项目部署说明书

1.项目简介

项目名称:智能售货机管理系统

前端技术栈: Vue 3 + Element Plus + Axios

后端技术栈: Spring Boot 3 + Spring Web + Spring Data JPA +

MySQL

通信方式:前后端分离,使用 RESTful API

部署环境: Windows, Node.js + Java 17 + Maven + MySQL

2.项目结构

```
system/
                  # 后端 Maven 构建文件
├─ pom.xml
 - springboot/
                  # Spring Boot 后端项目根目录
  └─ src/
     └─ main/
        ├─ java/
          └─ com/example/
             ├─ bean/
                           # 实体辅助类
(例如包装类、常量类等)
         — common/
                           # 公共类、统一
返回封装等
        # 控制器类(处
理 REST API 请求)
           ├─ dto/
                           # 数据传输对象
DTO
           ├─ entity/
                           # 实体类(映射
数据库表)
             ├─ factory/
                           # 工厂类
```

```
├─ interfaces/ # JPA
Repository 接口
                                 # 业务逻辑层
                └─ service/
(如 MessageService, OrderService)
                   ├─ MessageService.java
                    ├─ OrderService.java
SpringbootApplication.java # 启动类
          └─ resources/
                                # 配置文件等资
源(未展开)
                     # 前端 Vue 3 项目
 — vue/
  ├── package.json # 前端依赖配置
  ├── vite.config.js # Vite 配置文件
   ├─ .env.development # 开发环境变量(如
VITE_API_BASE_URL)
   ├─ .env.production # 生产环境变量
   └── src/
                     # 静态资源(图片、图标等)
      — assets/
                     # 路由配置(Vue Router)
      ├─ router/
      ├─ utils/
                     # 工具类(如 axios 封装)
      ├─ views/
                   # 页面视图组件(如订单页面等)
      ├─ App.vue
                   # 根组件
      └─ main.js
                      # 入口文件
```

3.部署环境

工具

工具	版本推荐
Node.js	>= 16.x
npm / pnpm	>= 8.x

工具	版本推荐
Java JDK	>= 17
Maven	>= 3.6
MySQL (可选)	>= 5.7

后端

```
server:
 port: 9090
# 数据库配置
spring:
  datasource:
    driver-class-name: com.mysql.cj.jdbc.Driver
   username: root
    password: 123
   url: jdbc:mysql://localhost:3306/vendingmachine?
useUnicode=true&characterEncoding=utf-
8&allowMultiQueries=true&useSSL=false&serverTimezone=
GMT%2b8&allowPublicKeyRetrieval=true
 mvc:
    pathmatch:
     matching-strategy: ant_path_matcher
# 优雅的操作数据库而不用写SQL
 #使用步骤
 #定义实体类(使用注解映射到表)
 #定义接口(继承 JpaRepository)
  jpa:
    hibernate:
     ddl-auto: update
    show-sql: true
```

```
mvn clean install
mvn spring-boot:run
```

页面

```
http://localhost:9090/
```

前端

依赖安装

```
npm install
```

启动

```
npm run dev
```

页面

```
http://localhost:5173/
```