系统设计报告

20301024-王圳

[年]

## 目录

[1需求分析 2](#_Toc138015177)

[2技术选择 4](#_Toc138015178)

[3系统设计 4](#_Toc138015179)

[4实现过程 6](#_Toc138015180)

## 1需求分析

本任务要求实现出货 REST API，主要功能包括航运和运输服务。需根据 RESTful API 设计规范来进行接口设计。

#### 1.1系统背景和简介

随着电商业务的快速发展，越来越多的人们开始在互联网上购物，物流服务也变得越来越重要。为了提高物流效率、降低物流成本、提升客户满意度，需要开发一款基于云计算平台的物流管理系统。

#### 1.2系统功能需求

用户管理：实现用户的注册、登录、信息修改等功能，保证用户的身份和权限安全。

订单管理：管理订单的创建、编辑、查询、删除等操作，确保订单信息的及时、准确、完整与安全。

运输管理：对运输车辆、货物、司机等信息进行管理，确保运输的正常、安全、高效和准确。

路线规划：通过路线规划算法，根据货物发货点和收货点地址信息自动规划运输路线，优化运输效率。

仓库管理：管理货物入库、出库、存储、查找等操作，确保货物的存储安全性和可追踪性。

费用管理：管理物流服务费用、运输费用、仓储费用等，确保费用按照合同执行，降低成本。

报表管理：提供运输记录、仓库使用情况、费用情况、客户满意度等报表分析，方便管理者掌握物流服务的全面情况。

客户服务：通过在线客服、反馈与投诉等方式，及时回应客户需求并提供维护服务。

#### 1.3系统非功能需求

安全性：保证用户信息、订单信息、货物信息等数据的安全性和隐私性，采用合适的加密算法并遵循相关安全标准。

可扩展性：能够支持多平台、多终端、多语言、多样式等自定义配置和扩展，便于对不断变化的业务需求做出快速响应。

可靠性：在高并发、断电等异常情况下，系统能够保证运行稳定，恢复效率较快。

用户友好性：系统操作简单、直观、易学，便于用户使用和培训。

性能要求：响应速度快，界面交互流畅，系统能够同时支持多个用户进行各种物流操作。

兼容性：系统能够兼容不同的浏览器、操作系统和设备，确保用户体验一致性和无缝性。

#### 1.4总结

在当前快速发展的电商行业中，物流服务是非常关键的一环。为了提高此类商业运营的运营效率和成本效益，需要一个基于云计算的物流管理系统。该系统主要包含用户、订单、运输、路线规划、仓库、费用、报表和客户服务八个模块。同时，该系统还需要具备安全性、可扩展性、可靠性、用户友好性、性能要求和兼容性等非功能需求，从而满足不断变化的业务需求。

## 2技术选择

为实现出货 REST API，我们采用了以下技术组件：

Spring Boot：用于快速构建可独立运行的 Spring 应用程序。

Spring MVC：基于 Spring 框架的 MVC框架，用于处理 Web 请求。

JPA/Batis：用于实现对 MySQL 数据库的访问，提供了方便的 CRUD 功能。

MySQL：关系型数据库，用于存储数据。

JWT：用于 API 认证和授权，提高系统安全性。

## 3系统设计

#### 3.1模块划分

该系统主要由用户管理、货物管理、订单管理、运输管理等多个功能模块组成。

（1）用户管理

用户管理模块用于实现用户的注册、登录、信息修改等功能，保证用户的身份和权限安全。此模块需要实现用户信息的录入、修改、查询、删除等功能，并对用户进行身份认证和权限控制。

（2）货物管理

货物管理模块用于管理货物的录入、查询、修改、删除等操作，确保货物信息的及时、准确、完整与安全。此模块需要实现对货物的各种属性的管理，包括重量、体积、描述、存储方式等。

（3）订单管理

订单管理模块用于管理客户订单的创建、编辑、查询、删除等操作，确保订单信息的及时、准确、完整与安全。此模块需要实现对订单状态的管理，包括待发货、已发货、待收货等状态。

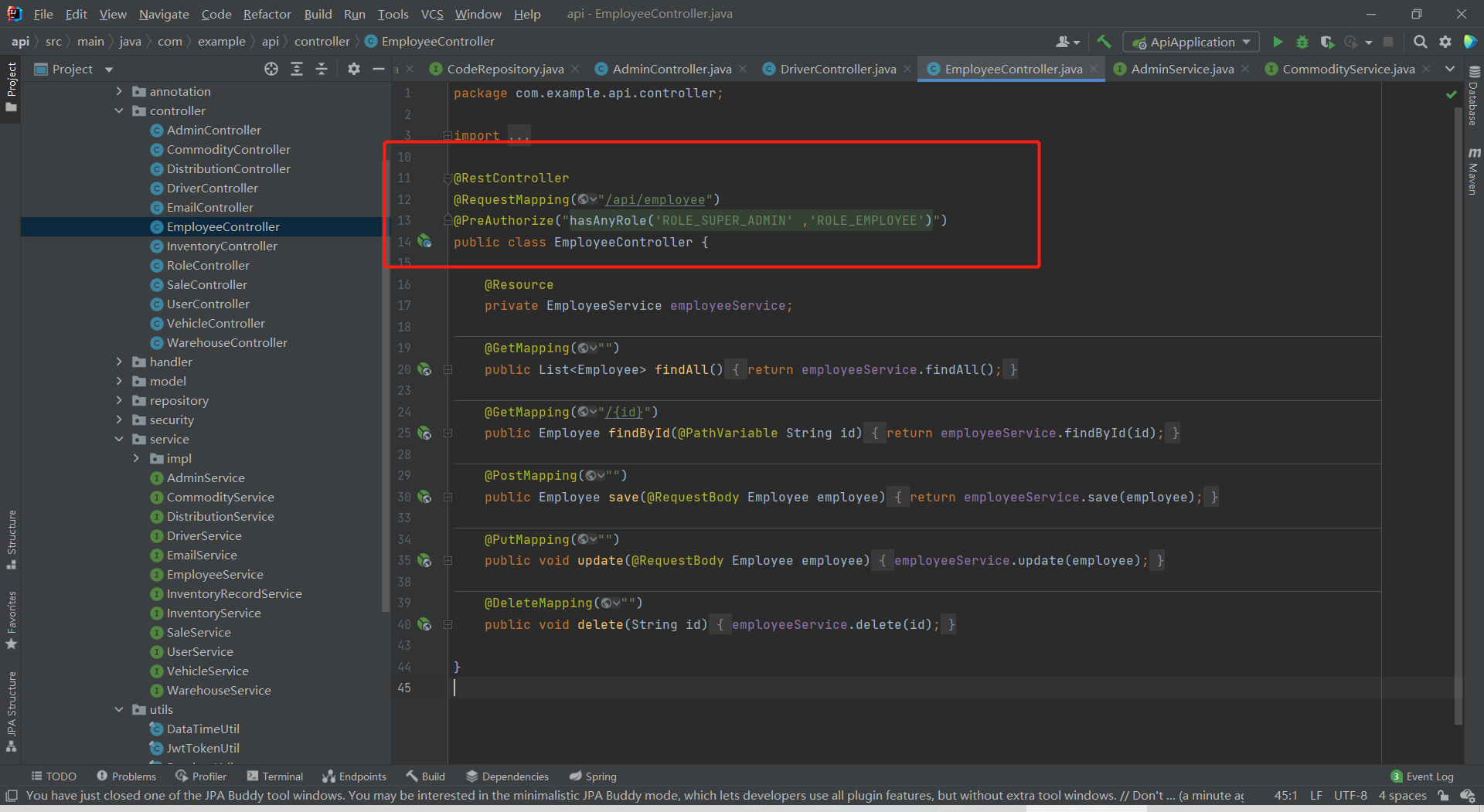
（4）运输管理

运输管理模块用于对运输车辆、货物、司机等信息进行管理，确保运输的正常、安全、高效和准确。此模块需要实现车辆调度、货物装卸、运输跟踪等功能。

#### 3.2RESTful API 接口设计

航运服务接口设计：包括创建一个新的航运记录、获取指定航运记录、获取全部航运记录等操作。

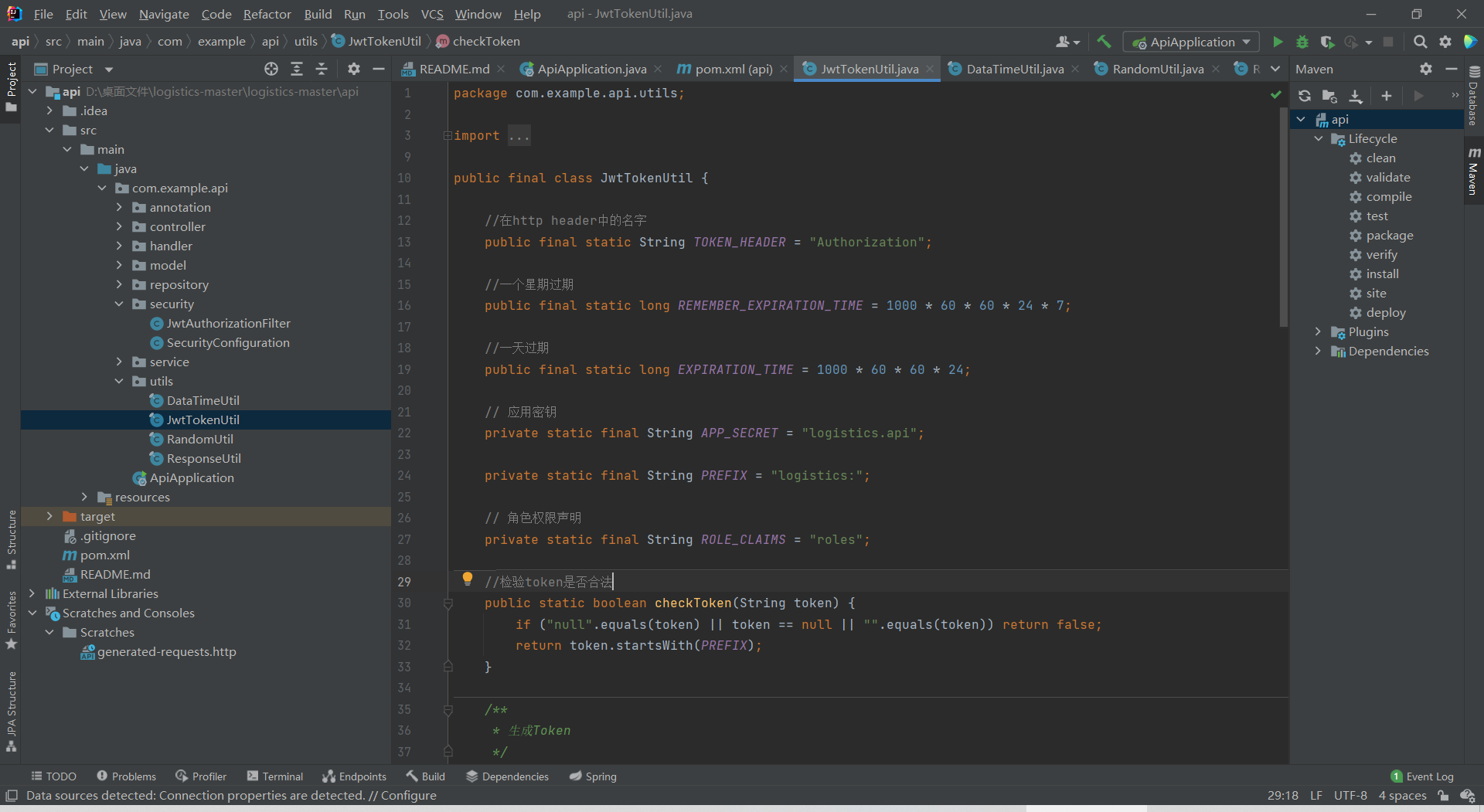
运输服务接口设计：包括创建一个新的运输记录、获取指定运输记录、获取全部运输记录等操作。



#### 3.3安全机制设计

使用 JWT 实现 API 认证和授权，通过 Access Token 控制用户权限。

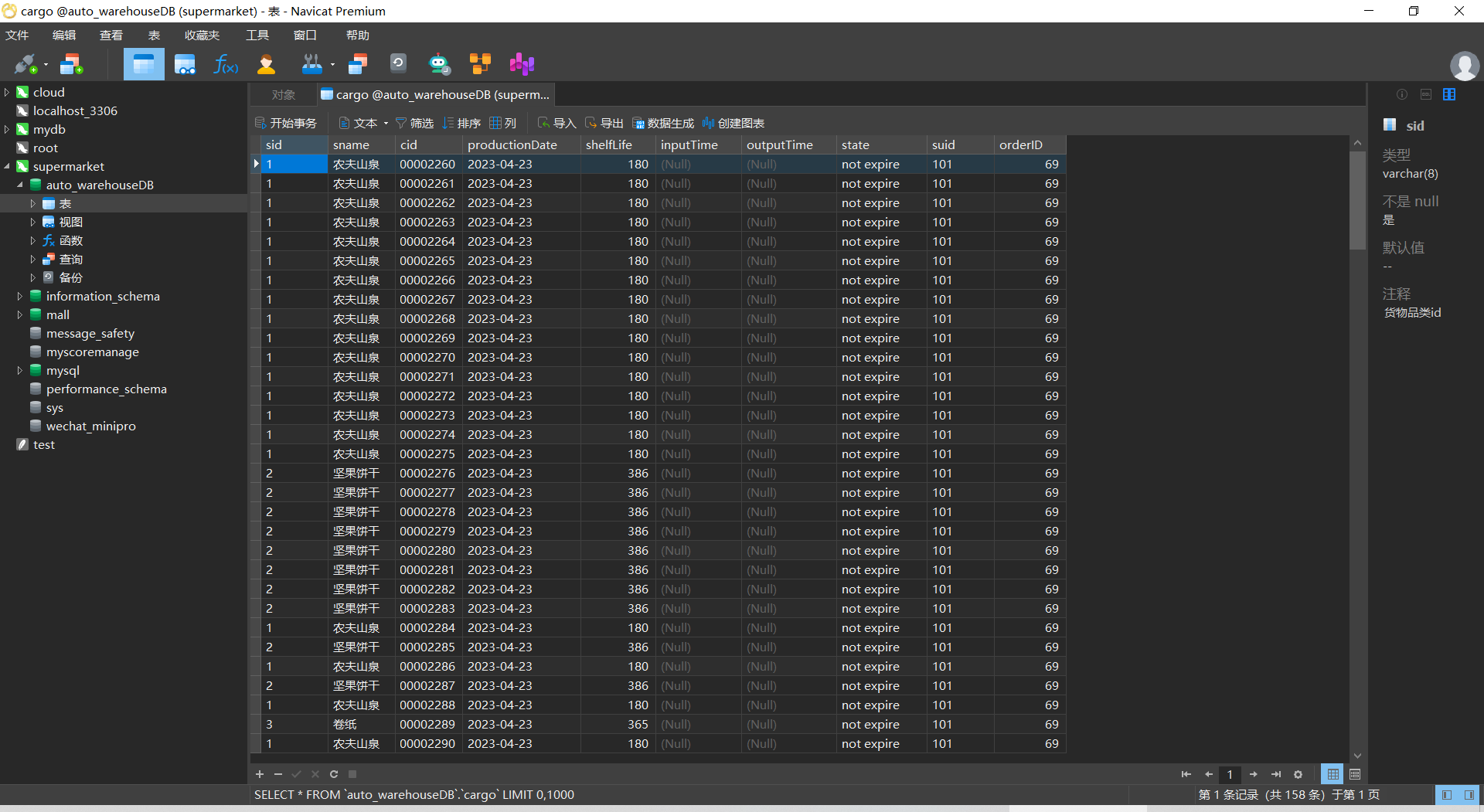
在用户请求时需要携带有效的 Access Token。可以使用 Interceptor 拦截器来验证 Access Token 的有效性。



#### 3.4数据库设计

设计两个数据表：航运记录和运输记录，包含对应的字段，如航班编号、航线、出发地点、目的地点、开始时间、结束时间等字段。

在 MySQL 中创建以上两个数据表并设置外键约束。



## 4实现过程

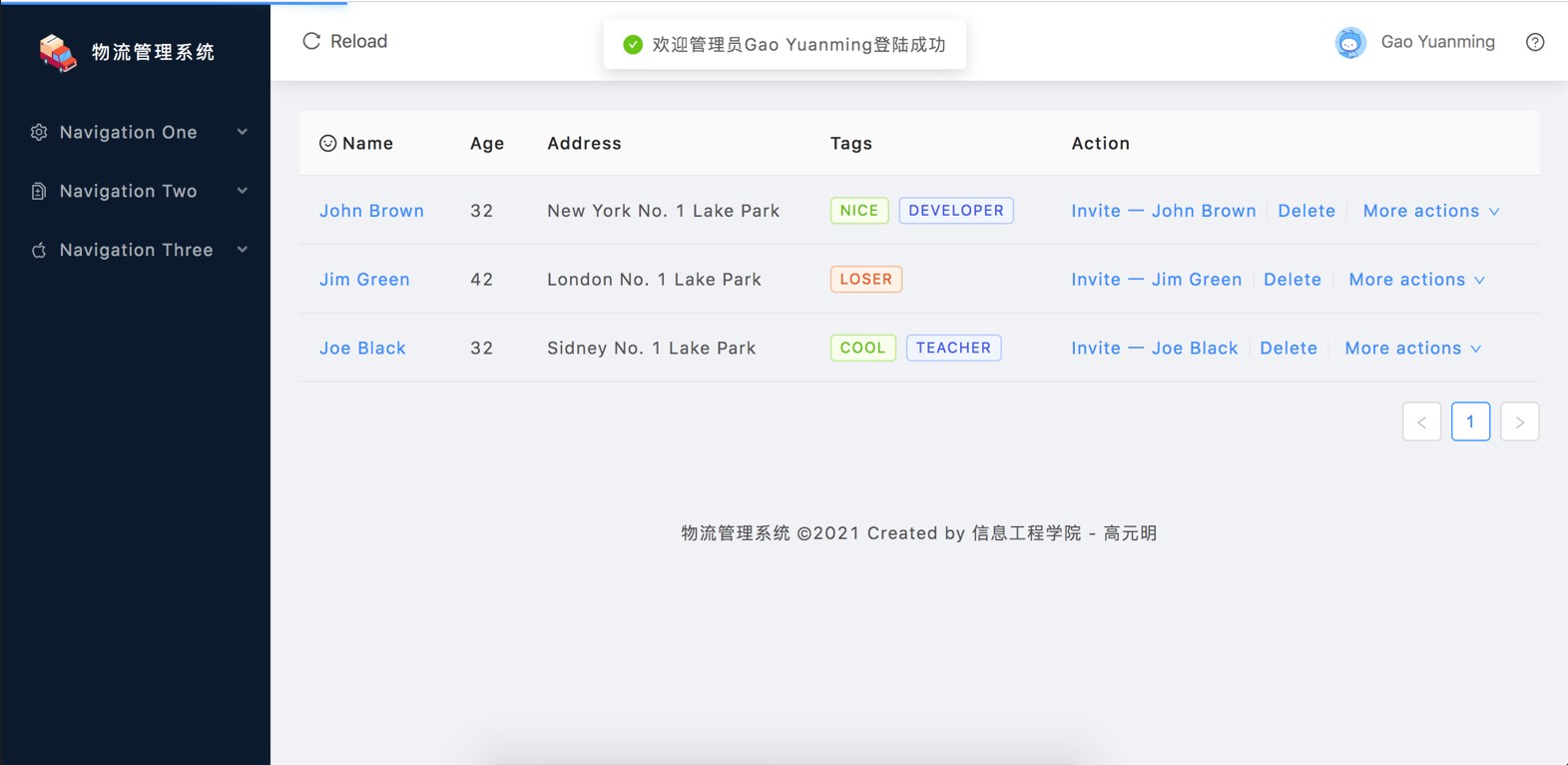
创建 Spring Boot 项目。

使用 Spring MVC 实现 RESTful API，分别编写航运服务和运输服务接口方法，并通过 Swagger 来开发和发布 API 文档。

使用 JPA/Batis 实现对 MySQL 数据库的访问，以实现数据的增删改查等操作。

使用 JWT 实现用户认证和授权，基于 Spring Security + JWT 技术栈实现用户登录控制和权限验证。

通过 Postman 等工具对 API 接口进行测试，保证其正常有效。

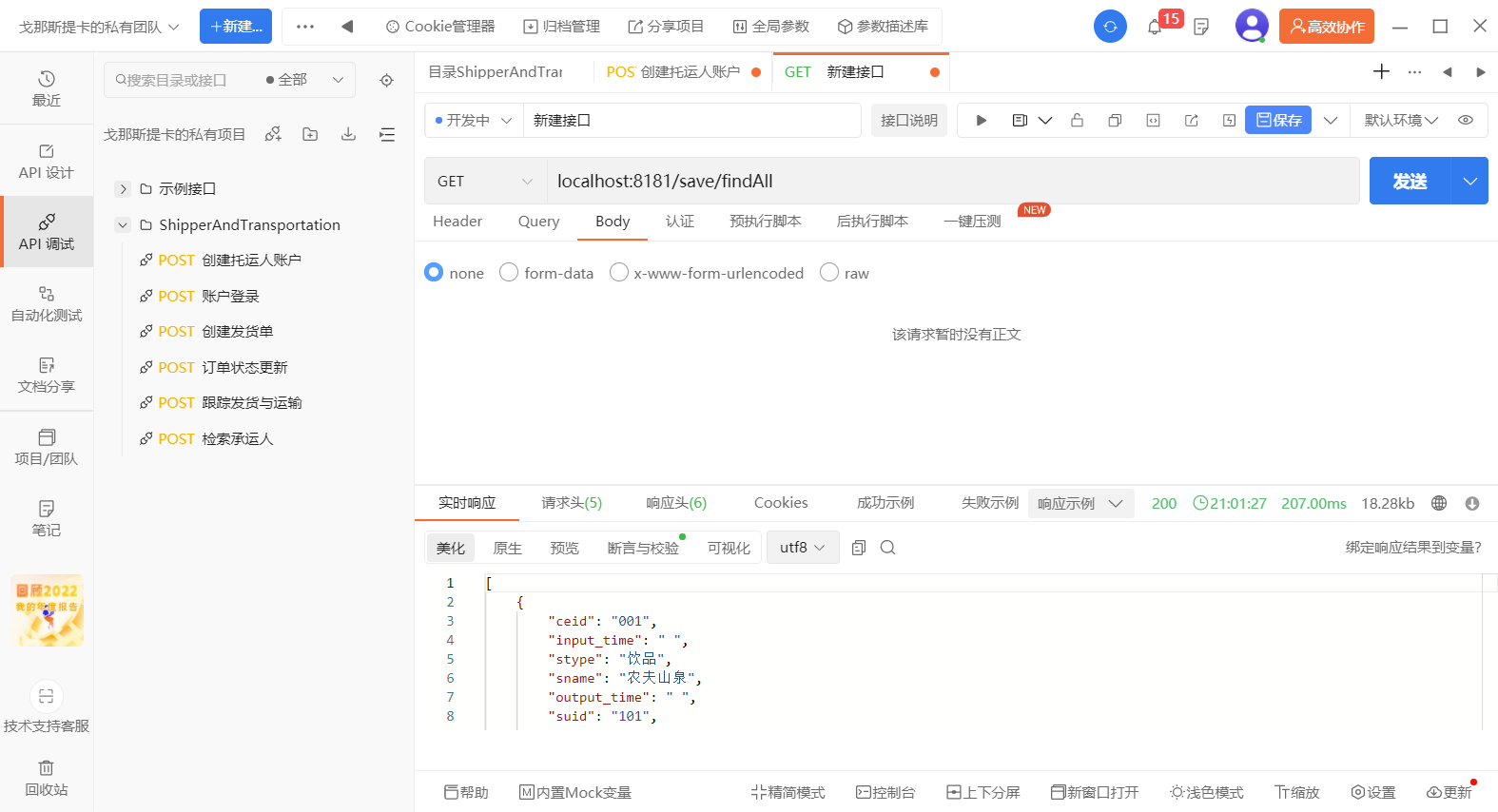


五、测试与部署

进行 API 接口功能测试和安全性测试，确保接口及其安全机制的有效性和可靠性。

配置服务器环境，将数据库、应用程序和 Web 服务器等组件进行配置和集成。

启动应用程序，测试程序在生产环境中的运行情况。



六、总结

本次任务中我们通过使用 Spring Boot、Spring MVC、JPA/Batis、MySQL 和 JWT 等技术组件来实现了 RESTful API 的设计和实现，保证了 API 接口的稳定性、扩展性和安全性。同时，我们采用了 Swagger 来开发和发布 API 文档，便于用户查询和使用。在测试与部署环节，我们进行了严格测试和部署过程，保证了应用程序在生产环境中的正常运行。