E-mail: [papunkumardas0@gmail.com](mailto:papunkumardas0@gmail.com)

Contact: +917008543889

**SportyShoes.com**

In this project, I developed Restful Spring Boot E-Commerce portal named SportyShoes.com that uses internal memory database i.e., Spring Data Base to store all the information about the products. users who purchased it & this information can be fetched up by date & Customer name as well.

Here, Admin can manage all the products by categorizing them as brand, color, size & price.

Admin can see the purchase list filtered by date & category.

Admin can Search user by their name and see the details by userID.

GitHub Link:[***https://github.com/PapunKumarDas/SportyShoes.com.git***](https://github.com/PapunKumarDas/SportyShoes.com.git)

As I have developed the RESTful API, it can be run on postman tool.

1. I have used SWAGGER in my spring boot RESTful API project so that the project documentation can be well understood and can easily be verified.
2. URL for the SWAGGER documentation:

[**https://localhost:8080/swagger-ui.html**](https://localhost:8080/swagger-ui.html)

**Headers End-Point**

1. **Get/Post/Put to get, add & update all Shoes---------------- /shoes**
2. **Get Shoes by Category:**
3. **Get shoes by brand ----------------------- /shoes/brand/{brand}**
4. **Get shoes by color ------------------------ /shoes/color/{color}**
5. **Get shoes by size -------------------------- /shoes/size/{size}**
6. **Get shoes by price ------------------------ /shoes/price/{price}**
7. **Get/Delete shoes by Id---------------------------- /shoes/{id}**
8. **Post/Put to add & update user---------------------- /user**
9. **Get to get all users info. --------------------------- /users**
10. **Get Users by:**
11. **Get users by userId ------------------------- /user/{userId}**
12. **Get users by username-------------------- /user/username/{userName}**
13. **Get users by userAge ---------------------- /user/userAge/{userAge}**
14. **Get/Delete users by id ------------------------------ /user/{userId}**

**Screen Shots of the output result:**

1. **Database Storing all the different versions of products**