系统设计报告

# 1.系统需求分析

## 1.1 可行性分析

在该项目中，我们将采用Spring MVC + Spring Data JPA/Mybatis + Thymeleaf技术，以实现一个货运和运输网络应用程序。这些技术都是业内广泛使用的技术，并且都有强大的社区支持和文档资料。因此，我们可以保证项目的可行性。

## 1.2 项目要解决的问题

该项目将解决货运和物流过程中的一系列问题，例如货物管理、订单管理、位置追踪、交付安排和运输费用计算等。通过该应用程序，用户可以轻松地管理货运和物流过程，从而提高物流效率和降低成本。

## 1.3 项目目标

该项目的目标是实现一个高性能、可靠、安全的货运和运输网络应用程序。该应用程序将提供各种功能，例如货物管理、运输管理、位置追踪、交付安排和运输费用计算等。用户可以通过该应用程序轻松地管理货运和物流过程，从而提高物流效率和降低成本。

# 2.功能设计

## 2.1 功能划分

根据项目目标和要求，我们将该应用程序分为以下几个主要功能：

* 用户管理
* 订单管理
* 订单状态追踪
* 交付安排

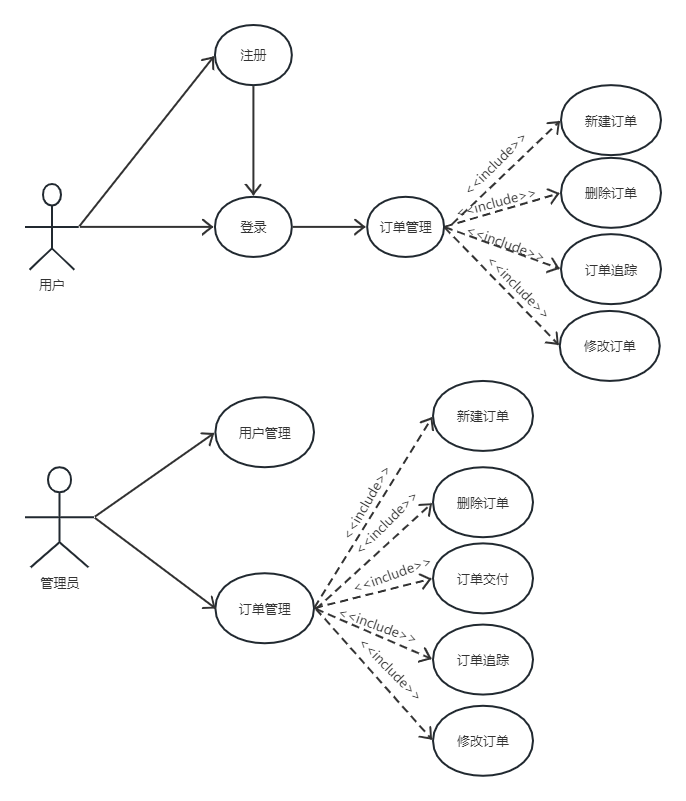
## 2.2 功能描述

* 用户管理：用户可以注册、登录和注销。已登录用户可以查看和编辑其个人信息。
* 订单管理：用户可以创建、修改和删除订单。用户可以查看其订单的状态。
* 订单状态追踪：用户可以查看其订单的当前状态和历史轨迹。
* 交付安排：用户可以安排货物的交付，并跟踪交付状态。

## 2.3 功能设计

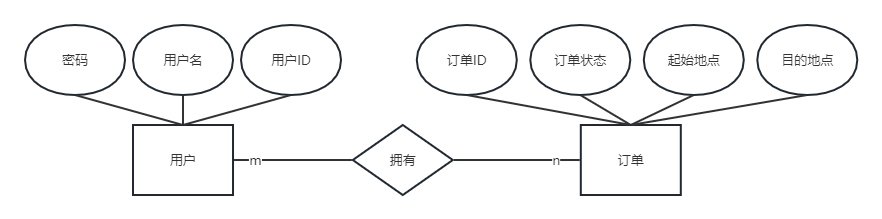
### 2.3.1 用例图

以下是该应用程序的用例图：



### 2.3.2 E-R图

以下是该应用程序的E-R图：



# 3.详细设计

## 3.1 后台设计

在后台设计中，我们将采用Spring MVC和Spring Data JPA/Mybatis技术。Spring MVC将用于处理HTTP请求和响应，而Spring Data JPA/Mybatis将用于管理数据访问。我们将使用标准的三层架构模式，将应用程序分为以下几层：

控制器层：用于处理HTTP请求和响应，负责调用服务层和渲染视图。

服务层：用于处理业务逻辑，负责调用数据访问层。

数据访问层：用于管理数据访问，包括与数据库交互和实现持久性存储。

## 3.2 前台设计

在前台设计中，我们将采用Thymeleaf技术来呈现动态内容。我们将使用Bootstrap框架来构建响应式UI，以便用户可以在各种设备上方便地访问应用程序。

# 4.数据操作及表格设计

## 4.1 数据操作

在数据操作方面，我们将采用Mybatis Plus技术，以实现对象关系映射（ORM）和持久性存储。我们将使用MySQL数据库来存储和管理数据。

## 4.2 表格设计

在数据库方面，我们将创建以下表格：

用户表：包括用户ID、用户名、密码、电子邮件和其他个人信息。

订单表：包括订单ID、起始地点、目的地点、状态等信息。

# 5.开发环境

## 5.1 开发工具

我们将使用以下工具和技术来开发该应用程序：

JDK 8+

Spring MVC

Mybatis Plus

Thymeleaf

Bootstrap

MySQL

## 5.2 结构：三层架构

我们将应用程序分为三个层次：控制器层、服务层和数据访问层。控制器层用于处理HTTP请求和响应，服务层用于处理业务逻辑，数据访问层用于管理数据访问和持久性存储。这种三层架构将应用程序的不同方面分离，提高了可维护性和可扩展性。