

## Assignment documentation

### 1. API Gateway - first service

- if client is valid then only allow otherwise 401
- authentication, authorization, routing
- auth token validation should be done here

### 2. Member - second service

DB → Postgres

#### a) POST → /member/register

request body

```
1 {
2   "email": "carttest@gmail.com",
3   "password": "Cart@123"
4 }
```

Positive case

```
1 {
2   "success": true,
3   "message": "Registration successful"
4 }
```

Error case

```
1 {
2   "success": false,
3   "message": "Invalid payload",
4   "errorCode": "INVALID_PAYLOAD"
5 }
6 {
7   "success": false,
8   "message": "Email already registered",
9   "errorCode": "EMAIL_ALREADY_EXISTS"
10 }
```

email

password - hashed password

username

register - Password hashed using Spring Security PasswordEncoder

generate one unique member id, which we can use for cart

#### b) login - Validate password using PasswordEncoder

POST → /member/login (email and password is required)

email

password

issuing token

- Hash password using `BCryptPasswordEncoder`
- Generate unique `memberId` (UUID or long sequence)

Request

```
1 {
2   "email": "carttest@gmail.com",
3   "password": "Cart@123"
```

```
4 }
```

#### Positive case

```
1 {
2   "success": true,
3   "message": "Login successful",
4   "data": {
5     "email": "carttest@gmail.com"
6   },
7   "accessToken":
"eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJjYXJ0dGZdEBnbWFpbC5jb20iLCJpc3MiOiJnYXRld2F5LXN1cnZpY2UiLCJpYXQiOiE3NjQ4OTcxMjIsImV4cCI6MTc2NDkwMDcyMi
widHlwZSI6ImFjY2VzcyJ9.YZr2J08dWIrIA83zohLf2cAz90HY2yJhaWLBzvUIrV9xQMciHJzAYFKy7-Jy626p6tn8EJj2qY0MKMS_CCUuk10y19IMA-Uz0u4i-
0Mv15DyqgReGvM9bjP6fWtc4F8xTL6k7b5LvD7QTpfXrXsZlGEjtHQ6wJP1w19rC4_K5Fw7BjTq5h2fUTY_waCUMq47y2QtMyfHz0JIEh1UA_oZnxf06XwSwU06UNEBO9QznH2T
-HnMhIEhL0I77CChw_XUF7pDeCfeAXkAZury-dIY1HMTX6R0F0NCvU7p_d2yMT0h0IKpHNNVn7ccNZBDy5LoWmkIrkhsG91NNLGz7JH1w",
8   "refreshToken":
"eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJjYXJ0dGZdEBnbWFpbC5jb20iLCJpc3MiOiJnYXRld2F5LXN1cnZpY2UiLCJpYXQiOiE3NjQ4OTcxMjIsImV4cCI6MTc2NTUwMTkyMi
wianRpijoizTI3MmEyMWItZWVhZ00TE4LTkwZ6UtdMDdkNGQ3Y2NhODIyIiwidHlwZSI6InJlZnJlc2giQ.1enACrz5HHapJdnhrCI3Hf1ngxLN7AdYH04BZQBDcF8PzcXkZX
AMA2emnjppfHqL6zCjKpSBZvoPHcoIwZVsp0bk53ecmhH01Kb8fRscjqoExy6GHUnb-
rE1kTtJC0CqVtdP8_Excmb1kZ0Mto9XwF2Eljka8EYqrQ42m3I2We0wvXFXd_X4MIJs_913rSZXBkJ2SP_uuB13CZif2qh0geDtC1L86xRhev2KAvbmRiLdhJCHhYIXfcTenkRoJ
Y87174LukLt0o-aPYW0ARRimAaNNkB6xEa27EAY9qB047bA-e7RW1x6f1j78exD9J3NRHrbD_6KFeHnLSge6wz-Ucw",
9   "tokenType": "Bearer",
10  "expiresIn": 3600
11 }
```

#### Error case

```
1 {
2   "success": false,
3   "message": "Invalid email or password",
4   "errorCode": "INVALID_CREDENTIALS"
5 }
```

### 3. Product - third service

DB → Mongo

need to use post with query param

a)

POST → /product/search?page=0&size=10&searchTerm=searchTermEnteredByUser

getproducts with pagination

Positive case

```
1 {
2   "content": [
3     {
4       "id": "693007779319b7099d1cf866",
5       "sku": "ME931QU2",
6       "name": "Rustic Iron Watch",
7       "description": "Laudantium itaque tempore fuga hic neque quo consectetur corrupti dignissimos repudiandae quas ratione
molestiae.",
8       "price": 22.85,
9       "category": "Electronics & Garden",
10      "tags": [
11        "Aluminum",
12        "Concrete",
13        "Cotton",
14        "Bronze"
15      ],
16      "images": [
17        "https://picsum.photos/seed/F53C6B45/600/600",
18        "https://picsum.photos/seed/A3797EFF/600/600",
19        "https://picsum.photos/seed/2A71DC7F/600/600"
20      ]
21    }
22  ],
23  "pageable": {
24    "pageNumber": 0,
25    "pageSize": 10
26  },
27  "totalElements": 10,
28  "totalPages": 1,
29  "last": true
30 }
```

error case

```

1 {
2   "errorCode": "400",
3   "errorMessage": "Product not found with the search term"
4 }
5
6 {
7   "errorCode": "INVALID_PAYLOAD",
8   "errorMessage": "Invalid payload"
9 }

```

b] GET →

/product/{sku}

get product details

sku, product name, product description, price

Positive case

```

1 {
2   "id": "693007779319b7099d1cf866",
3   "sku": "ME931QU2",
4   "name": "Rustic Iron Watch",
5   "description": "Laudantium itaque tempore fuga hic neque quo consectetur corrupti dignissimos repudiandae quas ratione molestiae.",
6   "price": 22.85,
7   "category": "Electronics & Garden",
8   "tags": [
9     "Aluminum",
10    "Concrete",
11    "Cotton",
12    "Bronze"
13  ],
14  "images": [
15    "https://picsum.photos/seed/F53C6B45/600/600",
16    "https://picsum.photos/seed/A3797EFF/600/600",
17    "https://picsum.photos/seed/2A71DC7F/600/600"
18  ]
19 }

```

error case

```

1 {
2   "errorCode": "400",
3   "errorMessage": "Product not found in the system"
4 }

```

c] POST → /createProduct

to create products

Request body

```

1 {
2   "category": "testcategory1",
3   "description": "testdescriptio1",
4   "images": [
5     "testimaage1"
6   ],
7   "name": "testname1",
8   "price": 12301,
9   "sku": "testsku1",
10  "tags": [
11    "testtag1"
12  ]
13 }

```

Positive case

```

1 {
2   "success": true,
3   "message": "Product created successfully",
4   "data": {
5     "id": "69314c9a3969c33ed0dadaab",
6     "sku": "testsku1",
7     "name": "testname1",
8     "description": "testdescriptio1",
9     "price": 12301.0,
10    "category": "testcategory1",

```

```

11     "tags": [
12         "testtag1"
13     ],
14     "images": [
15         "testimage1"
16     ]
17 }
18 }

```

error case

```

1 {
2     "errorCode": "400",
3     "errorMessage": "Product name is required"
4 }

```

#### 4. Cart - fourth service

DB → Mongo

a) Add product to cart - with product id and quantity passed in request body

POST → cart/addToCart

List cartItems → productId, productName, quantity, price

Request body

```

1 { "quantity": 1, "sku": "ME931QU2"}
2 member id passed in header

```

Positive case

```

1 {
2     "success": true,
3     "message": "Product added to cart successfully",
4     "data": {
5         "totalPrice": 114.25,
6         "totalQuantity": 5,
7         "items": [
8             {
9                 "id": "693007779319b7099d1cf866",
10                "name": "Rustic Iron Watch",
11                "sku": "ME931QU2",
12                "description": "Laudantium itaque tempore fuga hic neque quo consectetur corrupti dignissimos repudiandae quas ratione molestiae.",
13                "price": 22.85,
14                "category": "Electronics & Garden",
15                "tags": [
16                    "Aluminum",
17                    "Concrete",
18                    "Cotton",
19                    "Bronze"
20                ],
21                "images": [
22                    "https://picsum.photos/seed/F53C6B45/600/600",
23                    "https://picsum.photos/seed/A3797EFF/600/600",
24                    "https://picsum.photos/seed/2A71DC7F/600/600"
25                ],
26                "quantity": 5
27            }
28        ]
29    },
30    "timestamp": "2025-12-04T10:10:12.697246Z"
31 }

```

error case

```

1 {
2     "success": false,
3     "message": "An unexpected error occurred: Error fetching product details: 400 Bad Request",
4     "data": null,
5     "timestamp": "2025-12-05T01:32:10.030651Z"
6 }

```

## b] View cart - getting cart details

GET → cart/items

### Request body

```
1 member id passed in header
```

### Positive case

```
1 {
2   "success": true,
3   "message": "Cart retrieved successfully",
4   "data": {
5     "totalPrice": 22.85,
6     "totalQuantity": 1,
7     "items": [
8       {
9         "id": "693007779319b7099d1cf866",
10        "name": "Rustic Iron Watch",
11        "sku": "ME931QU2",
12        "description": "Laudantium itaque tempore fuga hic neque quo consectetur corrupti dignissimos repudiandae quas ratione
molestiae.",
13        "price": 22.85,
14        "category": "Electronics & Garden",
15        "tags": [
16          "Aluminum",
17          "Concrete",
18          "Cotton",
19          "Bronze"
20        ],
21        "images": [
22          "https://picsum.photos/seed/F53C6B45/600/600",
23          "https://picsum.photos/seed/A3797EFF/600/600",
24          "https://picsum.photos/seed/2A71DC7F/600/600"
25        ],
26        "quantity": 1
27      }
28    ]
29  },
30  "timestamp": "2025-12-04T12:10:01.249073Z"
31 }
```

### error case

```
1 {
2   "success": false,
3   "message": "Cart not found for user ID: 58080",
4   "data": null,
5   "timestamp": "2025-12-05T01:47:03.429491Z"
6 }
```

## c] Delete product from cart

DELETE → /deleteCartItem

### Request body

```
1 member id need to be passed in header
```

### Positive case

```
1 {
2   "success": true,
3   "message": "Cart items deleted successfully",
4   "data": {
5     "totalPrice": 0.0,
6     "totalQuantity": 0,
7     "items": []
8   },
9   "timestamp": "2025-12-05T01:48:49.488118Z"
10 }
```

### error case

```
1 {
2   "success": false,
3   "message": "Cart not found for user ID: 50908908",
4   "data": null,
```

```
5   "timestamp": "2025-12-05T01:49:01.690625Z"  
6 }
```

## Diagram

