# Project Documentation for Sporty Shoes E-Commerce Portal

**Project Objective**

The aim of this project is to design and develop an e-commerce portal for Sporty Shoes, a sports shoe manufacturing and selling company. The project leverages Spring Boot, Spring JPA, and Spring Security frameworks to deliver a robust and secure e-commerce solution. The scope of the project includes planning the development, creating the source code, and uploading it to a GitHub repository.

**Sprint Information**

**Sprint 1: Requirements Gathering and Analysis**

In this phase, the primary task is to collect all the necessary requirements and analyze them. The result of this sprint will be a comprehensive requirements document.

**Sprint 2: Architecture and Design**

The next sprint focuses on finalizing the system architecture, using the Spring Boot framework for a scalable and maintainable microservice design. We will also outline the database design using Spring JPA for efficient data management.

**Sprint 3: Development of Core Features**

During this sprint, the main features of the application will be developed. These features include product listing, cart management, order management, and user management. Each feature will be implemented as a self-contained Spring Boot microservice.

**Sprint 4: Implementation of Admin Features**

This sprint involves the development of administrative features. This includes creation of an admin panel, management of users, management of products, order overview, and other admin controls. Spring Security will be used to ensure only authenticated administrators have access to these functionalities.

**Sprint 5: Security Implementation**

This sprint will focus on applying Spring Security to ensure secure user authentication and data encryption across the application. This includes implementing user login, registration, and session management functionalities, along with the prevention of common web vulnerabilities such as CSRF and XSS attacks.

**Sprint 6: Testing and Debugging**

In this final sprint, thorough testing and debugging will be performed to ensure that the application is bug-free and ready for deployment. Both unit testing (for individual components) and integration testing (for the entire system) will be conducted.

**Java Concepts and Technologies Used**

**Spring Boot**: This is a framework that simplifies the bootstrapping and development of new Spring applications. It provides a default setup for building Spring-powered applications.

**Spring JPA (Java Persistence API)**: JPA is a specification for accessing, persisting, and managing data between Java objects and a relational database. Spring JPA is a part of the larger Spring Data family and provides easy data access while reducing boilerplate code.

**Spring Security**: This is a powerful and customizable authentication and access-control framework. It is the standard for securing Spring-based applications. It handles a range of authentication, authorization, and other security features for the application.

**Thymeleaf**: This server-side Java template engine is used for creating views in Spring Boot applications. It provides natural templating capabilities that can be viewed in browsers and work as static prototypes, reducing time in the development cycle.

**Maven**: It is a powerful project management tool that is based on POM (project object model). It is used for projects build, dependency and documentation.

**Generic Features of the Product**

* **Product Management**: Admin can add, update, delete, and view products. Customers can view products and their details.
* **User Management**: Admin can view, add, update, and delete users. Users can register, login, and update their profiles.
* **Cart Management**: Users can add products to their cart, update quantities, and checkout.
* **Order Management**: Users can view their past orders. Admin can view all orders.
* **Admin Management**: An admin login is required to access the admin dashboard where all admin operations can be performed.

All these features will be implemented ensuring the principles of clean code and object-oriented programming to provide a robust and efficient e-commerce platform.