

DiasporaEat E-Commerce Service Layer Design

Victor Oluwamakin

30th October 2024

1. Authentication Endpoints

POST/API/auth/register

Purpose: Allow customers to create new accounts

// request

```
{
  "Email": "user@example.com",
  "Password": "securePassword123",
  "firstName": "John",
  "lastName": "Doe",
  "phoneNumber": "+234123456789"
}
```

// Success Response - 201 Created

```
{
  "userId": "12345",
  "email": "user@example.com",
  "firstName": "John",
  "lastName": "Doe",
  "token": "jwt_token_here"
}
```

// Error Response - 400 Bad Request

```
{
  "error": "Email already exists"
}
```

ii. POST/api/auth/login

Purpose: Authenticate existing users

// Request

```
{
  "email": "user@example.com",
  "password": "securePassword123"
}
```

// Success Response - 200 OK

```
{
```

```
"userId": "12345",  
"token": "jwt_token_here"  
}
```

```
// Error Response - 401 Unauthorized  
{  
  "error": "Invalid credentials"  
}
```

2. Product Endpoints

GET/api/products

Purpose: Retrieve lists of available raw food products.

```
// Request Query Parameters  
?category=vegetables&page=1&limit=20&sort=price_asc
```

```
// Success Response - 200 OK  
{  
  "products": [  
    {  
      "id": "prod123",  
      "name": "Organic Cassava",  
      "description": "Fresh organic cassava from Nigerian farms",  
      "price": 15.99,  
      "category": "root_vegetables",  
      "stockQuantity": 100,  
      "images": ["url1", "url2"],  
      "origin": "Nigeria"  
    }  
  ],  
  "totalPages": 2,  
  "currentPage": 1  
}
```

ii. GET/api/products/{productId}

Purpose: Get detailed information about a specific product.

// Success Response - 200 OK

```
{
  "id": "prod123",
  "name": "Organic Cassava",
  "description": "Fresh organic cassava from Nigerian farms",
  "price": 15.99,
  "category": "root_vegetables",
  "stockQuantity": 100,
  "images": ["url1", "url2"],
  "origin": "Nigeria",
  "nutritionalInfo": {},
  "storageInstructions": "Store in a cool, dry place"
}
```

// Error Response - 404 Not Found

```
{
  "error": "Product not found"
}
```

3. Cart Endpoints

POST/api/cart

Purpose: Add items to the shopping cart

// Request

```
{
  "productId": "prod123",
  "quantity": 2
}
```

// Success Response - 200 OK

```
{
  "cartId": "cart123",
  "items": [
    {
```

```
    "productId": "prod123",
    "quantity": 2,
    "price": 15.99,
    "subtotal": 31.98
  }
],
"total": 31.98
}

// Error Response - 400 Bad Request
{
  "error": "Insufficient stock"
}
```

ii. GET/api/cart

Purpose: Retrieve current cart contents

```
// Success Response - 200 OK
{
  "cartId": "cart123",
  "items": [
    {
      "productId": "prod123",
      "name": "Organic Cassava",
      "quantity": 2,
      "price": 15.99,
      "subtotal": 31.98
    }
  ],
  "total": 31.98
}
```

iii. PUT/api/cart/{itemId}

Purpose: Update cart item quantity

```
// Request
{
  "quantity": 3
}
```

```
// Success Response - 200 OK
{
```

```
"cartId": "cart123",  
"items": [...],  
"total": 47.97  
}
```

iv. DELETE/api/cart/{itemId}

Purpose: Remove item from the cart

// Success Response - 204 No Content

4. Order Endpoints

POST/api/orders

Purpose: Create new order from cart

// Request

```
{  
  "shippingAddress": {  
    "street": "123 Main St",  
    "city": "Lagos",  
    "state": "Lagos State",  
    "country": "Nigeria",  
    "postalCode": "100001"  
  },  
  "paymentMethod": "card"  
}
```

// Success Response - 201 Created

```
{  
  "orderId": "order123",  
  "status": "pending",  
  "total": 47.97,  
  "paymentStatus": "pending",  
  "estimatedDelivery": "2024-11-05"  
}
```

ii. GET/api/orders

Purpose: Retrieve user's order history

// Success Response - 200 OK

```
{  
  "orders": [  
    {  
      "orderId": "order123",
```

```
    "date": "2024-11-01",
    "status": "processing",
    "total": 47.97
  }
]
```

GET/api/orders/{orderId}

Purpose: Get detailed order information

// Success Response - 200 OK

```
{
  "orderId": "order123",
  "status": "processing",
  "items": [...],
  "shippingAddress": {...},
  "total": 47.97,
  "paymentStatus": "paid",
  "trackingInfo": {...}
}
```

5. User Profile Endpoints

GET/api/profile

Purpose: Retrieve user profile information.

// Success Response - 200 OK

```
{
  "userId": "12345",
  "email": "user@example.com",
  "firstName": "John",
  "lastName": "Doe",
  "phoneNumber": "+234123456789",
  "addresses": [...]
}
```

ii. PUT/api/profile

Purpose: Update user profile information

// Request

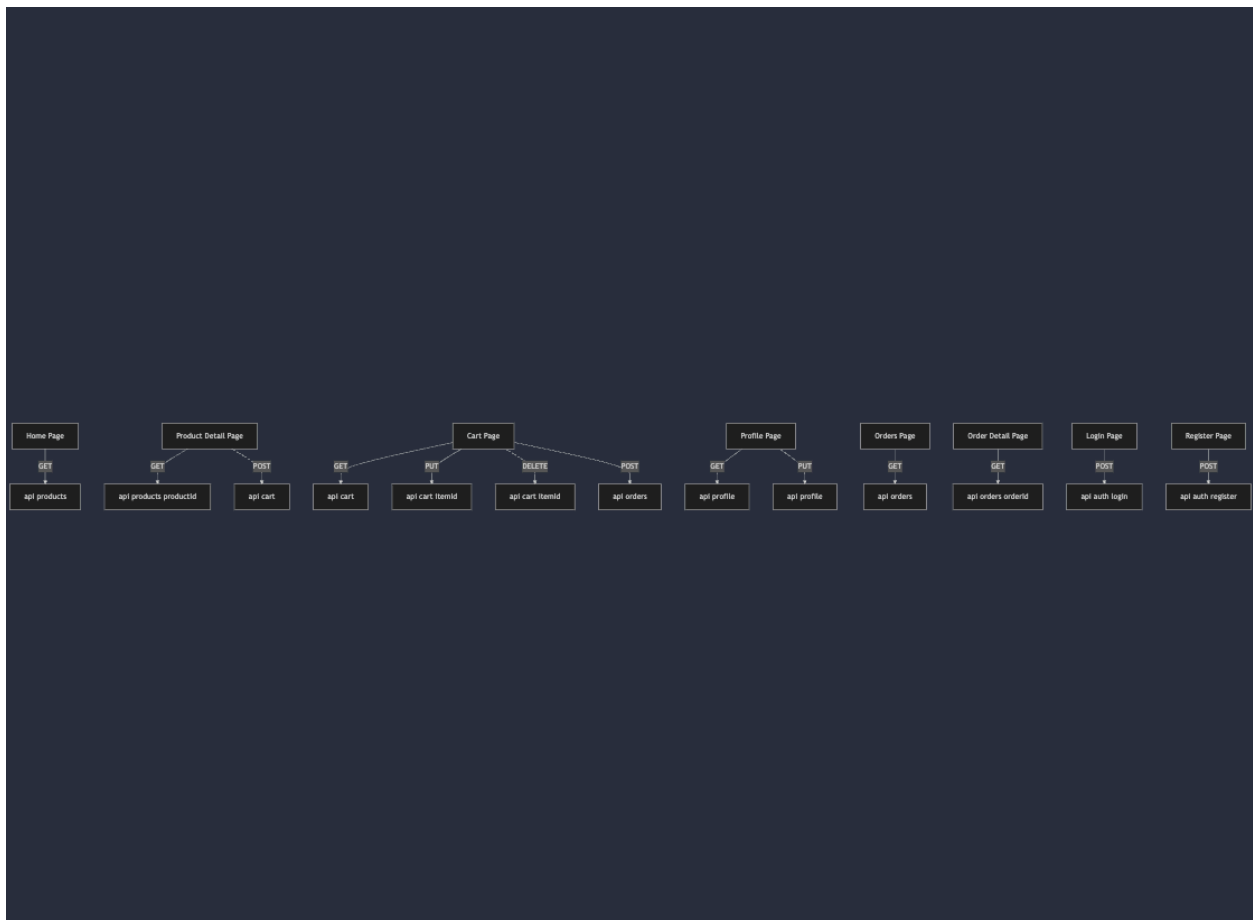
```
{
```

```
"firstName": "John",  
"lastName": "Doe",  
"phoneNumber": "+234123456789"  
}
```

// Success Response - 200 OK

```
{  
  "userId": "12345",  
  "email": "user@example.com",  
  "firstName": "John",  
  "lastName": "Doe",  
  "phoneNumber": "+234123456789"  
}
```

Page to Endpoint Mapping Diagram



Security Considerations

1. All endpoints except login/register and product listing require JWT authentication
2. Rate limiting applied to authentication endpoints
3. Input validation and sanitization for all requests
4. CORS policies to be implemented
5. SSL/TLS encryption is required for all API communication

Error Handling

All endpoints follow a consistent error response format:

```
{  
  
  "error": "Error message here",  
  
  "code": "ERROR_CODE",  
  
  "details": {} // Optional additional error details  
}
```

Standard HTTP status codes:

- 200: Success
- 201: Created
- 204: No Content
- 400: Bad Request
- 401: Unauthorized
- 403: Forbidden
- 404: Not Found
- 500: Internal Server Error