**Project Name: tracking-api**

**Root folder:** /tracking-api

**Tools and Technologies**

* Java 17
* Spring Framework
* Spring Boot
* Open API
* Maven
* JUnit

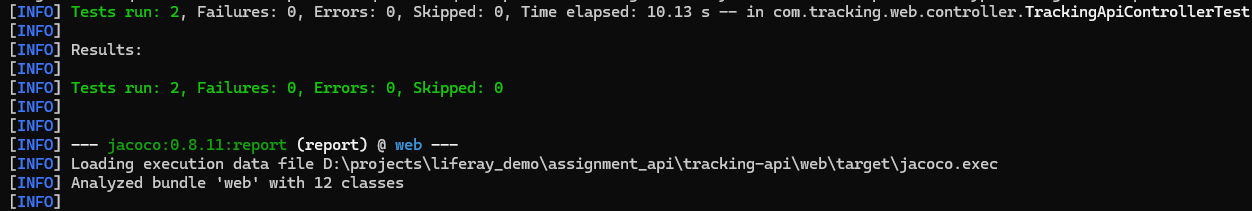
**Modules**

* **Application**
  + A configurations and command module to run the spring boot applications
* **Service**
  + Contains the services/dao for back-end logic, for example to generate tracking number
* **Web**
  + The api controller, api request lands here and internally it will call the services from Service module

**Unit Testing**

* Developed some test cases using Spring framework + Junit+ WebMvcTest + Mockito
* Run following command from root folder to run the test cases

“mvn clean install”



**Code Formatting**

* Integrated “spotless” for the code formatting
* Run following command to format the code from the root folder of the application

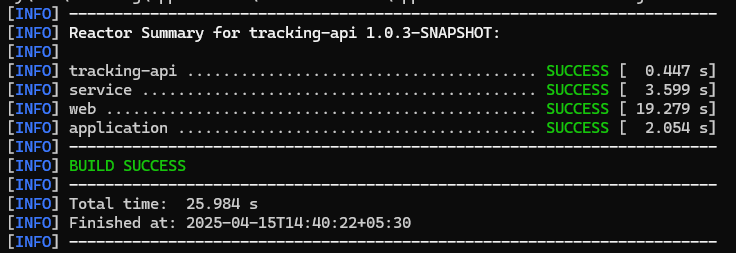
“mvnw spotless:apply”

**Build & Run**

* Run following command to build and run the application
* Build command from root folder

“mvn clean install”

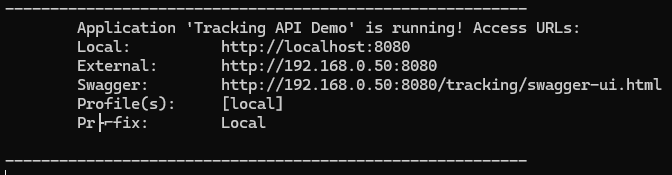
* Successful build



* Run command from root folder

“mvnw spring-boot:run -Dspring-boot.run.profiles=local -pl application”

* Successful run and start application



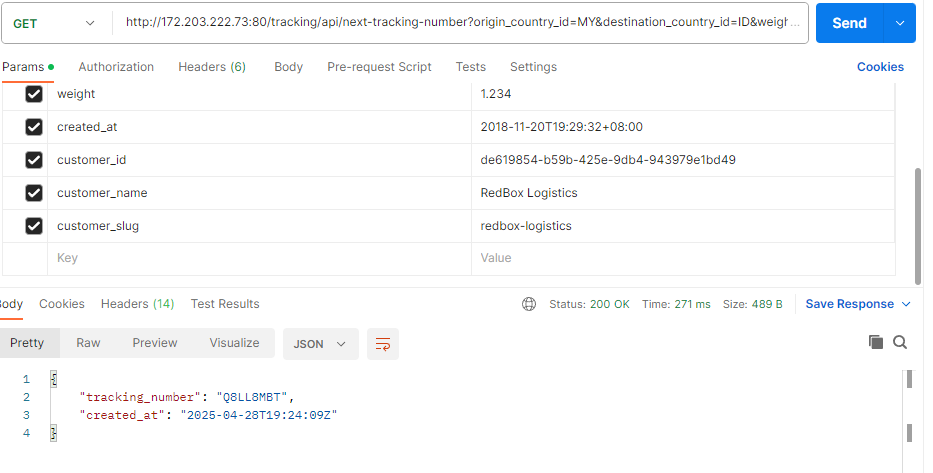
* Change the profiles (local, uat, prod) based on the environment)

**README File**

* Available in root folder of the project

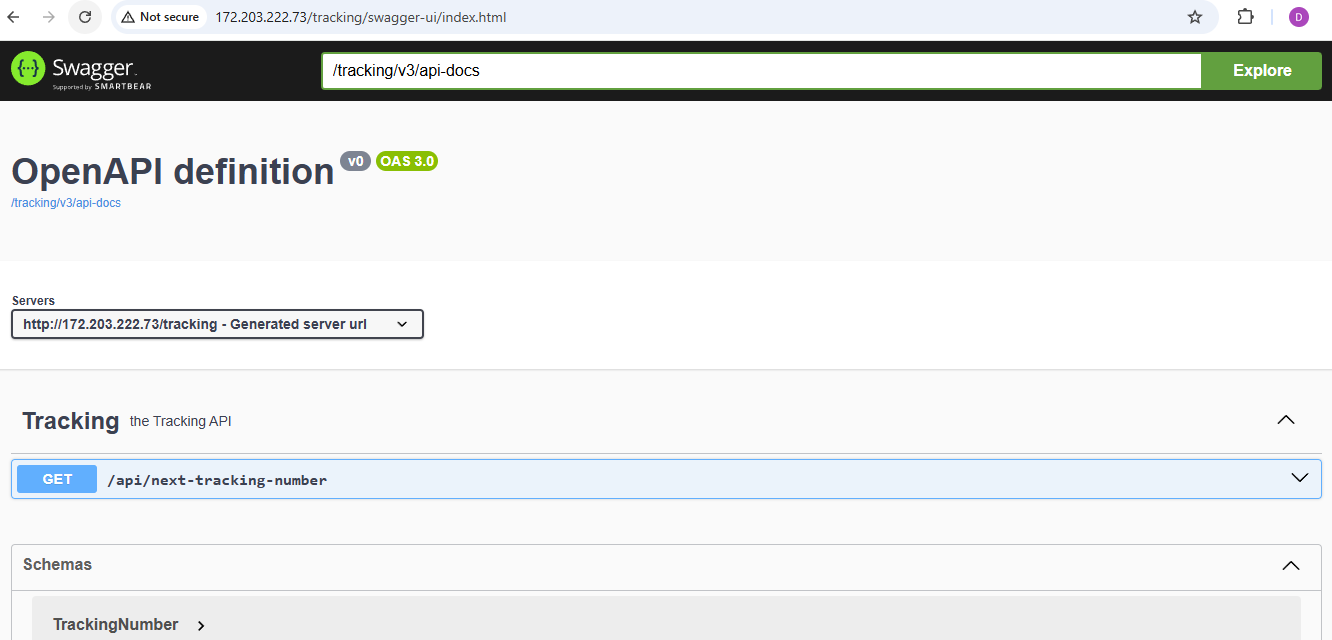
**Call API**

* URL: http://172.203.222.73:80/tracking/api/next-tracking-number?origin\_country\_id=MY&destination\_country\_id=ID&weight=1.234&created\_at=2018-11-20T19:29:32+08:00&customer\_id=de619854-b59b-425e-9db4-943979e1bd49&customer\_name=RedBox Logistics&customer\_slug=redbox-logistics
* Authentication: No auth
* Header: Content-Type application/json



**Swagger UI**

* URL: http://172.203.222.73/tracking/swagger-ui/index.html



**PLEASE CHANGE THE HOST AND POST BASED ON THE ENVIRONMENT**

**Deployment on free VM (AWS/Google Cloud Apps)**

* **Cloud: AZURE**
* **API Host: 172.203.222.73**
* **API: Port: 80**
* **API Protocol: http**

**Logic to generate 16 digit unique tracking number**

-- originCountryId + current day + destinationCountryId + last two digit of host ip address

+ last four digit of customerId + last four digit of current ms time System.currentTimeMillis() )

-- convert generated tracking number string to upper case

-- Regex : "^[A-Z0-9]{16}$"

**API Mandatory Query Params**

-- originCountryId

-- destinationCountryId

-- customerId

**Some Suggestions**

* This should be post api instead of GET as some important information (like customer\_id) supplied in query param
* The given regex to generate tracking number “^[A-Z0-9]{1,16}$” will generate the tracking number between1 to 16 (for example, “ERFR0W3M47Z1FJ9W”. “AR1”, “A”), so the regex should be “^[A-Z0-9]{16}$” which every time generates 16-character unique string as tracking number
* Created\_At should be system generated instead of passing in query param