

My Projects

İçecek / Yiyecek Otomat Makine Tasarımı

Case Projede Kullanılan Tasarım Kalıpları ve Frameworkler

- Repository Pattern
 - CQRS Pattern
 - MediatR Pattern (MediatR Framework kullanıldı.)
 - Dependency injection (Microsoft.Extensions.DependencyInjection framework kullanıldı.)
 - ORM => Entity Framework
 - Validasyon => FluentValidation
 - Mapping => AutoMapper
 - Test
 - xUnit
 - Moq
 - Fluent Assertions
 - Microsoft.AspNetCore.Mvc Testing
-

Api Projeleri

Projede 4 api projesi oluşturuldu.

- ShoppingCartApi : Seçim işlemleri bu api üzerinde yapılıyor.
 - OrderApi : Sipariş oluşumu bu api üzerinde yapılıyor.
 - ProductApi : Ürün ve Kategori işlemleri bu api üzerinde yapılıyor.
 - ManagementApi : Yeni bir ödeme seçeneği bu api üzerinden yapılıyor.
-

Proje Adım Senaryosu



Doğru istek yapabilmek için Slot - Ürün eşleştirmelerini görmek için Sql tarafında oluşturulan View a sorgu atılması yardımcı olacaktır.

```
Select * From vw_AutomatProducts
```

- Ürün Seçimi

POST

AddToCart

<https://localhost:44320/api/shopping/addtocart>

Ürün seçiminin yapıldığı endpointtir. Slot, ürün ve ürün seçenek bilgileri ile istek yapılıyor.

Request

Body Parameters

featureOptionQuantity

OPTIONAL

integer

Ürün seçenek adet

featureOptionId

OPTIONAL

integer

Ürün seçenek Id

productId

REQUIRED

integer

Ürün Id

slotId

REQUIRED

integer

Seçim yapılan slot Id

Response

● 200: OK

Ürün sepete ekleme sonrası response bilgisi

```
1  {
2    "success": true,
3    "message": "OK",
4    "statusCode": 200,
5    "result": {
6      "processId": "3292e2db-d012-4207-9913-b02cbdde387f",
7      "cartItem": {
8        "productId": 52,
9        "productName": "Schweppes Mandalina",
10       "featureOptionId": 3,
11       "featureName": "Buzlu",
12       "featureQuantity": 3
13     },
14     "message": "Ürün seçimi yapıldı."
15   },
16   "errorResult": null
17 }
```

Örnek Request Body

```
{ "slotId": 11, "productId": 52, "featureOptionId": 3, "featureOptionQuantity": 3 }
```

- Adet Seçimi

POST SelectProductQuantity

<https://localhost:44320/api/shopping/selectquantity>

Sepete ürün ekleme yapıldıktan sonra adet seçimi adımına geçiyoruz. Bu endpointte adet bilgisi seçimi yapıyoruz. İstek yaparken ürün ekleme sonrası oluşan processId ile istek yapıyoruz.

Request

Body Parameters

processId

REQUIRED

string

AddToCart sonrası oluşan ProcessId

productId

REQUIRED

integer

Ürün Id

quantity

REQUIRED

integer

Ürün adedi

Response

● 200: OK

Ürün adet seçimi sonrası response bilgisi

```
1  {
2    "success": true,
3    "message": "OK",
4    "statusCode": 200,
5    "result": {
6      "processId": "3292e2db-d012-4207-9913-b02cbdde387f",
7      "message": "Adet seçimi yapıldı. Schweppes Mandalina ürününd",
8      "productId": 52,
9      "productName": "Schweppes Mandalina",
10     "quantity": 3
11   },
12   "errorResult": null
13 }
```

Örnek Request Body

```
{ "processId": "3292e2db-d012-4207-9913-b02cbdde387f", "productId": 52,
  "quantity": 3 }
```

- Ödeme Yöntemi Seçimi

POST SelectPaymentMethod

<https://localhost:44320/api/shopping/selectpaymentmethod>

Ödeme yapıldığın enpointtir. İstek yaparken processId ile istek yapıp ödeme yapılacak yöntemi seçiyoruz.

Request

Body Parameters

processId

REQUIRED

string

İşlem Id

paymentTypeOptionId

REQUIRED

integer

Ödeme Seçenek Id

Response

● 200: OK

Ödeme seçimi yapıldıktan sonra dönen response bilgisi

```
1  {
2    "success": true,
3    "message": "OK",
4    "statusCode": 200,
5    "result": {
6      "processId": "3292e2db-d012-4207-9913-b02cbdde387f",
7      "paymentTypeId": 2,
8      "paymentTypeName": "Nakit",
9      "paymentTypeOptionId": 4,
10     "paymentTypeOptionName": "Kağıt Para",
11     "message": "Ödeme tipi seçimi yapıldı. Kağıt Para olarak öde",
12   },
13   "errorResult": null
14 }
```

Örnek Request Body

```
{ "processId": "3292e2db-d012-4207-9913-b02cbdde387f",
  "paymentTypeOptionId": 4 }
```

- Sipariş Oluřturma

POST Pay

<https://localhost:44378/api/order/pay>

Tüm seçimler tamamlandıktan sonraki final işlemi olan ödeme yapıp sipariş oluřturma endpointidir.

OrderApi ye process id ile işlem yapıp sipariş verme işlemi yapılıyor

Request

Body Parameters

processId

REQUIRED

string

İşlem yapılan Id

paidMoney

REQUIRED

object

Yükleme yapılan tutar

Response

● 200: OK

Sipariş oluşumu sonrası dönen endpoint bilgisi

```
1  {
2    "success": true,
3    "message": "OK",
4    "statusCode": 200,
5    "result": {
6      "orderId": 9,
7      "orderCode": "1d82c4b3-9fcb-45eb-a6f8-d0a00fdc882f",
8      "orderDate": "2021-08-30T18:59:50.2180394+03:00",
9      "message": "Siparişiniz oluşturuldu!",
10     "refundAmount": 27.50,
11     "paymentTotal": 22.50,
12     "product": {
13       "productId": 52,
14       "productName": "Schweppes Mandalina",
15       "categoryId": 2,
16       "categoryName": "Soğuk İçecek",
17       "unitPrice": 7.50,
18       "quantity": 3,
19       "featureOptionId": 3,
20       "featureName": "Buzlu",
21       "featureQuantity": 3
22     },
23     "paymentMethod": {
24       "paymentTypeId": 2,
25       "paymentTypeName": "Nakit",
26       "paymentTypeOptionId": 4,
27       "paymentTypeOptionName": "Kağıt Para"
28     }
29   },
30   "errorResult": null
31 }
```



Örnek Requist Body

```
{ "processId": "3292e2db-d012-4207-9913-b02cbdde387f", "paidMoney": 50 }
```

- Ödeme Bilgisini Görüntüleme

GET

Get Order

<https://localhost:44378/api/orders/9>

Sipariş detaylarını görüntülemek için kullanılan enpointtir.

Request

Headers

id

REQUIRED

integer

Sipariş Id Bilgisi

Response

● 200: OK

Sipariş bilgileri çekimi sonrası oluşan response bilgisi

```
1  {
2    "success": true,
3    "message": "OK",
4    "statusCode": 200,
5    "result": {
6      "orderId": 9,
7      "orderCode": "1d82c4b3-9fcb-45eb-a6f8-d0a00fdc882f",
8      "orderDate": "2021-08-30T18:59:50.217",
9      "message": "Siparişiniz bilgileri çekildi.",
10     "refundAmount": 27.50,
11     "paymentTotal": 22.50,
12     "product": {
13       "productId": 52,
14       "productName": "Schweppes Mandalina",
15       "categoryId": 2,
16       "categoryName": "Soğuk İçecek",
17       "unitPrice": 7.50,
18       "quantity": 3,
19       "featureOptionId": 3,
20       "featureName": "Buzlu",
21       "featureQuantity": 3
22     },
23     "paymentMethod": {
24       "paymentTypeId": 2,
25       "paymentTypeName": "Nakit",
26       "paymentTypeOptionId": 4,
27       "paymentTypeOptionName": "Kağıt Para"
28     }
29   },
30   "errorResult": null
31 }
```



Bu endpoint bilgilerinin dışında Ürün, Kategori CRUD işlemlerini yapan endpointler ve Ödeme Yöntemi ve Seçenekleri insert işlemlerini yapan endpoint bilgileride bulunmaktadır.

Postman Collection bilgilerini proje klasörü içinde görebilirsiniz.

Test Code Coverage

Son yapılan analiz sonrası unittest ve integration test code coverage bilgisi görseldeki gibidir.

