

E-Ticaret Projesi Raporu

Proje Adı: Ecommerce Backend API

1. Proje Tanıtımı

Bu projede Spring Boot kullanılarak bir e-ticaret sisteminin backend yapısı oluşturulmuştur. Sistemde müşteri kaydı, ürün yönetimi, sepet işlemleri, sipariş oluşturma ve stok yönetimi gibi temel e-ticaret işlemleri gerçekleştirilmiştir.

2. Kullanılan Teknolojiler

- Java 17
- Spring Boot
- Spring Data JPA
- H2 Database
- Postman
- IntelliJ IDEA
- Git & GitHub

3. Karakter Senaryosu

Bu senaryoda kullanıcı olarak Ayşe Yılmaz isimli bir müşteri sistemde işlem yapacaktır. Ürün olarak ise **kulaklık** ve **telefon** kullanılacaktır.

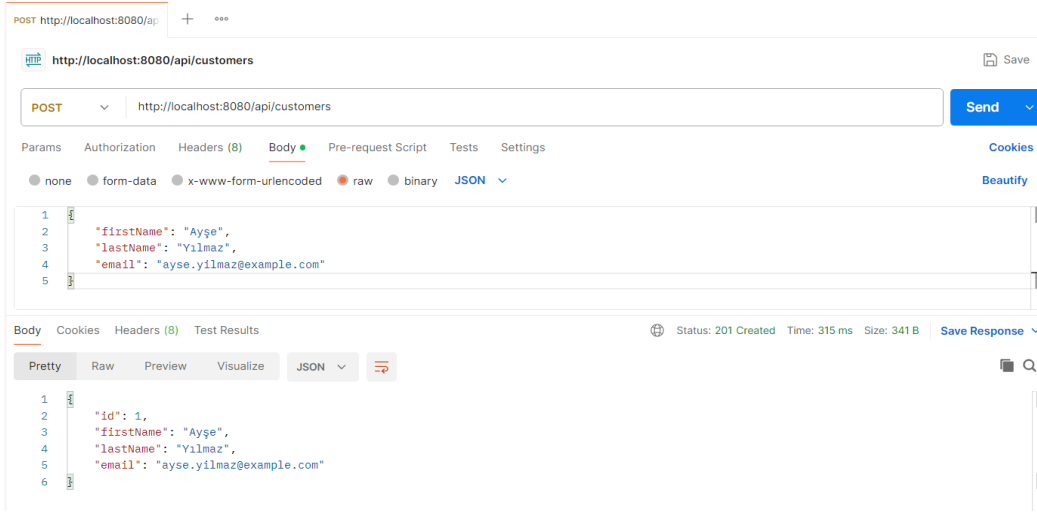
4. Müşteri Oluşturma

4.1. Müşteri tablosu (Customer) ilk hali:

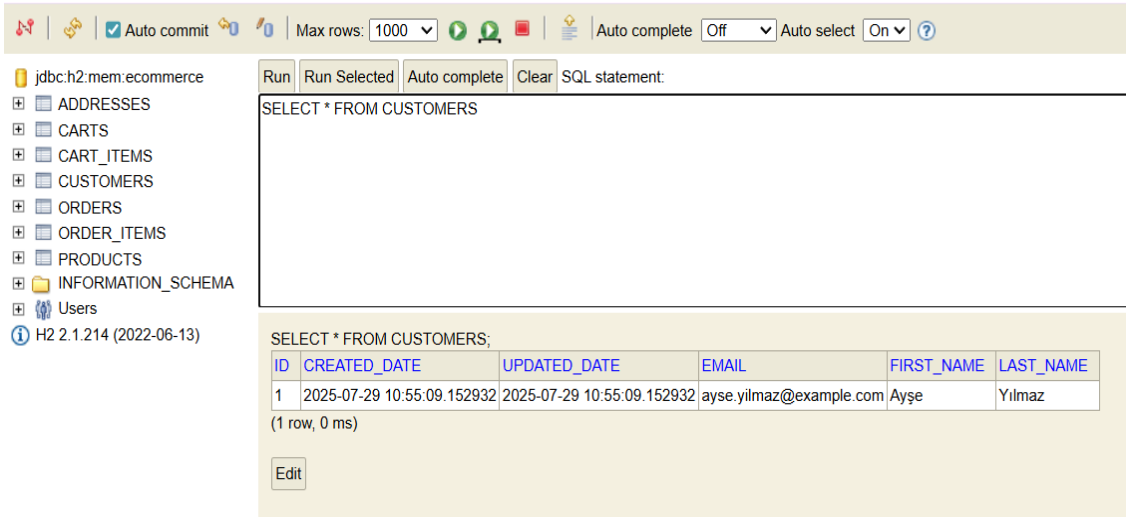
The screenshot shows a database client interface with a sidebar on the left listing database objects: jdbc:h2:mem:ecommerce, ADDRESSES, CARTS, CART_ITEMS, CUSTOMERS, ORDERS, ORDER_ITEMS, PRODUCTS, INFORMATION_SCHEMA, Users, and H2 2.1.214 (2022-06-13). The main area displays the SQL statement 'SELECT * FROM CUSTOMERS' and the result set. The result set is empty, showing '(no rows, 1 ms)'. The columns listed are ID, CREATED_DATE, UPDATED_DATE, EMAIL, FIRST_NAME, and LAST_NAME. There is an 'Edit' button at the bottom.

ID	CREATED_DATE	UPDATED_DATE	EMAIL	FIRST_NAME	LAST_NAME
----	--------------	--------------	-------	------------	-----------

4.2. Postman ile müşteri oluşturma:



4.3. Müşteri oluşturulduktan sonraki tablo:



5. Ürün Oluşturma

5.1. Ürün tablosunun ilk hali:

The screenshot shows a database client interface with a sidebar on the left listing database objects: jdbc:h2:mem:ecommerce, ADDRESSES, CARTS, CART_ITEMS, CUSTOMERS, ORDERS, ORDER_ITEMS, PRODUCTS, INFORMATION_SCHEMA, and Users. The main area displays the SQL statement: `SELECT * FROM PRODUCTS;`. Below the statement, the table structure is shown with columns: ID, CREATED_DATE, UPDATED_DATE, DESCRIPTION, IMAGE_URL, NAME, PRICE, and STOCK_QUANTITY. The status indicates "(no rows, 1 ms)".

5.2. Postman ile 2 ürün ekleme:

The first screenshot shows a POST request to `http://localhost:8080/api/products` with a JSON body: `{ "name": "Akıllı Telefon X", "description": "Yüksek performanslı son model telefon", "price": 12000.00, "stockQuantity": 5, "imageUrl": "http://example.com/telefon.jpg" }`. The response is a 201 Created status with a time of 30 ms and a size of 429 B. The response body is: `{ "id": 1, "name": "Akıllı Telefon X", "description": "Yüksek performanslı son model telefon", "price": 12000.00, "stockQuantity": 5, "imageUrl": "http://example.com/telefon.jpg" }`.

The second screenshot shows a second POST request to `http://localhost:8080/api/products` with a JSON body: `{ "name": "Kablosuz Kulaklık Y", "description": "Gürültü engellemeli yüksek ses kalitesi", "price": 2500.00, "stockQuantity": 10, "imageUrl": "http://example.com/kulaklik.jpg" }`. The response is a 201 Created status with a time of 6 ms and a size of 436 B. The response body is: `{ "id": 2, "name": "Kablosuz Kulaklık Y", "description": "Gürültü engellemeli yüksek ses kalitesi", "price": 2500.00, "stockQuantity": 10, "imageUrl": "http://example.com/kulaklik.jpg" }`.

5.3. Ürün tablosunun dolu hali:

jdbc:h2:mem:ecommerce

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM PRODUCTS

ID	CREATED_DATE	UPDATED_DATE	DESCRIPTION	IMAGE_URL	NAME	PRICE	STOCK_QUANTITY
1	2025-07-29 10:56:32.738467	2025-07-29 10:56:32.738467	Yüksek performanslı son model telefon	http://example.com/telefon.jpg	Akıllı Telefon X	12000.00	5
2	2025-07-29 10:58:08.152057	2025-07-29 10:58:08.152057	Gürültü engellemeli yüksek ses kalitesi	http://example.com/kulaklik.jpg	Kablosuz Kulaklık Y	2500.00	10

(2 rows, 0 ms)

Edit

6. Adres Tanımlama

6.1. Adres tablosu ilk hali:

jdbc:h2:mem:ecommerce

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM ADDRESSES

ID	CREATED_DATE	UPDATED_DATE	ADDRESS_TYPE	CITY	COUNTRY	STATE	STREET	ZIP_CODE	CUSTOMER_ID
----	--------------	--------------	--------------	------	---------	-------	--------	----------	-------------

(no rows, 1 ms)

Edit

6.2. Postman ile adres ekleme:

http://localhost:8080/api/customers/1/addresses

POST http://localhost:8080/api/customers/1/addresses

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

cookies Beautify

```
1 {
2   "street": "Gül Sokak No: 7",
3   "city": "İzmir",
4   "state": "İzmir",
5   "zipCode": "35000",
6   "country": "Türkiye",
7   "addressType": "SHIPPING"
8 }
```

Body Cookies Headers (8) Test Results

Status: 201 Created Time: 45 ms Size: 406 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "customerId": 1,
4   "street": "Gül Sokak No: 7",
5   "city": "İzmir",
6   "state": "İzmir",
7   "zipCode": "35000",
8   "country": "Türkiye",
9   "addressType": "SHIPPING"
10 }
```

6.3. Adres tablosunun güncel hali:

jdbc:h2:mem:ecommerce

Max rows: 1000

Auto complete: Off

Auto select: On

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM ADDRESSES

ID	CREATED_DATE	UPDATED_DATE	ADDRESS_TYPE	CITY	COUNTRY	STATE	STREET	ZIP_CODE	CUSTOMER_ID
1	2025-07-29 11:00:17.259895	2025-07-29 11:00:17.259895	SHIPPING	İzmir	Türkiye	İzmir	Gül Sokak No: 7	35000	1

(1 row, 1 ms)

Edit

7. Sepete Ürün Ekleme

7.1. Cart ve CartItem tablolarının boş hali:

jdbc:h2:mem:ecommerce

Max rows: 1000

Auto complete: Off

Auto select: On

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM CARTS

ID	CREATED_DATE	UPDATED_DATE	CUSTOMER_ID
----	--------------	--------------	-------------

(no rows, 2 ms)

Edit

jdbc:h2:mem:ecommerce

Max rows: 1000

Auto complete: Off

Auto select: On

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM CART_ITEMS

ID	CREATED_DATE	UPDATED_DATE	PRICE	QUANTITY	CART_ID	PRODUCT_ID
----	--------------	--------------	-------	----------	---------	------------

(no rows, 2 ms)

Edit

7.2. Postman ile sepete telefon ve kulaklık ekleme:



http://localhost:8080/api/customers/1/cart/items

POST



http://localhost:8080/api/customers/1/cart/items

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

JSON



```
1  {
2    "productId": 1, // Akıllı Telefon ID'si
3    "quantity": 1
4  }
```

Body

Cookies

Headers (8)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
4  "items": [
5    {
6      "id": 1,
7      "productId": 1,
8      "productName": "Akıllı Telefon X",
9      "productPrice": 12000.00,
10     "quantity": 1,
11     "totalPrice": 12000.00
12   },
13   {
14     "id": 1,
15     "productId": 1,
16     "productName": "Akıllı Telefon X",
17     "productPrice": 12000.00,
18     "quantity": 1,
19     "totalPrice": 12000.00
20   }
21 ],
22 "totalCartPrice": 24000.00
23 }
```



http://localhost:8080/api/customers/1/cart/items

POST



http://localhost:8080/api/customers/1/cart/items

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

JSON



```
1  {
2    "productId": 2, // Kablosuz Kulaklık ID'si
3    "quantity": 2
4  }
```

Body

Cookies

Headers (8)

Test Results

Pretty

Raw

Preview

Visualize

JSON



```
15  "productId": 2,
16  "productName": "Kablosuz Kulaklık Y",
17  "productPrice": 2500.00,
18  "quantity": 2,
19  "totalPrice": 5000.00
20 },
21 {
22   "id": 2,
23   "productId": 2,
24   "productName": "Kablosuz Kulaklık Y",
25   "productPrice": 2500.00,
26   "quantity": 2,
27   "totalPrice": 5000.00
28 }
29 ],
30 "totalCartPrice": 22000.00
31 }
```

HTTP <http://localhost:8080/api/customers/1/cart>

GET <http://localhost:8080/api/customers/1/cart>

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
7   "productId": 1,
8   "productName": "Akıllı Telefon X",
9   "productPrice": 12000.00,
10  "quantity": 1,
11  "totalPrice": 12000.00
12  },
13  {
14    "id": 2,
15    "productId": 2,
16    "productName": "Kablosuz Kulaklık Y",
17    "productPrice": 2500.00,
18    "quantity": 2,
19    "totalPrice": 5000.00
20  }
21  ],
22  "totalCartPrice": 17000.00
23 }
```

7.3. CartItem güncel hali:

jdbc:h2:mem:ecommerce

Run Run Selected Auto complete Clear SQL statement:

SELECT * FROM CART_ITEMS

SELECT * FROM CART_ITEMS;

ID	CREATED_DATE	UPDATED_DATE	PRICE	QUANTITY	CART_ID	PRODUCT_ID
1	2025-07-29 11:02:18.423145	2025-07-29 11:02:18.423145	12000.00	1	1	1
2	2025-07-29 11:05:00.902047	2025-07-29 11:05:00.902047	2500.00	2	1	2

(2 rows, 1 ms)

Edit

8. Sipariş Oluşturma

8.1. Postman ile sipariş oluşturma:

HTTP <http://localhost:8080/api/customers/1/orders>

POST <http://localhost:8080/api/customers/1/orders>

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "shippingAddressId": 1 // Ayşe'nin adres ID'si
3 }
```

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "customerId": 1,
4   "orderDate": "2025-07-29T11:09:31.9853142",
5   "orderStatus": "PROCESSING",
6   "shippingAddressId": 1,
7   "totalPrice": 17000.00,
8   "orderItems": [
9     {
10      "id": 1,
11      "productId": 1,
12      "productName": "Akıllı Telefon X",
13      "quantity": 1,
14      "price": 12000.00,
15      "totalPrice": 12000.00
16    },
17    {
18      "id": 2,
19      "productId": 2,
20      "productName": "Kablosuz Kulaklık Y",
21      "quantity": 2,
22      "price": 2500.00,
23      "totalPrice": 5000.00
24    }
25  ]
26 }
```

HTTP <http://localhost:8080/api/customers/1/orders>

POST <http://localhost:8080/api/customers/1/orders>

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "shippingAddressId": 1 // Ayşe'nin adres ID'si
3 }
```

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
10 {
11   "id": 1,
12   "productId": 1,
13   "productName": "Akıllı Telefon X",
14   "quantity": 1,
15   "price": 12000.00,
16   "totalPrice": 12000.00
17 },
18 {
19   "id": 2,
20   "productId": 2,
21   "productName": "Kablosuz Kulaklık Y",
22   "quantity": 2,
23   "price": 2500.00,
24   "totalPrice": 5000.00
25 }
26 ]
```


9. Stok Güncellenmesi

9.1. Sipariş sonrası ürün stoklarının güncellenmiş hali:

HTTP

http://localhost:8080/api/products/1

GET

http://localhost:8080/api/products/1

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

Key	Value
Key	Value

Body

Cookies

Headers (8)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1  {
2    "id": 1,
3    "name": "Akıllı Telefon X",
4    "description": "Yüksek performanslı son model telefon",
5    "price": 12000.00,
6    "stockQuantity": 4,
7    "imageUrl": "http://example.com/telefon.jpg"
8  }
```

HTTP

http://localhost:8080/api/products/2

GET

http://localhost:8080/api/products/2

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

Key	Value
Key	Value

Body

Cookies

Headers (8)

Test Results

Pretty

Raw

Preview


Visualize

JSON

```
1  {
2    "id": 2,
3    "name": "Kablosuz Kulaklık Y",
4    "description": "Gürültü engellemeli yüksek ses kalitesi",
5    "price": 2500.00,
6    "stockQuantity": 8,
7    "imageUrl": "http://example.com/kulaklik.jpg"
8  }
```

10. Siparişleri Listeleme

10.1. Postman ile Ayşe'nin sipariş geçmişini görme:

 http://localhost:8080/api/customers/1/orders/1

GET

▼

http://localhost:8080/api/customers/1/orders/1

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

	Key	Value
	Key	Value

Body

Cookies

Headers (8)

Test Results


Pretty

Raw


Preview

Visualize

JSON ▼



```
1  [
2    {
3      "id": 1,
4      "customerId": 1,
5      "orderDate": "2025-07-29T11:09:31.985314",
6      "orderStatus": "PROCESSING",
7      "shippingAddressId": 1,
8      "totalPrice": 17000.00,
9      "orderItems": [
10       {
11         "id": 1,
12         "productId": 1,
13         "productName": "Akillı Telefon X",
14         "quantity": 1,
15         "price": 12000.00,
16         "totalPrice": 12000.00
17       },
18     ]
19   }
20 ]
```

 http://localhost:8080/api/customers/1/orders/1

GET

▼

http://localhost:8080/api/customers/1/orders/1

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

	Key	Value
	Key	Value

Body

Cookies

Headers (8)

Test Results


Pretty

Raw

Preview

Visualize

JSON ▼



```
10  {
11    "id": 1,
12    "productId": 1,
13    "productName": "Akillı Telefon X",
14    "quantity": 1,
15    "price": 12000.00,
16    "totalPrice": 12000.00
17  },
18  {
19    "id": 2,
20    "productId": 2,
21    "productName": "Kablosuz Kulaklık Y",
22    "quantity": 2,
23    "price": 2500.00,
24    "totalPrice": 5000.00
25  }
26 ]
```

11. Hata Senaryosu: Stok Bitmesi

11.1. Stokta olmayan ürüne tekrar sipariş verme:

HTTP <http://localhost:8080/api/customers/1/cart/items>

POST <http://localhost:8080/api/customers/1/cart/items>

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary **JSON** ☐

```
1 {
2   "productId": 1, // Akıllı Telefon X ID'si
3   "quantity": 5   // Stokta 4 vardı, 5 istiyoruz
4 }
```

Body Cookies Headers (7) Test Results 🌐 Status: 400

Pretty Raw Preview Visualize **JSON** ☐

```
1 {
2   "timestamp": "2025-07-29T11:13:36.7518253",
3   "status": 400,
4   "error": "Bad Request",
5   "message": "Not enough stock for product: Akıllı Telefon X. Available: 4, Requested: 5",
6   "path": "uri=/api/customers/1/cart/items"
7 }
```

12. Genel Değerlendirme

Bu projede temel bir e-ticaret sisteminin uçtan uca işlemleri başarıyla gerçekleştirilmiştir. Her bir endpoint için veritabanı öncesi ve sonrası durum gözlemlenmiş, senaryo başarıyla tamamlanmıştır.