|  |  |
| --- | --- |
| 产品名称Product name | 密级Confidentiality level |
| 设计说明书 | 内部公开 |
| 产品版本Product version | Total pages 共 17页 |
| 1.0 |

**天猫商城设计说明书**

|  |  |  |  |
| --- | --- | --- | --- |
| 拟制人： | 邹先锋 | 时间： | 2022/10/18 |
| 评审人： | 高嘉仪 | 时间： | 2022/10/21 |
| 批准人： | 周杭 | 时间： | 2022/10/22 |

**软通动力技术服务有限公司**

**版权所有 侵权必究**

**修订记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **修订内容** | **修订人** |
| 1.0 | 2022/10/18 | 初稿 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目录**

[1 研发背景 4](#_Toc117335409)

[2 阅读对象 4](#_Toc117335410)

[3 参考资料 4](#_Toc117335411)

[4 术语、缩略语 4](#_Toc117335412)

[5 概要设计 5](#_Toc117335413)

[5.1 需求概述 5](#_Toc117335414)

[5.2 系统功能整体流程图 6](#_Toc117335415)

[5.3 系统数据流图 6](#_Toc117335416)

[6 数据库逻辑结构设计 9](#_Toc117335417)

[7 系统可靠性设计 9](#_Toc117335418)

# 研发背景

为了熟练掌握SSM框架熟悉Java开发全流程所套用的一个有SSM框架制作的模仿天猫商城的项目网站。

仿天猫商城是一个基于Spring Boot的综合性B2C电商平台，数据管理后台包含商品管理，订单管理，类别管理，用户管理和交易额统计等模块，实现了对整个商城的一站式管理和维护。

项目使用IntelliJ IDEA开发，使用IntelliJ IDEA的版本控制检出功能，拉取项目即可。

项目数据库为MySQL 5.7版本，下载SQL文件并导入到数据库中即可。

使用IDEA打开项目后，在maven面板刷新项目，下载依赖包。

配置数据库连接并启动即可。

# 阅读对象

工作室校企联合授课讲师、相关工作室负责导师、项目开发小组成员

# 参考资料

无

# 术语、缩略语

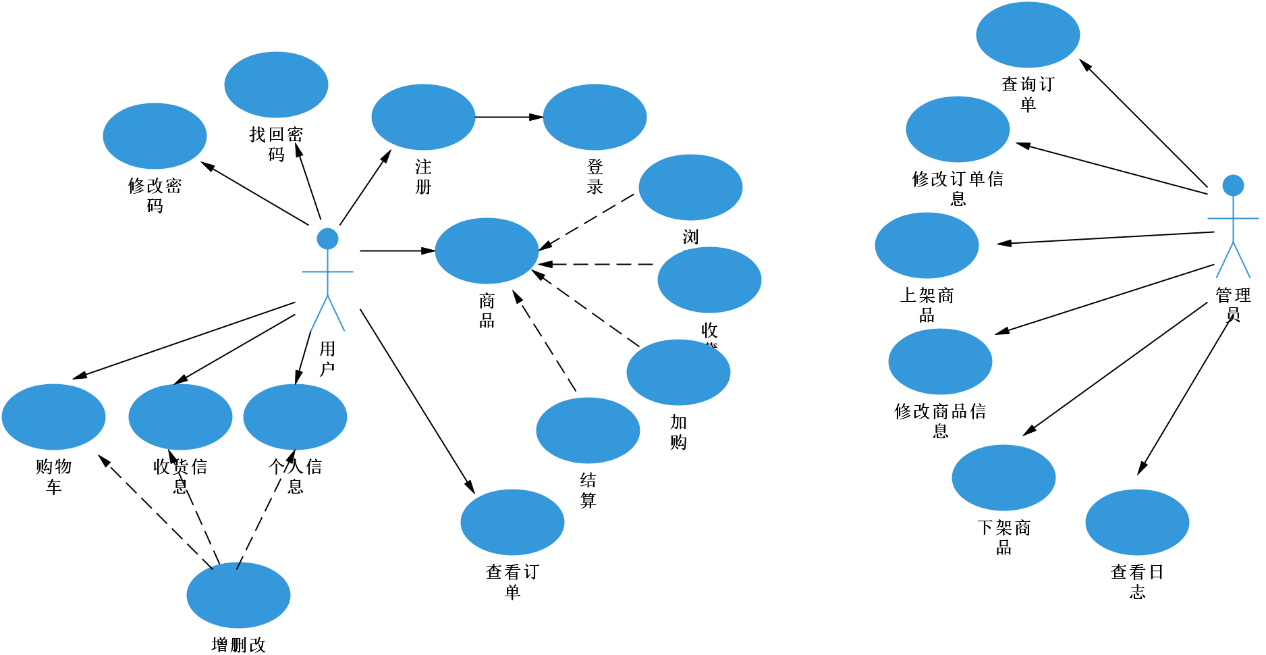
|  |  |
| --- | --- |
| 术语、缩略语 | 说明 |
| **MySQL** | MySQL是一个[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511?fromModule=lemma_inlink)，由瑞典[MySQL AB](https://baike.baidu.com/item/MySQL%20AB/2620844?fromModule=lemma_inlink) 公司开发，属于 [Oracle](https://baike.baidu.com/item/Oracle?fromModule=lemma_inlink) 旗下产品。MySQL 是最流行的[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511?fromModule=lemma_inlink)之一，在 [WEB](https://baike.baidu.com/item/WEB/150564?fromModule=lemma_inlink) 应用方面，MySQL是最好的 [RDBMS](https://baike.baidu.com/item/RDBMS/1048260?fromModule=lemma_inlink) (Relational Database Management System，关系数据库管理系统) 应用软件之一。MySQL是一种关系型数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL所使用的 SQL 语言是用于访问[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93/103728?fromModule=lemma_inlink)的最常用标准化语言。MySQL 软件采用了双授权政策，分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是[开放源码](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%BA%90%E7%A0%81/7176422?fromModule=lemma_inlink)这一特点，一般中小型和大型网站的开发都选择 MySQL 作为网站数据库。 |
| **Superset** | [Apache](https://so.csdn.net/so/search?q=Apache&spm=1001.2101.3001.7020) Superset是一个开源的、现代的、轻量级BI分析工具，能够对接多种数据源、拥有丰富的图表展示形式、支持自定义仪表盘，且拥有友好的用户界面，十分易用。 |
| **Hive** | hive是基于[Hadoop](https://baike.baidu.com/item/Hadoop/3526507?fromModule=lemma_inlink)的一个[数据仓库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E4%BB%93%E5%BA%93/381916?fromModule=lemma_inlink)工具，用来进行数据提取、转化、加载，这是一种可以存储、查询和分析存储在Hadoop中的大规模数据的机制。hive数据仓库工具能将结构化的数据文件映射为一张数据库表，并提供[SQL](https://baike.baidu.com/item/SQL/86007?fromModule=lemma_inlink)查询功能，能将[SQL语句](https://baike.baidu.com/item/SQL%E8%AF%AD%E5%8F%A5/5714895?fromModule=lemma_inlink)转变成[MapReduce](https://baike.baidu.com/item/MapReduce/133425?fromModule=lemma_inlink)任务来执行。Hive的优点是学习成本低，可以通过类似SQL语句实现快速MapReduce统计，使MapReduce变得更加简单，而不必开发专门的MapReduce应用程序。hive十分适合对数据仓库进行统计分析。 |
| **Hadoop** | Hadoop是一个由Apache基金会所开发的[分布式系统](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E7%B3%BB%E7%BB%9F/4905336?fromModule=lemma_inlink)基础架构。用户可以在不了解分布式底层细节的情况下，开发分布式程序。充分利用集群的威力进行高速运算和存储。Hadoop实现了一个[分布式文件系统](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E6%96%87%E4%BB%B6%E7%B3%BB%E7%BB%9F/1250388?fromModule=lemma_inlink)（ Distributed File System），其中一个组件是[HDFS](https://baike.baidu.com/item/HDFS/4836121?fromModule=lemma_inlink)（Hadoop Distributed File System）。HDFS有高[容错性](https://baike.baidu.com/item/%E5%AE%B9%E9%94%99%E6%80%A7/9131391?fromModule=lemma_inlink)的特点，并且设计用来部署在低廉的（low-cost）硬件上；而且它提供高吞吐量（high throughput）来访问[应用程序](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F/5985445?fromModule=lemma_inlink)的数据，适合那些有着超大数据集（large data set）的应用程序。HDFS放宽了（relax）[POSIX](https://baike.baidu.com/item/POSIX/3792413?fromModule=lemma_inlink)的要求，可以以流的形式访问（streaming access）文件系统中的数据。Hadoop的框架最核心的设计就是：[HDFS](https://baike.baidu.com/item/HDFS/4836121?fromModule=lemma_inlink)和[MapReduce](https://baike.baidu.com/item/MapReduce/133425?fromModule=lemma_inlink)。HDFS为海量的数据提供了存储，而MapReduce则为海量的数据提供了计算 。 |

# 概要设计

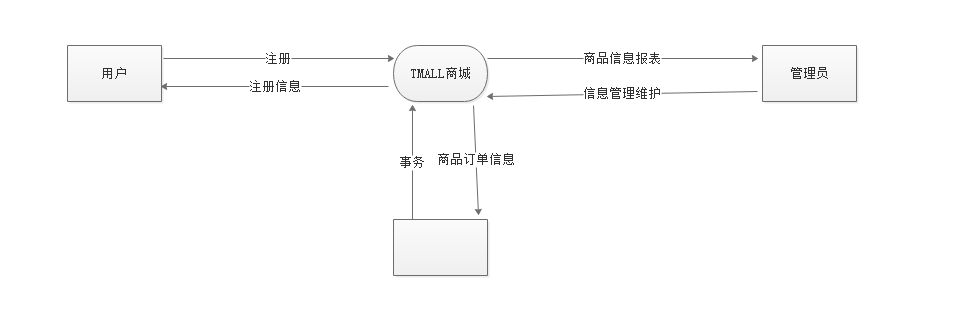
## 需求概述

需求设计主要参考天猫商城的购物流程：用户从注册开始，到完成登录，浏览商品，加入购物车，进行下单，确认收货，评价等一系列操作。

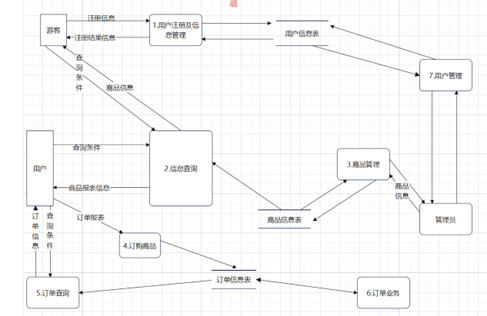
## 系统功能整体流程图



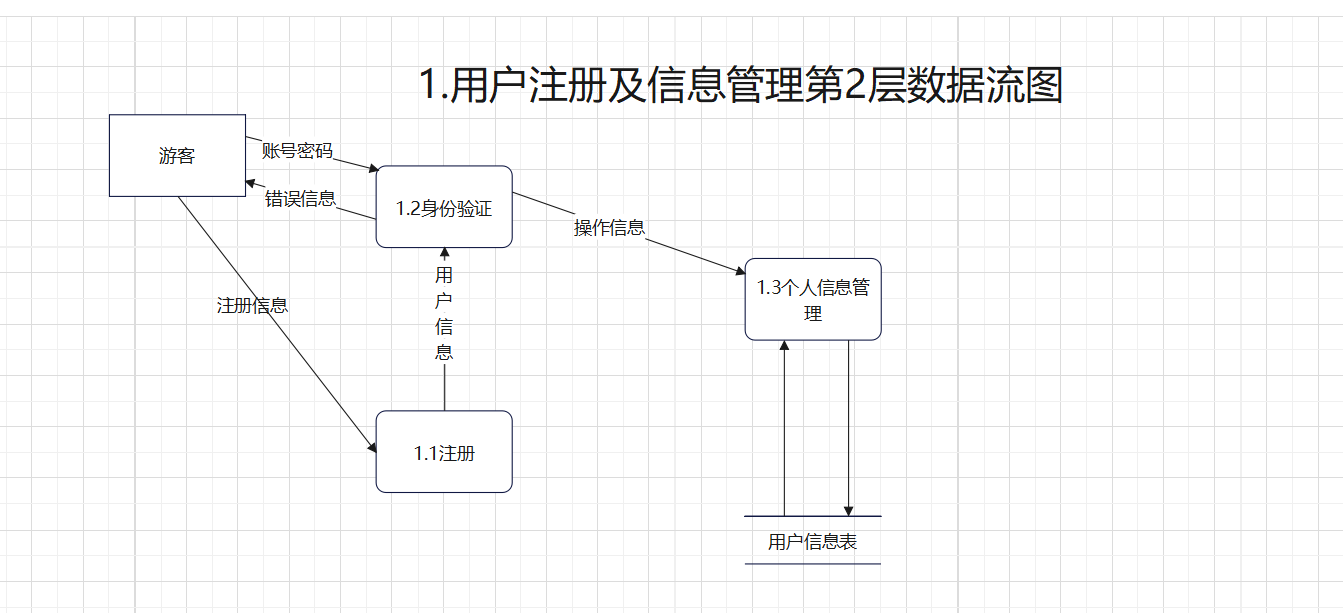
## 系统数据流图

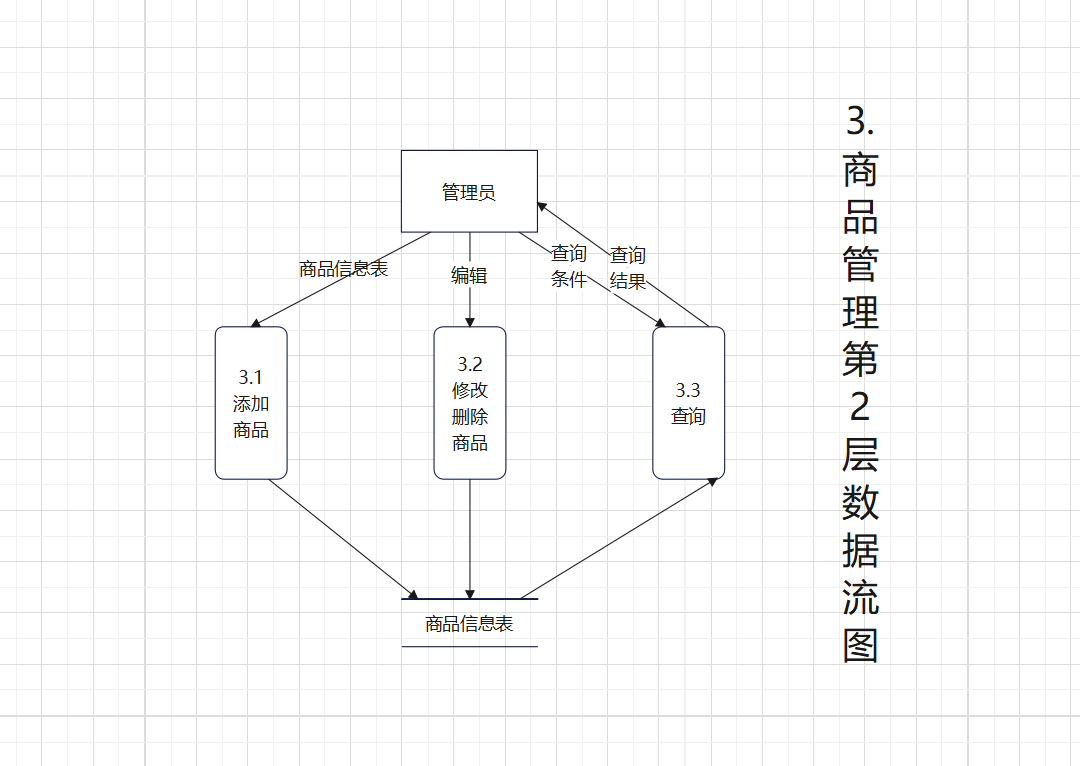


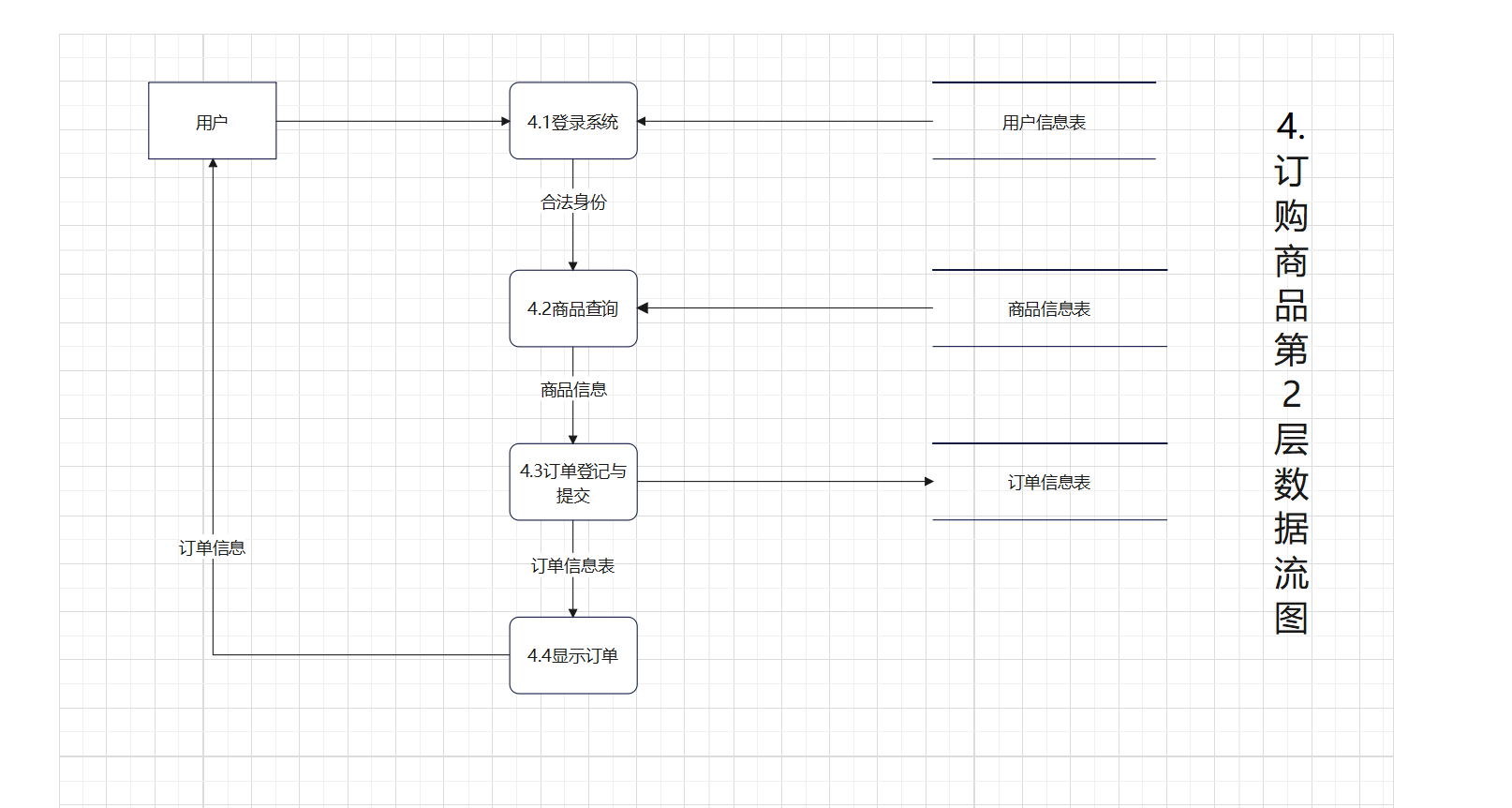
第0层数据流图

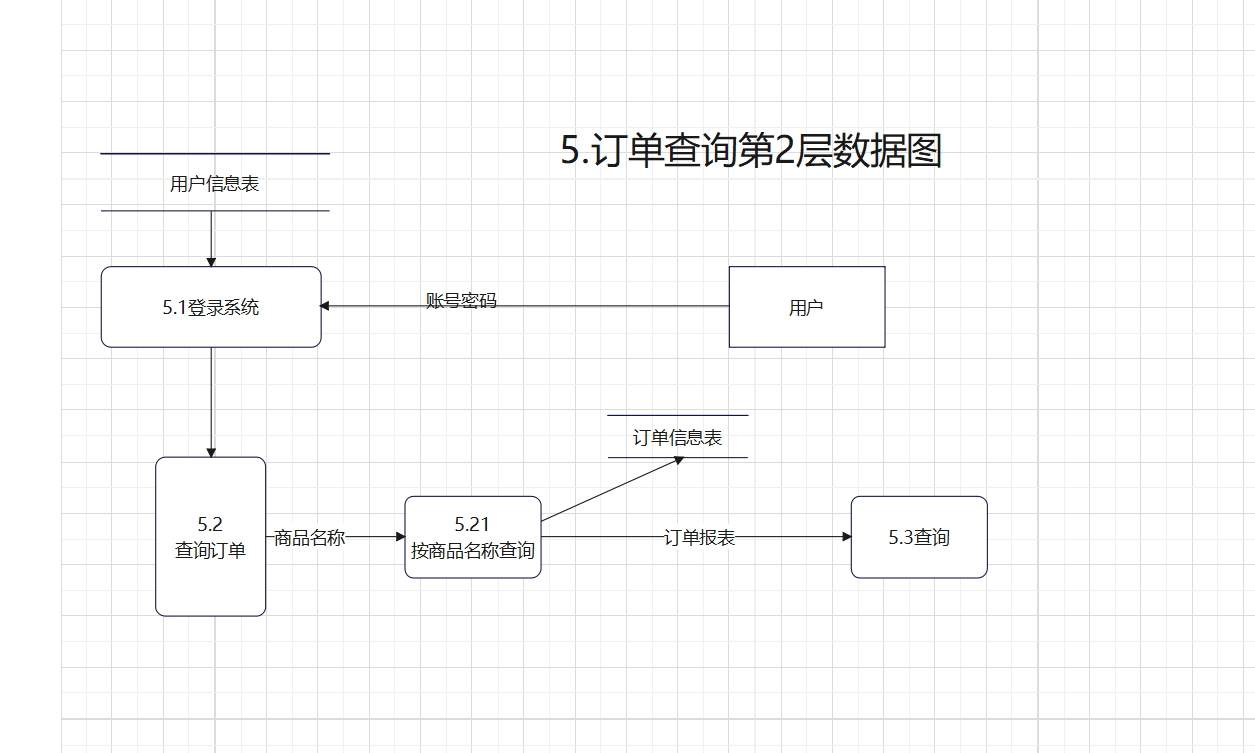


TMALL商城第一层数据流图









# 数据库逻辑结构设计

CREATE TABLE `address` (

`address\_areaId` char(6) NOT NULL COMMENT '地区编码',

`address\_name` varchar(50) NOT NULL COMMENT '省市名称',

`address\_regionId` char(6) NOT NULL COMMENT '父级省市id',

PRIMARY KEY (`address\_areaId`) USING BTREE,

KEY `address\_regionId` (`address\_regionId`) USING BTREE,

CONSTRAINT `address\_ibfk\_1` FOREIGN KEY (`address\_regionId`) REFERENCES `address` (`address\_areaId`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='地址表';

DROP TABLE IF EXISTS `admin`;

CREATE TABLE `admin` (

`admin\_id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '编号',

`admin\_name` varchar(25) CHARACTER SET utf8 COLLATE utf8\_bin NOT NULL COMMENT '账户名',

`admin\_nickname` varchar(50) NOT NULL DEFAULT '' COMMENT '昵称',

`admin\_password` varchar(50) CHARACTER SET utf8 COLLATE utf8\_bin NOT NULL COMMENT '密码',

`admin\_profile\_picture\_src` varchar(255) DEFAULT NULL COMMENT '头像地址',

PRIMARY KEY (`admin\_id`) USING BTREE,

UNIQUE KEY `un\_admin\_name` (`admin\_name`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='管理员表';

/\*Table structure for table `category` \*/

DROP TABLE IF EXISTS `category`;

CREATE TABLE `category` (

`category\_id` int(10) NOT NULL AUTO\_INCREMENT,

`category\_name` varchar(20) NOT NULL COMMENT '类别名称',

`category\_image\_src` varchar(255) NOT NULL COMMENT '类别图片',

`del\_flag` tinyint(1) DEFAULT '0' COMMENT '删除标识(1删除 0未删除）',

PRIMARY KEY (`category\_id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='类别表';

DROP TABLE IF EXISTS `product`;

CREATE TABLE `product` (

`product\_id` int(10) NOT NULL AUTO\_INCREMENT,

`product\_name` varchar(100) NOT NULL COMMENT '产品名称',

`product\_title` varchar(100) DEFAULT NULL COMMENT '产品标题',

`product\_price` decimal(10,2) DEFAULT NULL COMMENT '原价',

`product\_sale\_price` decimal(10,2) NOT NULL COMMENT '促销价',

`product\_create\_date` datetime NOT NULL COMMENT '创建日期',

`product\_category\_id` int(10) NOT NULL COMMENT '类别id',

`product\_isEnabled` tinyint(1) NOT NULL DEFAULT '0' COMMENT '是否可用',

PRIMARY KEY (`product\_id`) USING BTREE,

KEY `product\_ibfk\_1` (`product\_category\_id`) USING BTREE,

CONSTRAINT `product\_ibfk\_1` FOREIGN KEY (`product\_category\_id`) REFERENCES `category` (`category\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=90 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='产品表';

DROP TABLE IF EXISTS `productImage`;

CREATE TABLE `productImage` (

`productimage\_id` int(10) NOT NULL AUTO\_INCREMENT,

`productimage\_type` tinyint(1) unsigned NOT NULL COMMENT '类型(0:概述图片 1:详情图片)',

`productimage\_src` varchar(255) NOT NULL COMMENT '图片地址',

`productimage\_product\_id` int(10) NOT NULL COMMENT '产品id',

PRIMARY KEY (`productimage\_id`) USING BTREE,

KEY `productimage\_product\_id` (`productimage\_product\_id`) USING BTREE,

CONSTRAINT `productImage\_ibfk\_1` FOREIGN KEY (`productimage\_product\_id`) REFERENCES `product` (`product\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=1037 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='产品图片表';

DROP TABLE IF EXISTS `productOrder`;

CREATE TABLE `productOrder` (

`productorder\_id` int(10) NOT NULL AUTO\_INCREMENT,

`productorder\_code` varchar(30) NOT NULL COMMENT '订单号',

`productorder\_address` char(6) NOT NULL COMMENT '产品地址',

`productorder\_detail\_address` varchar(255) NOT NULL COMMENT '产品详细地址',

`productorder\_post` char(6) DEFAULT NULL COMMENT '邮政编码',

`productorder\_receiver` varchar(20) NOT NULL COMMENT '收货人',

`productorder\_mobile` char(11) NOT NULL COMMENT '联系方式',

`productorder\_pay\_date` datetime NOT NULL COMMENT '支付日期',

`productorder\_delivery\_date` datetime DEFAULT NULL COMMENT '发货日期',

`productorder\_confirm\_date` datetime DEFAULT NULL COMMENT '确认日期',

`productorder\_status` tinyint(1) NOT NULL COMMENT '订单状态(0:待付款 1:待发货 2:待确认 3:交易成功 4:交易关闭)',

`productorder\_user\_id` int(10) NOT NULL COMMENT '用户id',

PRIMARY KEY (`productorder\_id`) USING BTREE,

UNIQUE KEY `un\_productorder\_code` (`productorder\_code`) USING BTREE,

KEY `productorder\_address` (`productorder\_address`) USING BTREE,

KEY `productorder\_ibfk\_2` (`productorder\_user\_id`) USING BTREE,

CONSTRAINT `productOrder\_ibfk\_1` FOREIGN KEY (`productorder\_address`) REFERENCES `address` (`address\_areaId`),

CONSTRAINT `productOrder\_ibfk\_2` FOREIGN KEY (`productorder\_user\_id`) REFERENCES `user` (`user\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=222 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='产品订单表';

DROP TABLE IF EXISTS `productOrderItem`;

CREATE TABLE `productOrderItem` (

`productorderitem\_id` int(10) NOT NULL AUTO\_INCREMENT,

`productorderitem\_number` smallint(5) unsigned NOT NULL COMMENT '数量',

`productorderitem\_price` decimal(10,2) NOT NULL COMMENT '单价',

`productorderitem\_product\_id` int(10) NOT NULL COMMENT '关联产品id',

`productorderitem\_order\_id` int(10) DEFAULT NULL COMMENT '关联订单id',

`productorderitem\_user\_id` int(10) NOT NULL COMMENT '关联用户id',

`productorderitem\_userMessage` varchar(255) DEFAULT NULL COMMENT '用户备注',

PRIMARY KEY (`productorderitem\_id`) USING BTREE,

KEY `productorderitem\_product\_id` (`productorderitem\_product\_id`) USING BTREE,

KEY `productorderitem\_order\_id` (`productorderitem\_order\_id`) USING BTREE,

KEY `productorderitem\_user\_id` (`productorderitem\_user\_id`) USING BTREE,

CONSTRAINT `productOrderItem\_ibfk\_1` FOREIGN KEY (`productorderitem\_product\_id`) REFERENCES `product` (`product\_id`),

CONSTRAINT `productOrderItem\_ibfk\_2` FOREIGN KEY (`productorderitem\_order\_id`) REFERENCES `productOrder` (`productorder\_id`) ON DELETE CASCADE,

CONSTRAINT `productOrderItem\_ibfk\_3` FOREIGN KEY (`productorderitem\_user\_id`) REFERENCES `user` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=286 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='产品订单详细表';

DROP TABLE IF EXISTS `property`;

CREATE TABLE `property` (

`property\_id` int(10) NOT NULL AUTO\_INCREMENT,

`property\_name` varchar(25) NOT NULL COMMENT '属性名称',

`property\_category\_id` int(10) NOT NULL COMMENT '关联类别id',

PRIMARY KEY (`property\_id`) USING BTREE,

KEY `property\_category\_id` (`property\_category\_id`) USING BTREE,

CONSTRAINT `property\_ibfk\_1` FOREIGN KEY (`property\_category\_id`) REFERENCES `category` (`category\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=245 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='类别属性表';

DROP TABLE IF EXISTS `propertyValue`;

CREATE TABLE `propertyValue` (

`propertyvalue\_id` int(10) NOT NULL AUTO\_INCREMENT,

`propertyvalue\_value` varchar(100) NOT NULL COMMENT '属性值',

`propertyvalue\_property\_id` int(10) NOT NULL COMMENT '关联属性id',

`propertyvalue\_product\_id` int(10) NOT NULL COMMENT '关联产品id',

PRIMARY KEY (`propertyvalue\_id`) USING BTREE,

KEY `propertyvalue\_property\_id` (`propertyvalue\_property\_id`) USING BTREE,

KEY `propertyvalue\_product\_id` (`propertyvalue\_product\_id`) USING BTREE,

CONSTRAINT `propertyValue\_ibfk\_1` FOREIGN KEY (`propertyvalue\_property\_id`) REFERENCES `property` (`property\_id`) ON DELETE CASCADE,

CONSTRAINT `propertyValue\_ibfk\_2` FOREIGN KEY (`propertyvalue\_product\_id`) REFERENCES `product` (`product\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=716 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='产品属性管理表';

DROP TABLE IF EXISTS `review`;

CREATE TABLE `review` (

`review\_id` int(10) NOT NULL AUTO\_INCREMENT,

`review\_content` mediumtext NOT NULL COMMENT '内容',

`review\_createdate` datetime NOT NULL COMMENT '创建日期',

`review\_user\_id` int(10) NOT NULL COMMENT '关联用户id',

`review\_product\_id` int(10) NOT NULL COMMENT '关联产品id',

`review\_orderItem\_id` int(10) NOT NULL COMMENT '关联订单详细id',

PRIMARY KEY (`review\_id`) USING BTREE,

KEY `review\_user\_id` (`review\_user\_id`) USING BTREE,

KEY `review\_product\_id` (`review\_product\_id`) USING BTREE,

KEY `review\_orderItem\_id` (`review\_orderItem\_id`) USING BTREE,

CONSTRAINT `review\_ibfk\_1` FOREIGN KEY (`review\_user\_id`) REFERENCES `user` (`user\_id`),

CONSTRAINT `review\_ibfk\_2` FOREIGN KEY (`review\_product\_id`) REFERENCES `product` (`product\_id`),

CONSTRAINT `review\_ibfk\_3` FOREIGN KEY (`review\_orderItem\_id`) REFERENCES `productOrderItem` (`productorderitem\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=76 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='评论表';

DROP TABLE IF EXISTS `user`;

CREATE TABLE `user` (

`user\_id` int(10) NOT NULL AUTO\_INCREMENT,

`user\_name` varchar(25) CHARACTER SET utf8 COLLATE utf8\_bin NOT NULL COMMENT '用户名',

`user\_nickname` varchar(50) NOT NULL COMMENT '昵称',

`user\_password` varchar(50) CHARACTER SET utf8 COLLATE utf8\_bin NOT NULL COMMENT '密码',

`user\_realname` varchar(20) DEFAULT NULL COMMENT '姓名',

`user\_gender` tinyint(1) NOT NULL COMMENT '性别',

`user\_birthday` date NOT NULL COMMENT '出生日期',

`user\_address` char(6) NOT NULL COMMENT '所在地地址',

`user\_homeplace` char(6) NOT NULL COMMENT '家乡',

`user\_profile\_picture\_src` varchar(100) DEFAULT NULL COMMENT '用户头像',

`del\_flag` tinyint(1) DEFAULT '0' COMMENT '删除标识(1删除 0未删除）',

PRIMARY KEY (`user\_id`) USING BTREE,

UNIQUE KEY `un\_user\_name` (`user\_name`) USING BTREE,

KEY `user\_address` (`user\_address`) USING BTREE,

KEY `user\_homeplace` (`user\_homeplace`) USING BTREE,

CONSTRAINT `user\_ibfk\_1` FOREIGN KEY (`user\_address`) REFERENCES `address` (`address\_areaId`),

CONSTRAINT `user\_ibfk\_2` FOREIGN KEY (`user\_homeplace`) REFERENCES `address` (`address\_areaId`)

) ENGINE=InnoDB AUTO\_INCREMENT=23 DEFAULT CHARSET=utf8 ROW\_FORMAT=DYNAMIC COMMENT='用户表';

表1 admin

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| uname | varchar | 20 | 用户名 | not null |  |
| uphone | varchar | 50 | 手机号 | not null |  |
| password | varchar | 20 | 密码 | not null |  |
| email | varchar | 20 | 邮箱 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

表2 user

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| uname | varchar | 20 | 用户名 | not null |  |
| uphone | varchar | 50 | 手机号 | not null |  |
| password | varchar | 20 | 密码 | not null |  |
| email | varchar | 20 | 邮箱 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

表3 user\_info

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| uname | varchar | 20 | 用户名 | not null |  |
| uphone | varchar | 50 | 手机号 | not null |  |
| password | varchar | 20 | 密码 | not null |  |
| email | varchar | 20 | 邮箱 | not null |  |
| img | varchar | 30 | 头像 | not null |  |
| update\_time | timestamp |  | update\_time | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |

表4 file\_info

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| filename | varchar | 20 | 文件名 | not null |  |
| uphone | varchar | 50 | 手机号 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

表5 teacher

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| uname | varchar | 20 | 用户名 | not null |  |
| uphone | varchar | 50 | 手机号 | not null |  |
| password | varchar | 20 | 密码 | not null |  |
| email | varchar | 20 | 邮箱 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

表6 recycle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| time | varchar | 20 | 文件删除时间 | not null |  |
| way | varchar | 50 | 文件原路径 | not null |  |
| filename | varchar | 20 | 文件名 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

表7 sign

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 字段长度 | 描述 | 是否可为空 | 备注 |
| id | int | 10 | id | not null | 自增，主键 |
| time | varchar | 20 | 签到时间 | not null |  |
| date | varchar | 50 | 签到日期 | not null |  |
| create\_time | timestamp |  | create\_time | not null | DEFAULT  CURRENT\_TIMESTAMP |
| update\_time | timestamp |  | update\_time | null |  |

# 系统可靠性设计

镜像：CentOS 7.6

Hadoop版本：3.1.3

MySQL版本：5.7.37

IDEA版本：2021.3