**Project Title:**"SpringBootShop" – A Modular E-Commerce Web Application Using Spring Boot  
  
**Problem Statement:**  
Most small to medium-sized businesses lack affordable and customizable e-commerce platforms tailored to their specific needs.

Many existing platforms are either too rigid or too complex, leading to challenges in integrating with third-party services, managing users, processing orders, and ensuring secure payments.

There is a need for a lightweight, modular, and scalable e-commerce application that can serve as a base product for businesses.

The platform should provide key functionalities like user authentication, product management, cart and order processing, secure checkout, and admin control, all built using Spring Boot for fast development and easy maintenance.  
  
**Objective:**  
Develop a full-stack e-commerce web application using Spring Boot as the backend framework. The system should handle the following core modules:

User Authentication & Authorization (Login/Signup with role-based access using Spring Security)

Product Management (CRUD for products with categories, prices, and stock)

Model:

* Id
* Name
* Stock
* Brand
* Category (Clothing, electronics, costmetics)
* Description
* Price
* Discount

Controller Methods / Service: Product

* getAllProducts()
* getSingleProduct()
* createProduct()
* getProductByName()
* deleteProduct()
* updateProductById()

Order Management (Checkout, order placement, order history)

Model:

* orderId,
* orderStatus
* orderHistory <List>

Controller /Service:

* CheckOrder() – (GET)
* updateOrder()

Admin Dashboard (Manage products, users, and view analytics)

Model User:

* id
* name
* address
* orderId
* email
* username
* password
* contactNumber

Controller /service:

* addUser()
* updateUser()
* deleteUser()
* getUserById()

Expected Features:  
Customer Portal:  
Browse products by category  
Search and filter functionality  
View order history

Admin Portal:  
Add/Edit/Delete products  
View customer orders ( GET )

Tech Requirements:  
Spring Boot (RESTful APIs, layered architecture)  
Spring Data JPA with Hibernate for ORM  
PostgreSQL as the relational database  
Spring Security with JWT for authentication  
  
Optional: Swagger for API documentation  
  
Use following:  
A working prototype of an e-commerce platform using Spring Boot.  
Modular codebase following best practices (Service, Controller, Repository layers).  
Secure user and admin roles.  
Easy-to-extend structure for additional features (reviews, coupons, shipping).