到设定时间后交付全款，并等待茶农备货发货处理。

### 发起众筹（现货）

茶农登录到系统后，可以发起众筹，可以选择现货模式，填入产品信息，售价，众筹结束时间等信息。到规定时间后，统计众筹是否成功，成功后则等待茶农备货发货处理。

### 参与众包

茶农登录到系统后，查看消费者发起的正在进行的众包信息，根据自己的情况，参与众包，选择参与的数量等信息。到达众包结束时间后，统计结果，若众包成功，就茶农开始备货发货处理。

## 面向消费者的茶叶购买子系统

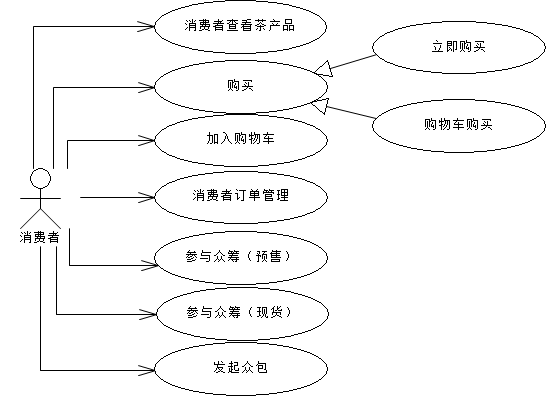


Figure ‑ 面向消费者的茶叶购买子系统用例图

### 消费者查看茶产品

消费者打开该系统后，可以查看茶产品的信息，同时对茶产品进行条件查找，并可以查看茶产品的详情信息。

### 立即购买

消费者登录后，填入相应的产品数量，收件地址等信息，确认后生成订单，支付成功后，等待茶农发货处理。

### 购物车购买

消费者登录后，进入购物车选择某些商品，填入收件地址等信息，确认后生成订单，支付成功后，等待茶农发货处理。

### 加入购物车

消费者登录后，填入相应的产品数量信息，把产品添加到购物车中，消费者可以到购物车中对已添加的商品进行查看或者购物操作。

### 发起众包

消费者登录后，填入商品信息，所需总量，价格等信息，发起众包功能。等到众包结束时间，系统判断众包是否成功，成功就消费者交付全款，然后等待农户进行发货处理。

### 参与众筹（现货）

消费者登录后，查看茶农发起的众筹信息，自主选择参与某些众筹，并交付全款等到众筹结束后，系统判断众筹是否成功，成功则等待茶农发货处理。

### 参与众筹（预售）

消费者登录后，查看茶农发起的众筹信息，自主选择参与某些众筹，并交付定金，等到众筹结束后，系统判断众筹是否成功，成功则等到规定的时间，交付尾款后，等待茶农发货处理。

### 消费订单管理

消费者登录后，查看待支付，待发货，已完成等类型的订单信息，并可以对商品进行确认收货的处理。

## 面向系统管理员的管理运营子系统

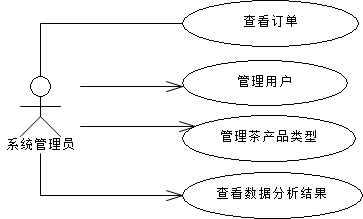


Figure ‑面向系统管理员的管理运营子系统用例图

### 管理用户

系统管理员可以登录到本子系统，可对用户进行增删查改，同时审批已经注册的茶农用户信息。

### 管理茶产品类型

系统管理员登录到本系统后，可对茶产品的类型进行增加，修改，删除等操作。

### 查看数据分析

系统管理员登录到本系统后，查看系统对产品的单价，销量等信息的预测。

### 订单管理

系统管理员登录到本系统后，查看系统对系统产生的订单进行查看，并进行条件查找等操作。

# 逻辑视图

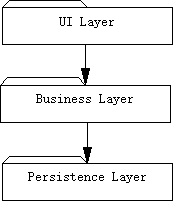


Figure 5‑1架构的逻辑视图

## 概述

本项目采用了MVC的架构风格，分为三层：表示层、业务逻辑层、数据持久层。表示层（UI层）采用Angular JS + HTML为用户提供接口以及与业务层交互，业务层使用了Spring MVC管理控制器，维护业务相关的类，以及数据持久层的操作；数据持久层通过使用 Hibernate对数据进行持久化管理。

### 面向茶农的茶叶售卖子系统

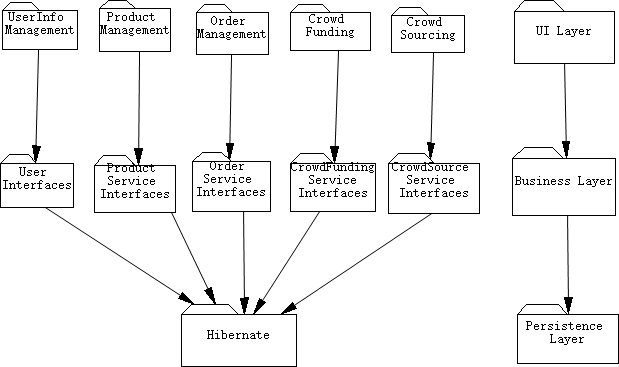


Figure 5‑2面向茶农的茶叶售卖子系统逻辑图

该子系统中，UI Layer使用Ionic框架，UI-Layer和Business Layer之间通过RESTful API通信，Business Layer通过Hibernate实现数据持久化。功能上实现了茶农产品管理，订单管理，个人管理以及众筹众包的处理等。

### 面向消费者的茶叶购买子系统

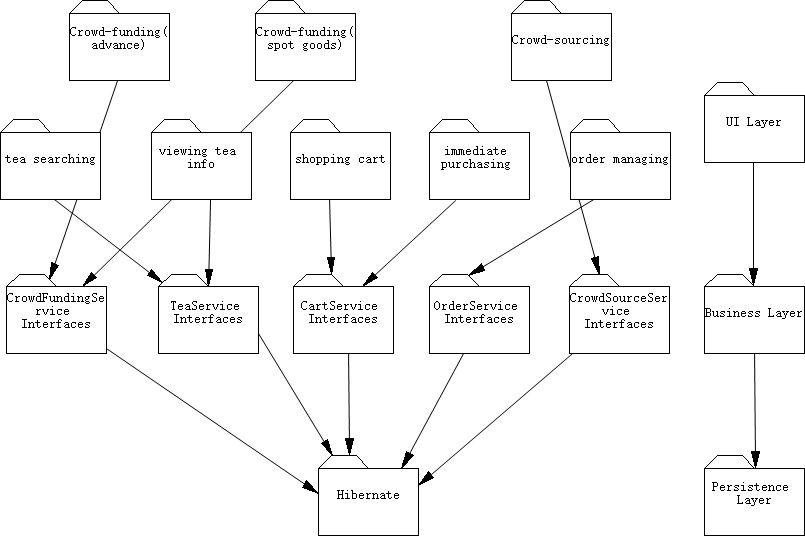


Figure 5‑2 面向消费者的茶叶购买子系统逻辑视图

在该子系统中，UI layer和Business layer之间通过RESTful API请求（数据格式为json）来传送数据，前端使用AngularJS展示页面，业务代码与前端代码解耦合，使得如果修改了后台的代码，前端的修改比较少。这样可以加快开发进度。

### 茶产品交易运营管理子系统

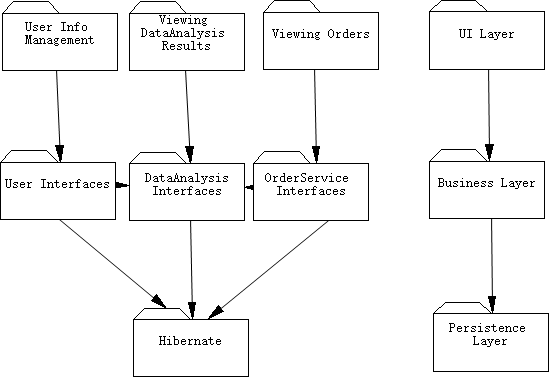


Figure 5‑3茶产品交易运营管理子系统逻辑图

# 进程视图

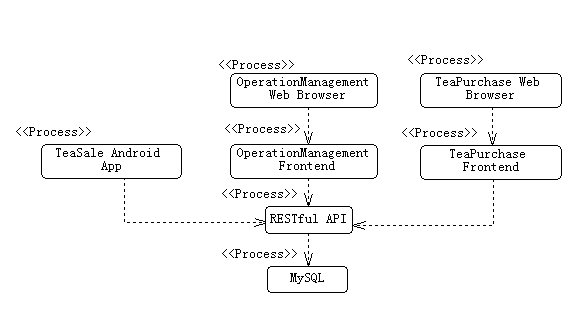


Figure 6‑1进程视图

由上图可知，当请求产生时，请求被发送给请求处理进程、处理后发送给server进行业务处理，业务逻辑控制进程将业务进行分发，由不同的server或同一server上的不同进程进行处理，并对数据库进行相应地读写。

# 部署视图

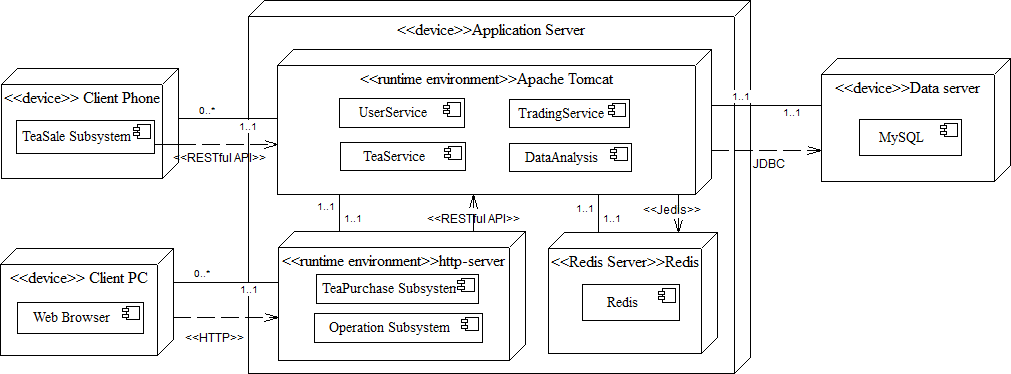


Figure 7‑1部署视图

由上图可以看出，茶行天下系统采用了RESTful API使得前后端解耦，用户通过手机或电脑上的浏览器进行访问，通过HTTP服务器通讯，得到应用服务器上的响应，并最终对MySQL数据服务器进行读写访问。

# 实施视图

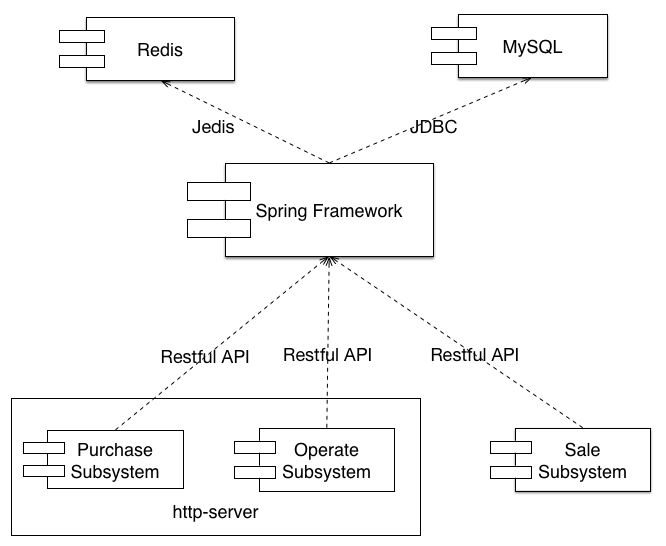


Figure 8‑1实施视图

茶行天下系统中的购买子系统和运营子系统部署在http-server上，通过浏览器访问，售卖子系统运行在手机上，三个子系统通过Restful API与服务器进行通信，并将服务器返回的数据以可视化的形式展现给用户。服务器使用Spring Framework框架，通过JDBC与MySQL数据库交互，完成数据持久化的操作。同时，茶行天下系统还使用内存数据库Redis提高数据访问效率，服务器通过Jedis与Redis进行通信。

# 数据视图

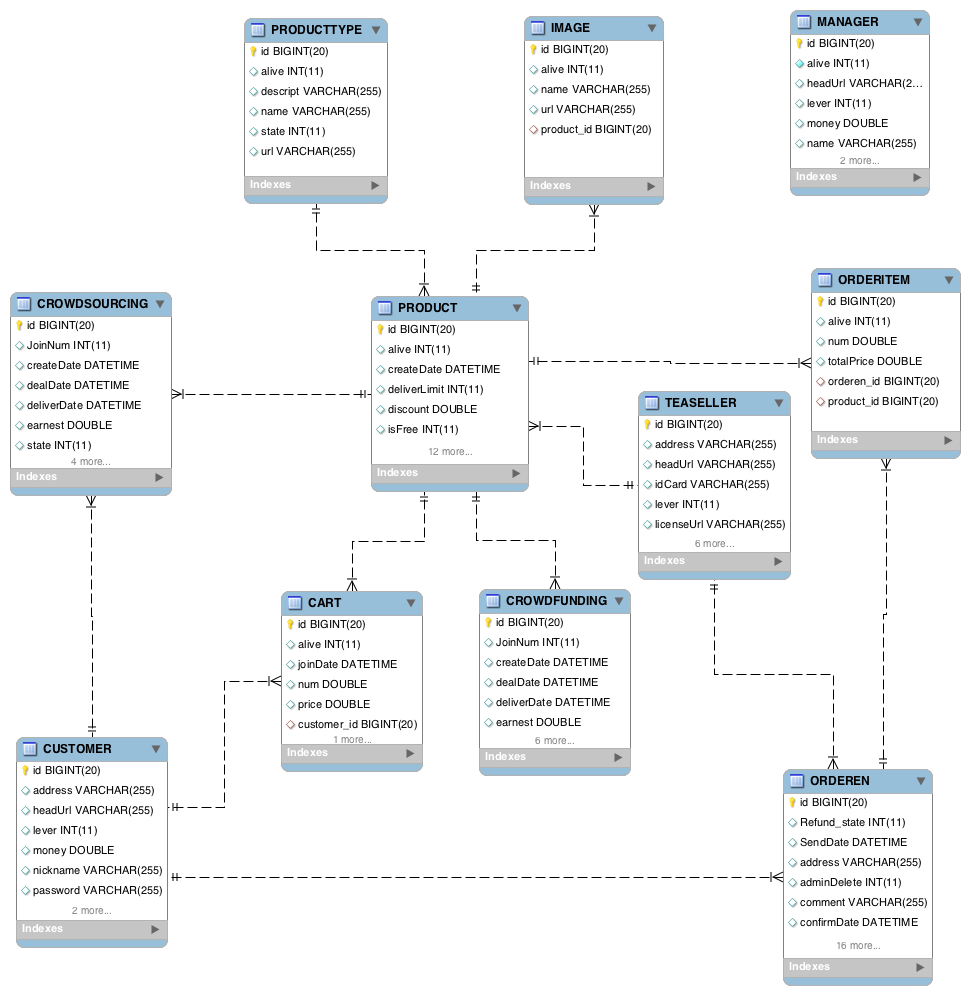


Figure 9‑1 ER图

关于系统的数据库，包括了购物车（Cart）、众筹（CrowdFunding）、众包（CrowdSourcing）、客户（Customer）、图片（Image）、管理员（Manager）、订单（OrderEn）、订单项（OrderItem）、产品（Product）、产品类型（ProductType）、茶农（TeaSaler）十一张表。

# 大小和性能

* 对事务的平均响应时间不超过1s，最长响应时间不超过7s
* 系统的可用时间百分比应大于99%
* 系统的平均修复时间应小于2h
* 系统平均故障间隔时间应大于7\*24h

# 质量

* 系统PC端支持各个主流操作系统（至少包括Windows，Mac OS）通过Chrome浏览器进行访问；
* 手机端支持Android 4.0及以上版本的适配；
* 开发语言为Java，Javascript、CSS和HTML，技术相对成熟。