**Ecommerce Microservices Backend**

* **Technologies Used:**
* **ASP.NET Core 6.0**
* **Eureka** for Service Discovery
* **Mass transit** using **RabbitMQ** for asynchronous event communication between microservices
* **Ocelot** for API Gateway
* **JWT Authentication**
* **Docker** for deployment
* **Identified Microservices Explanation:**

1. **Authentication Microservice:**

* Service that handles User login and registration. Upon login generates JWT Token
* By default two static users are added. Following credential can be used for login
* *Email: amangaur6515@gmail.com*

*Password: Aman@12345*

Its role is Admin

-  *Email:* [*test@gmail.com*](mailto:test@gmail.com)

*Password: Test@12345*Its role is Customer

1. **Product Microservice:**

* Service that handles CRUD upon Products. And has endpoints that Customers will use to browse the products list
* It displays product details using other microservice (Product Detail Microservice)

1. **Product Detail Microservice:**

* Service that handles product details

1. **Cart Microservice:**

* Service that handles add to cart and displays user cart items.

1. **API Gateway:**

* Api gateway is built using Ocelot.
* Running on <http://localhost:7085/>
* **DockerHub Login Credentials:**
* All microservices images are uploaded on DockerHub private Repository
* Email: [amangaur6515@gmail.com](mailto:amangaur6515@gmail.com)
* Password: Amangaur@2000
* Repo link: https://hub.docker.com/repository/docker/amangaur6515/ecommerce/general
* **How To Run:**
* First Login to docker using command:   
  >> *docker login*Enter above credentials
* Ensure that docker-compose.yml is at the directory level, and directory is open in docker terminal/wsl2
* Run the command:  
  >>*docker-compose up*
* **Postman Collection link:**

[*https://api.postman.com/collections/24864046-13518cbd-bea3-4be3-8445-220c0eb01de5?access\_key=PMAT-01HPA11D7R332544ZGTFSXSPZ3*](https://api.postman.com/collections/24864046-13518cbd-bea3-4be3-8445-220c0eb01de5?access_key=PMAT-01HPA11D7R332544ZGTFSXSPZ3)  
-collection json is also shared in the folder, use it to to run api endpoints