**Microservices Assignment**

**E-Commerce Application Backend**

**Microservices:-**

* **InventoryAPI:-**
  + User can see all the products list
  + User can see the product detail of specific product (by product id)
  + Authentication is not required for these two operations.
* **AdminAPI:-**
  + Admin can Add or Delete products from the inventory.
  + Admin can Delete products from the inventory.
  + These routes are authorized by JWT. Only Admin can perform these operations.
* **AuthAPI:-**
  + Anyone can login by entering username and password.
  + If the credentials are correct, then user will get a secret token which is valid for 1 day(it can be changed on the basis of requirement)
  + Role based authorization is implemented. User and Admin.
* **CartAPI:-**
  + LoggedIn user can add a product to cart by selecting the quantity.
  + If product is already present in the cart then the quantity would be added to that product only.
  + LoggedIn user can do checkout. On checking out a product this solution assumes that the payment is done and directly places the order.
  + User would see the success message on checkout.
* **APIGateway:-**
  + Ocelot is used for routing(upstream and downstream paths are defined)
  + Eureka is used as service discovery.
  + Round robin technique is used for load balancing.
  + All the instances are being fetched by Eureka(there is no hardcoded data)

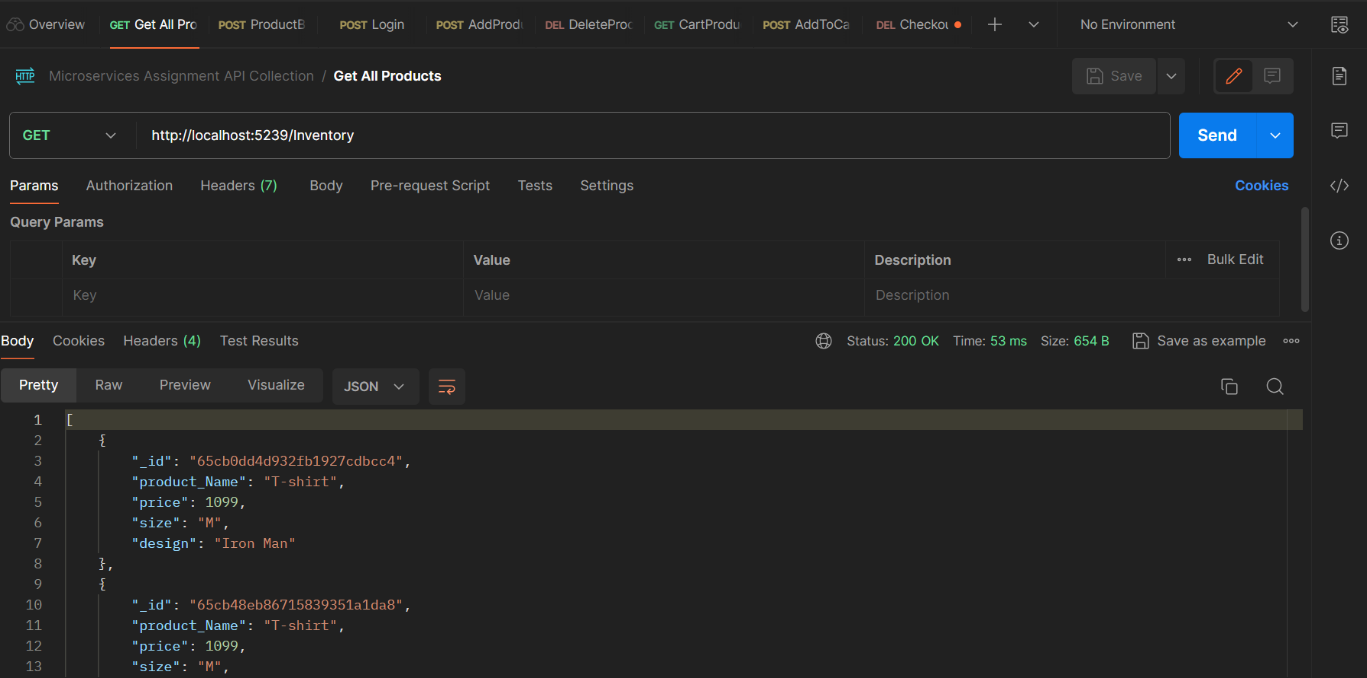
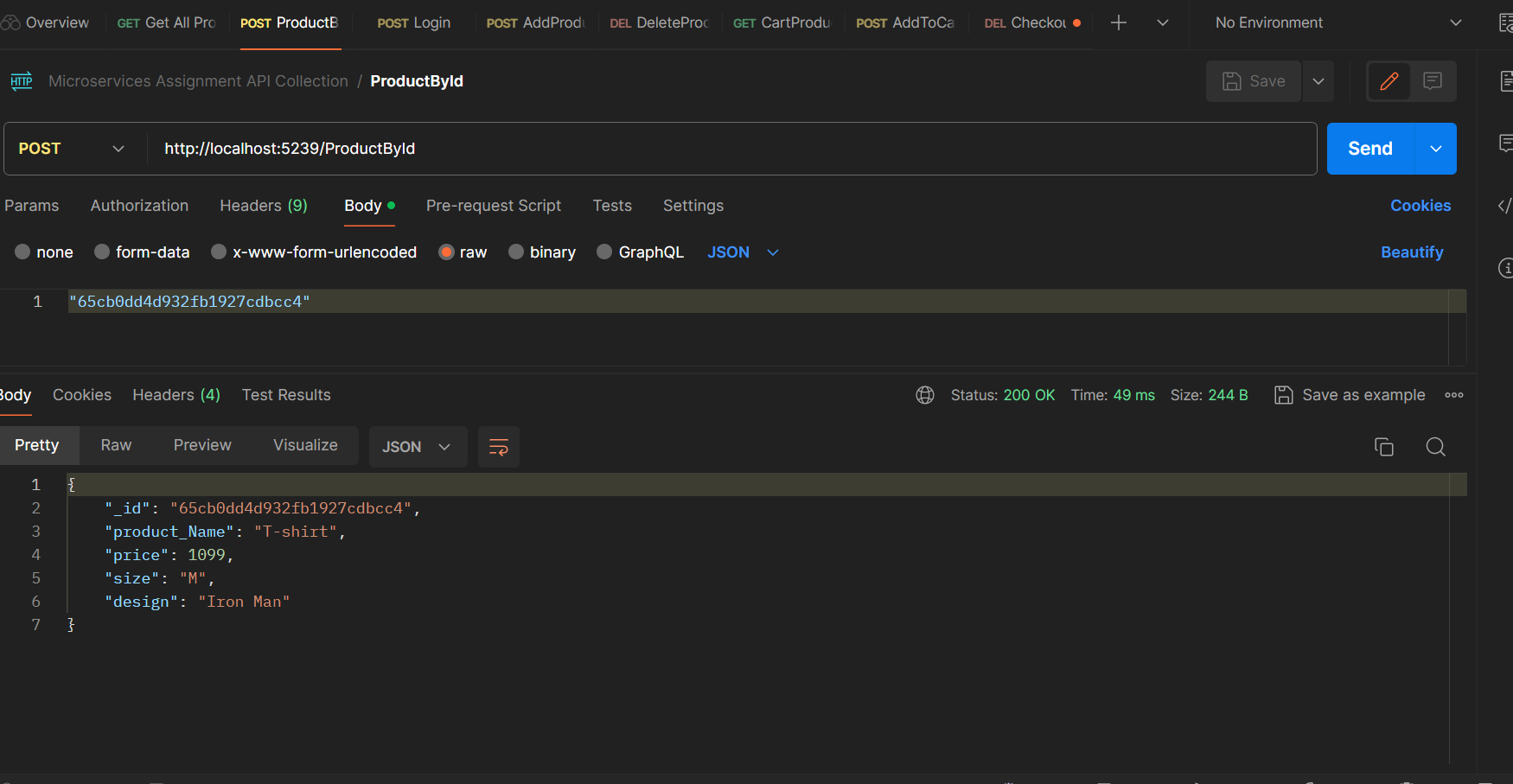
**Docker Images published on DockerHub:-**

* **InventoryAPI:-** <https://hub.docker.com/r/tusharawasthi268/inventoryapi>
* **AdminAPI:**- <https://hub.docker.com/r/tusharawasthi268/adminapi>
* **AuthAPI:**- <https://hub.docker.com/r/tusharawasthi268/authapi>
* **CartAPI:**- <https://hub.docker.com/r/tusharawasthi268/cartapi>
* **APIGateway:-** <https://hub.docker.com/r/tusharawasthi268/apigateway>
* **Eureka server:**- <https://hub.docker.com/r/steeltoeoss/eureka-server>

**Setup instructions:-**

* Download the source code.
* Navigate to docker-compose.yml file**.**
* Run ‘docker-compose up --build’ command to run the project.
* This commande will build the images and run it.
* Responses can be seen on Postman.
* Attached images of all APIs and their explanations below.

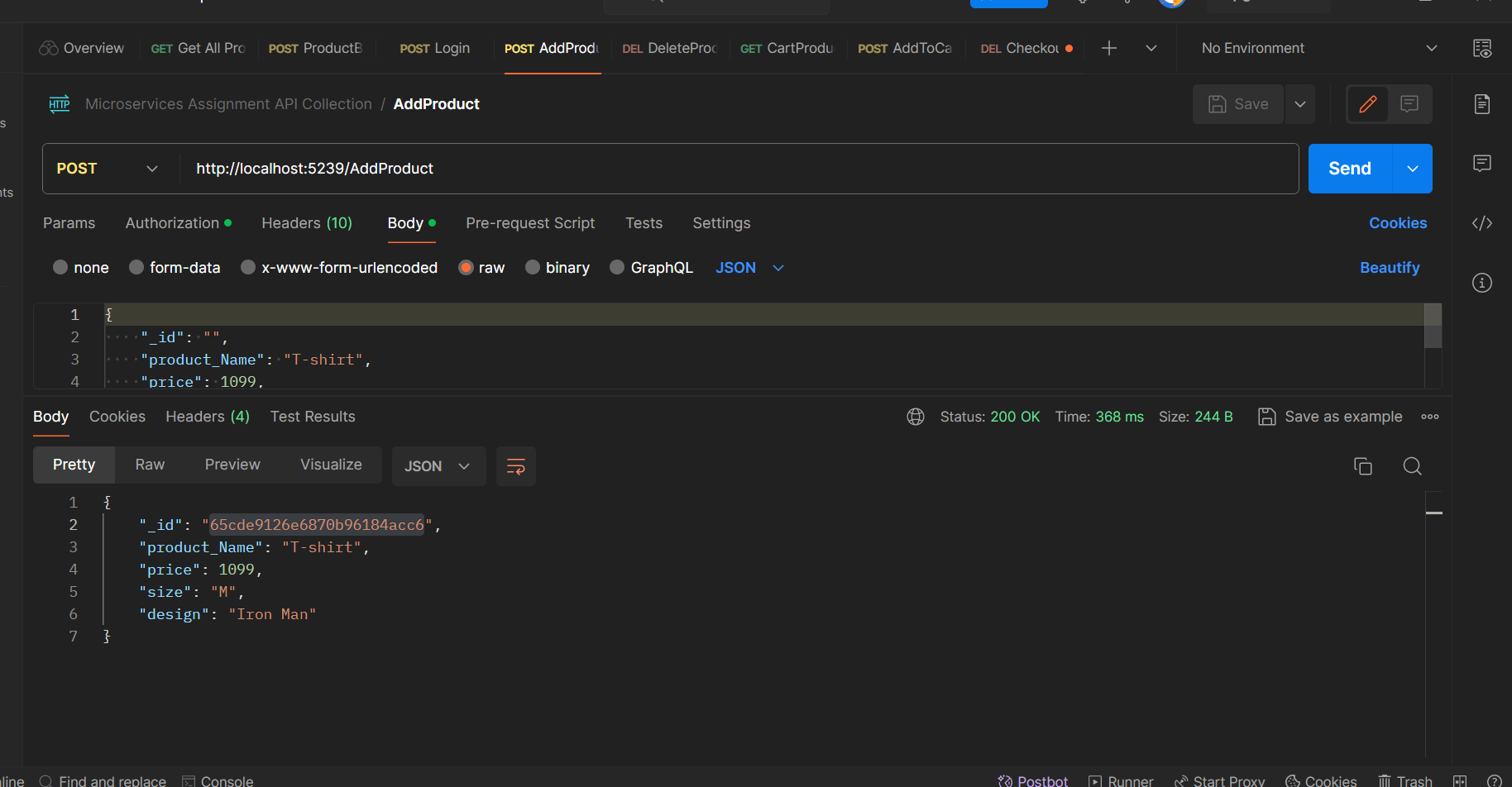
**Flow:-**

* **Fetch all Products:-**
* **Fetch Product Detail (Product by Id):-**
* **Login:-**

**A screenshot of a computer

Description automatically generated**

* **Add Products(Requires Admin access):-**

****

* A screenshot of a computer

  Description automatically generated**Delete product:-**
* **Fetch Products from User Cart:-**
* A screenshot of a computer

  Description automatically generatedA screenshot of a computer

  Description automatically generated**Add Products to cart:-**
* **Checkout:-**

**A screenshot of a computer

Description automatically generated**