

RookieB药房管理系统

一. 项目介绍

- 1. 这是一个不错的药房管理系统，包括 药品上下架，药品销售，药品管理，售后服务，客户售后评价等一系列服务，操作简单，功能全面。
- 2. **业务流程:**
 - 客户可进入商城浏览药品，购买商品和收藏时，需要通过注册方式注册账号，随后在邮箱中认证激活，方可登录进入商城浏览药品并购买，可在购物车中进行购买。
 - 店长可以在系统中快捷地对药房进行管理，统计每天的销售数量，方便可视化的记录每天的营业额。
 - 药房售货员可以使用本统进行工作,进行药品上下架，修改药品信息。

二.系统模块

1.后台模块

登录模块

- 1 -管理员通过手机号或者邮箱，密码登录后台
- 2 -密码和加密后进行比对
- 3 -按回车键进行登录
- 4 -登陆失败显示错误信息
 - 5 -- 用户名或密码错误，显示还有几次登录机会、
 - 6 --达到五次之后，再次登录显示账号已被锁定，请联系管理员!!!
- 7 -管理员账号信息由超级管理员（老板/人事部门）分配
- 8 -可选记住我功能并将该网页添加为书签，下次登录时直接进入管理页面
- 9 -记住我后，如果再次请求进入登录页面，直接跳转到主页面
- 10 -主页面显示管理员的名字
- 11 -管理员登录后台管理系统后，只显示与权限相关的页面，没有权限的页面一律不显示在页面中。
- 12 -安全退出后，下次登录无法使用记住我功能

防伪登陆 | 密码加密 | 记住我

- 1 1.导入Spring Security依赖，本次项目的后台管理使用SpringSecurity安全框架来对管理员进行身份验证和访问控制
- 2 Security:
- 3 **实现UserDetailsService，自定义验证规则和认证授权逻辑
- 4 **登录成功处理器:登陆成功后，清除之前所累积的登录失败次数，并将该用户的信息设置保存到Session中，跳转到主页面
- 5 **登陆失败处理器:
 - 6 -接受验证失败后抛出的异常信息，并给出相对应的错误信息，#特别的，如果是“用户名或密码错误”，则显示还有几次登录机会，最大次数5次
 - 7 -将错误信息返回到登陆页面中，提示给用户
- 8 * 配置自定义表单登录页面
- 9 * 配置权限拦截：哪些资源需要放行，那些需要拦截
- 10 * 记住我配置：记住我功能
- 11 * 退出登录配置
- 12 2. 引入BCryptPasswordEncoder，并在配置类中声明@Bean，交给Spring管理
- 13 BCryptPasswordEncoder:

- 14 -再设置用户密码/校验用户登录信息时，在后台对密码进行 加密/验证 操作
- 15 3. 记住我
- 16 RememberMeConfig配置类，导入PersistentTokenRepository类声明为@Bean交给Spring管理，在类中配置数据源，在方法中设置数据源和创建令牌表

角色模块

- 1 -可以搭配不同的权限，来设置不同的角色、管理员功能
- 2 老板-管理所有数据
- 3 人事-管理人员信息
- 4 会计-"统计收入,分析营业情况"
- 5 市场-负责拉客户
- 6 店员-负责订单管理和药品管理

权限模块

- 1 - 不同的权限意味着可以使用不同的功能，推出新的功能后，要及时的添加相应的权限，并分配给对应的角色
- 2 1, 首页, /index
- 3 2, 订单管理, /order/manage
- 4 3, 订单管理操作, /order/manage/operate
- 5 4, 查看营业情况, /business/condition
- 6 5, 查看注册用户, /show/register
- 7 6, 操作注册用户, /show/register/operate
- 8 7, 爆款药品, /hot/medicine
- 9 8, 爆款药品操作, /hot/medicine/operate
- 10 9, 系统管理, /system/all
- 11 10, 用户管理, /manager/all
- 12 11, 新建用户, /manager/add
- 13 12, 操作用户, /manager/operate
- 14 13, 角色管理, /role/all
- 15 14, 新建角色, /role/add
- 16 15, 操作角色, /role/operate
- 17 16, 权限管理, /permission/all
- 18 17, 新建权限, /permission/add
- 19 18, 操作权限, /permission/operate
- 20 19, 药店管理, /medicine/all
- 21 20, 药品信息, /medicine/manager
- 22 21, 新建药品, /medicine/add
- 23 22, 操作药品, /medicine/operate
- 24 23, 营业信息, /business/condition

后台首页

- 1 为了让使用本产品客户的操作更加方便，在首页提供了可能会经常使用的功能的快捷方式，不需要在一层一层的去寻找浪费客户的时间，进而提高了用户的办公效率
- 2 显示：订单管理、进日营业收益、查看注册用户、爆款药品

1.订单管理

- 1 - 用户提交订单后，会在该页面显示
- 2 - 显示信息：
 - 3 --订单编号
 - 4 --下单时间
 - 5 --总金额
 - 6 --订单状态
 - 7 --快递单号
 - 8 --操作
 - 9 --分页信息
- 10 - 管理员可以进行订单详情功能，对订单状态进行操作
 - 11 **状态一旦发货，自动生成快递单号
 - 12 --订单信息：订单编号、下单时间、订单状态、快递单号、收货地址
 - 13 --订单内容/费用信息：支付方式、总金额、药品图片、药品名称、数量、金额
 - 14 --联系人信息：会员姓名、性别、手机号、邮箱
- 15 - 保存/取消
- 16 - 分页显示 每页十条

2.进日营业收益

- 1 - 统计近一周营业额
- 2 - X轴：日期
- 3 - Y轴：营业额收入
- 4 - 内容：柱状图

3.查看注册用户

- 1 - 该页面显示在本药店注册的所有用户
- 2 - 显示信息：用户名 性别 手机号 邮箱 登录次数 用户状态 激活状态 操作
- 3 - 可进行查询用户操作
 - 4 -- KeyName: 关键字 可对用户名、性别、手机号、邮箱等关键字进行用户查询
- 5 - 操作
 - 6 -解冻/冻结
- 7 - 分页显示 每页十条

4.爆款药品

- 1 - 该功能展示出销量高的药品信息，低于10件库存的商品将被显示。
- 2 - 显示信息：药品图片 药品名称 药品描述 价格 库存 药品状态 药品类型
操作
- 3 - 操作：修改、启用/禁用 ---异步跟新，返回到该页面
- 4 - 新建功能：添加药品
 - 5 --药品图片 可选择要上传药品的素材图片
 - 6 --药品名称
 - 7 --药品类型：可在下拉框中选择对应的药品类型
 - 8 --根据需要设置价格、库存、开启状态
 - 9 --药品详情：会在客户端展示该药品的详情
- 10 - 分页显示 每页十条

系统管理

用户管理

- 1 - 展示所有管理员的相关信息:用户名 邮箱 电话号码 状态
- 2 - 操作: 详情、修改、分配角色、启用/禁用
- 3 --详情: 展示管理员名称、电话号、对应的角色以及角色所拥有的权限
- 4 --修改: 可对管理员的用户名、密码、电子邮箱、联系电话、用户状态进行修改
- 5 ---同步跟新
- 6 --分配角色: 给管理员分配搭配好权限的角色, 可以拥有不同的功能
- 7 ---同步更新
- 8 --启用禁用: 设置该管理员的登录状态, **true**可登录/**false** 不可登录效果
- 9 ---异步跟新, 返回到该页面
- 10 -保存和返回都要跳转到用户管理主页面
- 11 - 分页显示 每页十条

角色管理

- 1 - 展示已经搭配好权限的角色信息: 角色名称 角色功能 操作
- 2 - 新建: 新建角色名称 和 角色功能描述
- 3 - 操作: 修改、分配权限、删除
- 4 --修改: 更改角色名称 和 角色功能描述
- 5 --分配权限: 可以根据实际情况对该角色的权限进行增加或者删除, 增加或减少使用功能
- 6 --- 显示所有的权限信息, 但是已拥有权限前面复选框要选中, 没有权限前面的复选框不选中
- 7 --删除: 删除该角色信息 要进行确认证
- 8 - 保存和返回都要跳转到角色管理主页面

权限管理

- 1 - 展示权限信息: 权限名称 权限描述 操作
- 2 - 分页显示 每页十条
- 3 - 新建: 新建权限名称 和 权限功能
- 4 - 操作:
- 5 -- 修改: 修改权限名称 和 权限功能
- 6 -- 删除: 删除该权限信息 要进行确认证
- 7 - 保存和返回都要跳转到角色管理主页面

->系统管理模块

• 用户管理

- 1 1.新建用户: 超级管理员根据管理员信息分配账户
- 2 2.用户详情: 查看该管理员的详细信息
- 3 3. 修改: 修改管理员的信息
- 4 4.分配角色: 给管理员分配对应的角色
- 5 5.启用/禁用: 设置该管理员账号的状态

• 角色管理

- 1 1.新建: 创建新的角色并描述该角色的功能
- 2 2. 修改: 修改角色的信息
- 3 3.分配权限: 给该角色分配对应的权限
- 4 4.删除: 删除该角色

- 权限管理

- | | |
|---|--------------------------|
| 1 | 1. 新建: 创建新的权限并设置该权限的访问路径 |
| 2 | 2. 修改: 修改权限的信息 |
| 3 | 3. 删除: 删除该权限 |

药店管理模块

药品信息管理

- | | |
|----|--|
| 1 | - 该页面显示药店所有药品的详细信息 |
| 2 | - 显示信息: 药品图片、药品名称、药品描述、价格、库存、药品状态、药品类型、操作 |
| 3 | - 新建: |
| 4 | -- 药品图片: 可上传图片文件, 上传之后异步显示到右侧图片框内, 并且在后台生成文件夹路径、文件名称、文件名 |
| 5 | -- 药品名称 |
| 6 | -- 药品类型 |
| 7 | -- 价格 |
| 8 | -- 库存 |
| 9 | -- 状态 |
| 10 | -- 药品详情: 添加富文本插件, 可以显示不同样式的药品描述信息 |
| 11 | - 修改: 同步到表单页面, 对药品的信息做修改, 点击保存 同步到药品信息管理主页面 |
| 12 | - 保存和返回都要跳转到角色管理主页面 |

营业信息

- | | |
|---|-------------|
| 1 | - 统计近一周营业额 |
| 2 | - X轴: 日期 |
| 3 | - Y轴: 营业额收入 |
| 4 | - 内容: 柱状图 |

2.前台模块

登录 | 注册 | 退出 模块

- | | |
|----|--|
| 1 | 1. 进入登录页 可进行登录和注册操作 |
| 2 | - 登录: 手机号/邮箱、密码、验证码登录 可选择是否十天内免登录功能, 登录异步请求, 成功跳转, 不成功显示错误信息 |
| 3 | - 用户可以在不登录的状态下进行商品的浏览和查看详情 |
| 4 | - 点击收藏、购买按钮需要跳转到登录页面进行登录操作 |
| 5 | * 登录认证: |
| 6 | 1. 检查用户是否存在 |
| 7 | 2. 查看状态: 到了五次直接就显示冻结了 不在判断了 |
| 8 | 3. 校验密码: 密码不正确 登录次数累加 一共五次, 到达五次更改用户登录状态为 false, 下次登录显示已冻结 |
| 9 | 4. 查看激活状态: 查看用户是否激活 |
| 10 | - 注册: 需要用户输入个人信息 |
| 11 | -- 用户名 |
| 12 | -- 密码 |
| 13 | -- 邮箱 |
| 14 | -- 手机号 |
| 15 | -- 性别 |
| 16 | -- 出生日期 |

```

17         --验证码 进行判断
18         * 注册认证：
19             ***验证用户名是否重复
20             ***验证手机是否重复
21             ***验证邮箱是否重复
22             ***密码加密
23         --- 注册通过后，需要登录邮箱进行激活认证，激活成功后才能登录药房
24         ***设置用户状态FALSE 等待激活
25         *** 保存激活码 激活时比对
26         ***给用户的邮箱发送一封邮件，该邮件包含一个链接，连接中包含激活码
27         ***发送激活邮件
28     -退出：清除session数据、清除cookie数据,跳转到登录页

```

防伪登陆

```

1  本次项目的用户端使用Filter实现防伪登陆
2  -@webFilter("/")：对所有的请求路径进行拦截
3  -filterChain.doFilter(servletRequest, servletResponse)；判断指定放行的路径
4  如果请求的不是默认放行的资源，判断当前会话是否存在用户信息，存在放行，不存在封装错误信息，重定向到登录页面

```

记住我

```

1  登录认证通过后，进入验证码和记住我功能模块
2  获取从前台传入的数据user,loginAct,loginPwd, CheckCode, isRemPwd, request, response
3  - 将用户输入的验证码和验证码类存入session的验证码进行比对，失败抛出异常，封装错误数据以json格式返回到登录页面。
4  - 成功判断isRemPwd的值是否需要记住我功能，记不记住都要把用户信息发送到session中
5  --记住：将用户的账号、密码信息封装成Cookie对象并设置过期时间、路径、响应给浏览器保存

```

主页面

```

1  * 未登录
2  - 登录 | 注册
3  - 搜索框：用户可以输入药品名称和药品类型进行模糊查询，查询结果分页显示 每页十条
4  - 对药品类型进行分类形成导航栏，用户可以点击查看对应类型的药品信息,实现回车键搜索 和 点击按钮搜索 功能。
5  - 点击查看详情进入药品详情页面
6      -- 药品信息：显示药品名称，图片信息，价格，已售库存信息，是否收藏，提示信息
7      -- 药品评价：显示已购买用户的评价信息  用户名称，评价时间，评论内容
8      -- 药品详情：药品功效以及注意事项
9      -- 买药须知
10 * 已登录
11 - 欢迎回来 “用户名称”
12 - 个人信息
13 - 购物车
14 - 我的订单
15 - 安全退出
16 - 药品详情页面：
17     --如果已购买此商品时，展示评价功能。
18     --实时显示收藏状态  可点击进行收藏状态的改变      --异步更新

```

个人信息

- 1 展示用户的个人信息
- 2 -用户名
- 3 -密码：密文
- 4 -新密码
- 5 -确认新密码
- 6 -Email
- 7 -手机号
- 8 -性别
- 9 -出生日期
- 10 -收货地址
- 11 - 确定修改
- 12 -- 对表单元素进行检查 除新密码外不能为空
- 13 -- 新密码为空时，判定不修改密码
- 14 -- 新密码长度大于六位
- 15 -- 同步更新

购物车

- 1 用户点击收藏按钮后，将信息添加到数据库中，并且回显到页面中
- 2 - 显示信息：药品名称、价格、收藏数量信息、查看详情、悬浮框
- 3 - 收藏数量信息：
- 4 -- 数字框，可进行变动
- 5 --- 最小值1：小于1时判断是否取消收藏该商品
- 6 --- 最大值5：大于最大值不再进行变动
- 7 - 悬浮框：药品名称，数量，价格，总金额，去结算
- 8 --关联数字框：当数字框发生变化时，将数据同步更新到悬浮框内，价格和总金额也随之改变
- 9 - 全选：设置全选和全不选，悬浮框显示选中商品的信息
- 10 - 去结算：统计选中药品的信息和数量，跳转到订单页面

订单页

- 1 - 显示收货地址：获取当前用户的收货地址
- 2 -- 添加新的收货地址：点击按钮，清空收货地址文本框，通过三级地址菜单栏可以更换当前收货地址，最后再添加详细地址信息。
- 3 - 显示购物车中选中的商品、数量、价格等信息
- 4 - 显示最终的实付款金额、收货地址和当前登录用户的名称
- 5 - 提交订单按钮
- 6 -- 获取该订单页的全部信息，提交到后台生成订单信息
- 7 -- 后台自动生成默认信息
- 8 -- 添加到数据库中
- 9 - 跳转到我的订单页，显示所有的订单信息

我的订单

- 1 - 按时间倒序显示当前登录用户的所有订单信息：日期 订单编号 总金额 收货地址
- 2 - 每个订单信息独立显示，包括订单中所有的商品信息
- 3 - 未发货的订单后面显示未发货
- 4 - 已发货的订单后面显示已发货、快递单号、评论功能

评论

- 1

订单发起后，当后台管理员更改订单状态为发货之后，则判定该订单已完成，授予该用户对于订单商品的评论权
- 2

- 点击去评论：跳转到该药品详情页面，并且页面显示评价模块
- 3

- 评价：用户点击<添加评价文本框>，动态展示保存和取消按钮。用户可输入评论内容点击保存按钮发布评论信息，异步更新到药品评价的模块中的第一条信息
- 4

- 刷新页面后，评论表、用户表和药品三表联查该药品对应的评论信息，并按时间倒序排序

三.数据库设计

一个管理员：	多个角色	
一个角色：	多个权限	
一个用户：	多个订单	
一个订单	多个商品	
一个用户	多条评论	
一个订单	多条评论	
一个用户的	一个商品	多条评论

用户表:

```
1 DROP TABLE IF EXISTS `user`;
2 CREATE TABLE `user` (
3   `uid` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4   `username` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
  NULL,
5   `password` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
  NULL,
6   `birthday` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
  NULL,
7   `sex` varchar(4) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
8   `phoneNum` varchar(11) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
9   `email` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
10  `status` tinyint(1) NOT NULL,
11  `loginNum` int(11) NULL DEFAULT 0,
12  `active` tinyint(1) NOT NULL,
13  `activeCode` varchar(32) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
14  `address` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
15  PRIMARY KEY (`uid`) USING BTREE
16 ) ENGINE = InnoDB AUTO_INCREMENT = 3 CHARACTER SET = utf8 COLLATE =
  utf8_general_ci ROW_FORMAT = Dynamic;
17
```


管理员表:

```
1 DROP TABLE IF EXISTS `manager`;
2 CREATE TABLE `manager` (
3   `mid` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4   `name` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
5   `password` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
  NULL,
6   `phoneNum` varchar(11) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
  NULL,
7   `email` varchar(30) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
8   `LoginNum` int(11) NULL DEFAULT 0 COMMENT '登录次数',
9   `status` tinyint(1) NOT NULL,
10  PRIMARY KEY (`mid`) USING BTREE
11 ) ENGINE = InnoDB AUTO_INCREMENT = 4 CHARACTER SET = utf8 COLLATE =
  utf8_general_ci ROW_FORMAT = Dynamic;
12
```

角色表

```
1 DROP TABLE IF EXISTS `manager_role`;
2 CREATE TABLE `manager_role` (
3   `mid` int(11) NOT NULL,
4   `rid` int(11) NOT NULL
5 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT =
  Dynamic;
6
```

权限表

```
1 DROP TABLE IF EXISTS `permission`;
2 CREATE TABLE `permission` (
3   `pid` int(11) NOT NULL AUTO_INCREMENT,
4   `permissionName` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci
  NOT NULL,
5   `permissionDesc` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci
  NOT NULL,
6   PRIMARY KEY (`pid`) USING BTREE
7 ) ENGINE = InnoDB AUTO_INCREMENT = 24 CHARACTER SET = utf8 COLLATE =
  utf8_general_ci ROW_FORMAT = Dynamic;
8
```

药品表 medicine: 一个药品可以对应多条评论

```

1 DROP TABLE IF EXISTS `medicine`;
2 CREATE TABLE `medicine` (
3   `mid` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4   `medicineName` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci
   NULL DEFAULT NULL,
5   `medicineDesc` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci
   NULL DEFAULT NULL,
6   `price` double NOT NULL,
7   `stock` int(11) NOT NULL,
8   `status` tinyint(4) NULL DEFAULT NULL,
9   `mImage` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL,
10  `cid` int(11) NOT NULL,
11  PRIMARY KEY (`mid`) USING BTREE
12 ) ENGINE = InnoDB AUTO_INCREMENT = 111 CHARACTER SET = utf8 COLLATE =
   utf8_general_ci ROW_FORMAT = Dynamic;
13

```

药品类型表:(暂且认为一个药品对应一个药品类型)

```

1 DROP TABLE IF EXISTS `category`;
2 CREATE TABLE `category` (
3   `cid` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4   `cName` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
   DEFAULT NULL,
5   PRIMARY KEY (`cid`) USING BTREE
6 ) ENGINE = InnoDB AUTO_INCREMENT = 11 CHARACTER SET = utf8 COLLATE =
   utf8_general_ci ROW_FORMAT = Dynamic;
7

```

评论表

```

1 DROP TABLE IF EXISTS `remark`;
2 CREATE TABLE `remark` (
3   `remarkId` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
4   `remarkContent` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci
   NOT NULL,
5   `remarkDate` date NOT NULL,
6   `remarkOwn` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT
   NULL,
7   `mid` int(11) NOT NULL,
8   PRIMARY KEY (`remarkId`) USING BTREE
9 ) ENGINE = InnoDB AUTO_INCREMENT = 8 CHARACTER SET = utf8 COLLATE =
   utf8_general_ci ROW_FORMAT = Dynamic;
10

```

管理员角色表

```
1 DROP TABLE IF EXISTS `manager_role`;
2 CREATE TABLE `manager_role` (
3   `mid` int(11) NOT NULL,
4   `rid` int(11) NOT NULL
5 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT =
  Dynamic;
6
```

角色权限表

```
1 DROP TABLE IF EXISTS `role_permission`;
2 CREATE TABLE `role_permission` (
3   `rid` int(11) NOT NULL,
4   `pid` int(11) NOT NULL
5 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT =
  Dynamic;
6
```

收藏表

```
1 DROP TABLE IF EXISTS `favorite`;
2 CREATE TABLE `favorite` (
3   `uid` int(11) NULL DEFAULT NULL COMMENT '用户id',
4   `mid` int(11) NULL DEFAULT NULL COMMENT '药品id',
5   `unit` int(11) NULL DEFAULT 1 COMMENT '收藏件数'
6 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT =
  Dynamic;
7
```

订单表

```
1 DROP TABLE IF EXISTS `order`;
2 CREATE TABLE `order` (
3   `orderNo` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NOT NULL
  COMMENT '订单编号',
4   `orderDate` date NOT NULL COMMENT '下单时间',
5   `orderDateTime` datetime(0) NULL DEFAULT NULL,
6   `totalMoney` decimal(10, 0) NOT NULL COMMENT '总金额',
7   `orderStatus` tinyint(4) NULL DEFAULT NULL COMMENT '订单状态',
8   `deliveryNum` varchar(14) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL COMMENT '快递单号',
9   `address` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL,
10  PRIMARY KEY (`orderNo`) USING BTREE
11 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT
  = Dynamic;
12
```

订单详情表

```
1 DROP TABLE IF EXISTS `orderdetails`;
2 CREATE TABLE `orderdetails` (
3   `orderNo` varchar(255) CHARACTER SET utf8 COLLATE utf8_general_ci NULL
  DEFAULT NULL COMMENT '订单编号',
4   `medicineId` int(11) NOT NULL COMMENT '药品id',
5   `unit` int(11) NULL DEFAULT NULL COMMENT '单位',
6   `uid` int(11) NOT NULL COMMENT '用户id'
7 ) ENGINE = InnoDB CHARACTER SET = utf8 COLLATE = utf8_general_ci ROW_FORMAT =
  Dynamic;
8
```

登录令牌表：自动生成