

MİGROS YEMEK API DOKÜMANI

(GOURMET)

Versiyon	Tarih	Düzenleyen	Açıklama
1.2.0	16.01.2025	Uğur Can Balcı	Eklendi: StoreArea/GetStoreAreas Eklendi: StoreArea/ActivateArea Eklendi: StoreArea/DeActivateArea Düzenlendi: Order Created Webhook Model
1.1.2	05.08.2024	Uğur Can Balcı	Eklendi: Receipt/AddOrderReceipt Eklendi: Receipt/GetReceipts
1.1.1	01.06.2024	Uğur Can Balcı	Düzenlendi: Delivery Status Change Webhook Yeni Durum Eklenmesi
1.1.0	06.03.2024	Ali Şimşek	Eklendi: Order/v2/UpdateOrderStatus Eklendi: Order/v2/CancelOrder Eklendi: Mapping/v2/GetCancelReasons Kaldırılacak: Order/UpdateOrderStatus Kaldırılacak: Order/CancelOrder Kaldırılacak: Mapping/GetCancelReasons
1.0.10	21.08.2023	Uğur Can Balcı	Eklendi: Store/GetStoreViewStatus Eklendi: Store/AddStoreOffDate Eklendi: Store/RemoveStoreOffDate Eklendi: Menu/GetMenuDetailsByStoreAndStoreGroupId Eklendi: Menu/UpdateProductStatusByStoreId
1.0.9	19.06.2023	Uğur Can Balcı	Eklendi: OptionItem/ActivateOptionItemStatusByStore Eklendi: OptionItem/DeActivateOptionItemStatusByStore
1.0.8	16.05.2023	Uğur Can Balcı	Eklendi: Order/GetSummarizedOrder Eklendi: Migros Yemek Teslimatı için DeliveryStatusChange webhook Düzenlendi: Order/PendingOrdersWithStores metoduna, teslimat sağlayıcısı ve kurye adı ile ilgili alanlar eklendi Düzenlendi: Order/ActiveOrdersWithStores metoduna, teslimat sağlayıcısı ve kurye adı ile ilgili alanlar eklendi

			Düzenlendi: Order/GetPassiveOrders metoduna, teslimat sağlayıcısı ve kurye adı ile ilgili alanlar eklendi
1.0.7	18.04.2023	Uğur Can Balcı	Eklendi: Menu/UpdateBatchPrice
1.0.6	20.03.2023	Uğur Can Balcı	<p>Düzenlendi: Order/PendingOrdersWithStores metoduna, sponsorlara göre indirimli tutar bilgisi ile ilgili alanlar eklendi.</p> <p>Düzenlendi: Order/ActiveOrdersWithStores metoduna, sponsorlara göre indirimli tutar bilgisi ile ilgili alanlar eklendi.</p> <p>Düzenlendi: Order/GetPassiveOrders metoduna, sponsorlara göre indirimli tutar bilgisi ile ilgili alanlar eklendi.</p> <p>Düzenlendi: Order Created webhook metoduna sponsorlara göre indirimli tutar bilgisi ile ilgili alanlar eklendi.</p>
1.0.5	24.01.2023	Uğur Can Balcı	<p>Eklendi: Mapping/GetCancelReasons</p> <p>Düzenlendi: Order/UpdateOrderStatus (Request modeline cancelReason alanı eklendi)</p> <p>Düzenlendi: Order/PendingOrdersWithStores metodu içerisine teslimat adresine ait enlem-boylam alanları eklendi.</p> <p>Kaldırıldı: Menu/AssigneMenuToMultipleStore</p> <p>Kaldırıldı: Menu/UpdateMenuHeader</p> <p>Kaldırıldı: Menu/ActivateMenuHeader</p> <p>Kaldırıldı: Menu/DeactivateMenuHeader</p> <p>Kaldırıldı: Menu/UpdateMenuHeaderPriority</p> <p>Kaldırıldı: Menu/UpdateMenuHeaderItemPriority</p> <p>Kaldırıldı: Menu/UpdateMenuHeaderItem</p> <p>Kaldırıldı: Order/PassiveOrdersByStoreGroupIds</p> <p>Kaldırıldı: Order PendingOrdersByStoreGroupIds</p> <p>Kaldırıldı: Order/ActiveOrdersByStoreGroupIds</p>
1.0.4	24/11/2022	Uğur Can Balcı	Düzenlendi: Order/UpdateOrderStatus (Request Model Düzenlemesi – StoreId Eklendi)
1.0.3	17/10/2022	Uğur Can Balcı	<p>Eklendi: PaymentMethod/GetPaymentMethodsByStoreId</p> <p>Eklendi: PaymentMethod/UpdatePaymentMethodStatus</p> <p>Eklendi: Menu/UpdateProductStatusByStoreId</p> <p>Düzenlendi: Order/PendingOrdersByStoreGroupIds (Response Model Düzenlemesi – Ödeme Yöntemi, Konum, Opsiyon Tipi, Kurye Tercihi)</p> <p>Düzenlendi: Order/PassiveOrdersByStoreGroupIds (Response Model</p>

			<p>Düzenlemesi – Ödeme Yöntemi, Konum, Opsiyon Tipi, Kurye Tercihi)</p> <p>Düzenlendi: Order/ActiveOrdersByStoreGroupIds (Response Model Düzenlemesi – Ödeme Yöntemi, Konum, Opsiyon Tipi, Kurye Tercihi)</p> <p>Düzenlendi: Order/PendingOrdersWithStores (Response Model Düzenlemesi – Ödeme Yöntemi, Opsiyon Tipi)</p> <p>Düzenlendi: Order/GetPassiveOrders (Response Model Düzenlemesi – Ödeme Yöntemi, Konum ,Opsiyon Tipi)</p> <p>Düzenlendi: Order/ActiveOrdersWithStores (Response Model Düzenlemesi – Ödeme Yöntemi, Konum ,Opsiyon Tipi)</p>
1.0.2	12/09/2022	Uğur Can Balcı	<p>Kaldırıldı: Menu/CreateMenu</p> <p>Kaldırıldı: Menu/DeleteMenu</p> <p>Kaldırıldı: Menu/CreateMenuHeader</p> <p>Kaldırıldı: Menu/DeleteMenuHeader</p> <p>Kaldırıldı: Menu/DeleteMenuHeaderItem</p> <p>Kaldırıldı: Menu/CreateMenuHeaderItem</p> <p>Kaldırıldı: Order/OrderCountsWithStores</p> <p>Kaldırıldı: CheckPendingOrdersWithStores</p>
1.0.1	06/09/2022	Uğur Can Balcı	<p>Kaldırıldı: Menu/ActivateMenu</p> <p>Kaldırıldı: Menu/DeactivateMenu</p> <p>Kaldırıldı: Menu/UpdateMenuItemPrice</p> <p>Kaldırıldı: WorkingHour/UpdateActivenessStoreTimeSlot</p> <p>Kaldırıldı: Order/GetOrders</p> <p>Kaldırıldı: Order/GetOrders kaldırıldığı için</p> <p>Kaldırıldı: Mapping/GetOrderFilterList</p> <p>Eklendi: Order/GetPassiveOrders</p> <p>Düzenlendi: Order/PendingOrdersByStoreGroupIds (Response Model Düzenlemesi)</p> <p>Düzenlendi: Order/PassiveOrdersByStoreGroupIds (Response Model Düzenlemesi)</p> <p>Düzenlendi: Order/ActiveOrdersByStoreGroupIds (Response Model Düzenlemesi)</p> <p>Düzenlendi: Order/CancelOrder (Request Model'e UserId eklendi. Sipariş bilgisi içerisinde dönen long tipindeki Müşteri Id'si)</p> <p>Not: Servislerin açıklamasında yer alan offset bilgileri 0 olarak düzeltildi. Buradaki offset bilgisi n. Sayfa sayısı anlamına gelmekte ve 0 dan başlamaktadır.</p>

1.0.0	08/08/2022	Uğur Can Balci	İlk sürüm
-------	------------	-------------------	-----------

İçindekiler

1	Giriş	7
1.1	Şifreleme ve Yetkilendirme	7
1.2	Periyodik Sipariş Sorgulama Akışı	7
1.3	Webhook Akışı	7
2	Ters Akış	8
2.1	Sipariş Oluştur (Order Created Webhook)	8
2.2	Sipariş İptal Edildi (Order Canceled Webhook)	13
2.3	Kurye Durumu Değişti (Delivery Status Changed Webhook)	14
3	Mapping	15
3.1	GetOrderStatusList	15
3.2	V2/GetCancelReasons	16
3.3	GetCancelReasons	17
4	Menu	18
4.1	MenuByStoreGroupId	18
4.2	MenuDetails	19
4.3	GetMenuItems	22
4.4	GetStoreMenusByStoreIds	23
4.5	GetAllOptionsByMenuId	24
4.6	UpdateProductStatusByStoreId	27
4.7	UpdateBatchPrice	28
4.8	GetMenuDetailsByStoreAndStoreGroupId	30
5	Order	33
5.1	PendingOrdersWithStores	33
5.2	ActiveOrdersWithStores	38
5.3	GetPassiveOrders	44
5.4	V2/UpdateOrderStatus	49
5.5	V2/CancelOrder	51
5.6	GetSummarizedOrder	52
5.7	UpdateOrderStatus	54
5.8	CancelOrder	56
6	Payment Methods	57
6.1	GetPaymentMethodsByStoreId	57
6.2	UpdatePaymentMethodStatus	59
7	Store	60
7.1	GetStoreDetail	60
7.2	GetDefinedActiveRestaurantApiKeys	62
7.3	GetStoreGroups	64

7.4	GetStores	65
7.5	ActivateStore	67
7.6	DeActivateStore	68
7.7	GetStoreViewStatus.....	69
7.8	AddStoreOffDate	71
7.9	RemoveStoreOffDate	72
8	WorkingHours.....	73
8.1	GetStoreTimeSlot.....	73
8.2	UpsertStoreTimeSlot	74
9	OptionItem	75
9.1	ActivateOptionItemStatusByStore	75
9.2	DeActivateOptionItemStatusByStore	76
10	Receipt.....	77
10.1	AddOrderReceipt	77
10.2	GetReceipts.....	78
11	StoreArea.....	79
11.1	GetStoreAreas.....	79
11.2	ActivateArea	80
11.3	DeActivateArea	81

1 Giriş

1.1 Şifreleme ve Yetkilendirme

Migros Yemek Entegrasyonu amacı ile, entegratörlere test ve canlı ortam için birer adet secret key verilmektedir. Bu secret key bilgisi, şifrelemede kullanılmaktadır.

Mevcut sistemdeki **Store/GetDefinedActiveRestaurantApiKeys** haricindeki tüm **POST** metodlarına ileteceğiniz json body'lerin, sizlere iletilecek **secret key** ile **Rijndael - AES** şifreleme algoritmasına göre şifrenmesi beklenmektedir. Şifrenmiş olan veriyi, aşağıdaki formatta iletmeniz gerekmektedir.

```
{  
  "value": "ŞİFRELENMİŞ JSON BODY"  
}
```

Buna ek olarak, işlem yapmak istediğiniz Restorana ait api key ise göndereceğiniz requeste **"XApiKey"** header'ı ekleyerek değer olarak Restoran Api Keyi iletmenizi beklemekteyiz.

Burada, Restoran Api Key, public key işlevi görmekte iken, sizlere iletmiş olacağımız secret key ise private key görevi görecektir.

Şifrelemelerinizin doğru olup olmadığını kontrol etmeniz için Test route'u içerisinde **EncryptJson** ve **DecryptJson** metodları bulunmaktadır.

Şifrelemede kullanılacak parametreler şunlardır:

Chipper Mode = **ECB**

Padding = **PKCS7**

Block Size = **128**

KeySize = **256**

1.2 Periyodik Sipariş Sorgulama Akışı

Siparişlerin periyodik olarak sorgulanması için **Order/PendingOrdersWithStores** metodu kullanılmaktadır. Limit parametresi maksimum 20 olabilmektedir, offset parametresi ise sayfalama için kullanılmakta olup, kaç adet sipariştten sonrasını getirmek istediğinizi belirtir. Offset:0 olarak tanımlarsanız ilk 20 siparişi, Offset:20 olarak tanımlarsanız ilk 20 sipariştten sonraki 20 siparişi getirecektir. Onaylanan ya da onay süresi dolmuş siparişier bu servisin bir sonraki çağrımında artık gelmeyecektir.

1.3 Webhook Akışı

Webhook yönteminde, siparişler oluştuğunda, entegratörün bizlere ileteceği bir post metoduna aktarımı sağlanır. Hata alınması durumunda ilk hatada 10 saniye sonra, ikinci hatada 20 saniye sonra, üçüncü hata da ise 30 saniye sonra iletim tekrar denenmektedir. 3. Denemede başarısız olunması halinde artık sipariş iletimi gerçekleşmeyecektir.

Sipariş oluşması durumuna ek olarak, siparişin herhangi bir sebepten reddi veya iptali gerçekleştiğinde çağırabileceğimiz bir sipariş iptal metodu da gerekmektedir.

2 Ters Akış

2.1 Sipariş Oluştur (Order Created Webhook)

Sipariş oluştuğunda, ilgili entegratör firmaya ait POST metoduna istek atılmaktadır. İstek atılan metod, basic authentication ile yetkilendirilmelidir. Diğer yetkilendirme yöntemleri, mevcut yapıda işlenememektedir.

Retry yapısı: Sipariş oluştu bilgisi iletilirken, 3 kez deneme yapılır. Her deneme arasında sırasıyla 10-20-30 saniye bulunmaktadır. 3. Denemede de işlem başarısız olur ise, tekrar sipariş aktarımı denemesi gerçekleştirilmez.

Model

```
{
  "id": 0,
  "description": "string",
  "status": "string",
  "deliveryProvider": "string",
  "store": {
    "id": 0,
    "name": "string",
    "group": {
      "id": 0,
      "name": "string"
    }
  },
  "customer": {
    "id": 0,
    "firstName": "string",
    "lastName": "string",
    "fullName": "string",
    "phoneNumber": "string",
    "deliveryAddress": {
      "id": 0,
      "direction": "string",
      "detail": "string",
      "city": {
        "id": 0,
        "name": "string"
      },
      "town": {
        "id": 0,
        "name": "string"
      },
      "district": {
        "id": 0,
        "name": "string"
      },
      "geoLocation": {
        "latitude": 0,
        "longitude": 0
      }
    }
  },
  "prices": {
    "total": {
      "amountAsPenny": 0,
      "text": "string"
    },
    "discounted": {
      "amountAsPenny": 0,
      "text": "string"
    },
    "restaurantDiscounted": {
      "amountAsPenny": 0,
      "text": "string"
    },
    "migrosDiscounted": {
      "amountAsPenny": 0,
```



```

    "text": "string"
  }
},
"items": [
  {
    "id": 0,
    "productId": 0,
    "name": "string",
    "price": 0,
    "priceText": "string",
    "unitPrice": 0,
    "unitPriceText": "string",
    "amount": 0,
    "note": "string",
    "options": [
      {
        "optionHeaderId": 0,
        "objectOptionItemId": 0,
        "optionItemId": 0,
        "productId": 0,
        "parentObjectOptionItemId": 0,
        "headerName": "string",
        "itemNames": "string",
        "primaryDiscountedPrice": 0,
        "primaryPrice": 0,
        "primaryDiscountedPriceText": "string",
        "primaryPriceText": "string",
        "unitPrimaryPrice": 0,
        "unitPrimaryPriceText": "string",
        "quantity": 0,
        "excluded": true,
        "subOptions": [
          {
            "optionHeaderId": 0,
            "objectOptionItemId": 0,
            "optionItemId": 0,
            "productId": 0,
            "parentObjectOptionItemId": 0,
            "headerName": "string",
            "itemNames": "string",
            "primaryDiscountedPrice": 0,
            "primaryPrice": 0,
            "primaryDiscountedPriceText": "string",
            "primaryPriceText": "string",
            "unitPrimaryPrice": 0,
            "unitPrimaryPriceText": "string",
            "quantity": 0,
            "excluded": true,
            "subOptions": [],
            "objectOptionHeaderPriority": 0,
            "objectOptionItemPriority": 0,
            "optionType": "string"
          }
        ],
        "optionType": "string",
        "objectOptionHeaderIsMainHeader": true
      }
    ]
  }
],
"payment": {
  "type": {
    "hashedNameId": 0,
    "name": "string",
    "description": "string",
    "isOnlinePayment": true,
    "simplifiedName": "string",
    "simplifiedHashedNameId": 0
  }
},
"extendedProperties": {
  "orderNote": "string",
  "saveGreen": true,
  "contactlessDelivery": true,

```

```
"ringDoorBell": true,  
  "shortCode": "string"  
},  
"log": {  
  "createdAsMs": 0  
}  
}
```

- **id** (long): Sipariş Id bilgisidir, Migros Yemek paneli üzerindeki sipariş numarası bilgisi ile aynıdır.
- **description** (string): Siparişin özet metnini içerir. Ana ürün başlıkları yan yana virgülle ayrılmış şekilde gelmektedir.
- **status** (string): Siparişin durumu bilgisini verir.
- **deliveryProvider** (string): Teslimatın kim tarafından yapılacağını belirtir. "RESTAURANT" gelir ise, restoran kendi kuryesi ile teslim edecek, "MİGROS" gelir ise, Migros kuryeleri ile teslimat gerçekleştirilecektir.
- **store** (object): Siparişin geldiği restoran bilgilerini içerir.
 - **id** (long): Restoran Id bilgisidir.
 - **name** (string): Restoran adı bilgisidir.
 - **group** (object): Restoranın bağlı olduğu marka/zincir bilgisine ait modeldir. Her restoran, tek bir şubeden oluşuyor olsa dahi bir group (StoreGroup) kaydı bulunmaktadır.
 - **id** (long): Marka/Zincir Id bilgisidir.
 - **name** (string): Marka/Zincir adı bilgisidir.
- **customer** (object): Siparişi veren müşteriye ait bilgileri içeren modeldir.
 - **id** (long): Müşteri id bilgisidir.
 - **firstName** (string): Müşteri adı bilgisidir.
 - **lastName** (string): Müşteri soyadı bilgisidir.
 - **fullName** (string): Müşteri Adı ve Soyadı bilgilerinin birleştirilmiş halidir.
 - **phoneNumber** (string): Müşterinin telefon numarası bilgisidir.
 - **deliveryAddress** (object): Teslimat adresine ait bilgileri içeren modeldir.
 - **id** (long): Adresin id bilgisidir.
 - **direction** (string): Adres tarifi bilgisidir.
 - **detail** (string): Adres detay bilgisidir.
 - **city** (object): Adresin bulunduğu şehir bilgilerini içeren modeldir.
 - **id** (long): Şehir id bilgisidir.
 - **name** (string): Şehir adıdır.
 - **town** (object): Adresin bulunduğu ilçe bilgilerini içeren modeldir.
 - **id** (long): İlçe id bilgisidir.
 - **name** (string): İlçe adıdır.
 - **district** (object): Adresin bulunduğu mahalle bilgisini içeren modeldir.
 - **id** (long): Mahalle id bilgisidir.
 - **name** (string): Mahalle adıdır.

--- **geoLocation** (object) : İlgili siparişin konum bilgilerini içeren modeldir.

---- **latitude** (long): Konuma ait enlem bilgisidir.

---- **longitude** (long): Konuma ait boylam bilgisidir.

- **prices** (object): Sipariş tutar bilgilerini içeren modeldir.

-- **total** (object): Toplam sipariş tutarı bilgilerini içeren modeldir.

--- **amountAsPenny** (long): Kuruş cinsinden siparişin tutar bilgisidir. Örn: 1000

--- **text** (string): TL cinsinden siparişin tutar bilgisidir. Örn: 10.00 TL

-- **discounted** (object): Migros Yemek ve Restoran indirimlerinin sipariş tutarından düşülmesinden sonraki indirimli tutar bilgilerini içeren modeldir.

--- **amountAsPenny** (long): Kuruş cinsinden siparişin indirimli tutar bilgisidir. Örn: 1000

--- **text** (string): TL cinsinden siparişin indirimli tutar bilgisidir. Örn: 10.00 TL

-- **restaurantDiscounted** (object): Restoranın kendi yaptığı indirimlerin sipariş tutarından düşülmesinden sonraki indirimli tutar bilgilerini içeren modeldir.

--- **amountAsPenny** (long): Kuruş cinsinden siparişin indirimli tutar bilgisidir. Örn: 1000

--- **text** (string): TL cinsinden siparişin indirimli tutar bilgisidir. Örn: 10.00 TL

-- **migrosDiscounted** (object): Migros Yemek Kupon indirimlerinin sipariş tutarından düşülmesinden sonraki indirimli tutar bilgilerini içeren modeldir.

--- **amountAsPenny** (long): Kuruş cinsinden siparişin indirimli tutar bilgisidir. Örn: 1000

--- **text** (string): TL cinsinden siparişin indirimli tutar bilgisidir. Örn: 10.00 TL

- **items** (object array): Siparişin içerisinde bulunan ürün ve ürün altındaki opsiyon bilgilerini içeren modeldir.

-- **id** (long): Sipariş bilgisi ile ürün bilgisinin ilişkisini sağlayan sipariş satırı bilgisidir.

-- **productId** (long): İlgili sipariş satırındaki ürün bilgisine ait id bilgisidir.

-- **price** (long): İlgili ürün fiyatının kuruş cinsinden tutar bilgisidir. Adet (amount) ile çarpılmış olarak gelmektedir.

-- **priceText** (string): İlgili ürün fiyatının TL cinsinden tutar bilgisidir. Adet (amount) ile çarpılmış olarak gelmektedir.

-- **amount** (int): İlgili sipariş satırındaki ürüne ait adet bilgisidir.

-- **note** (string): Ürün notu bilgisidir.

-- **options** (object array): İlgili ürüne ait opsiyon (Yan ürün, Çıkarılacak Malzeme, Promosyon, Porsiyon Seçimi vb.) bilgilerini içeren liste halindeki bilgidir.

--- **optionHeaderId** (long): İlgili opsiyon başlığına ait id bilgisidir.

--- **objectOptionHeaderId** (long): İlgili opsiyon başlığının, ürün ile eşleştirilmesi ile oluşan ara tablo id bilgisidir.

--- **optionItemId** (long): İlgili opsiyonun id bilgisidir. Aynı isimdeki diğer opsiyonlar, aynı menü içerisinde her zaman aynı id bilgisine sahiptir. Yalnızca, aynı isimde aynı menü altında çıkarılacak malzeme tipinde bir opsiyon bilgisi var ise, böyle bir durumda iki farklı id oluşur. Biri çıkarılacak malzeme için, bir diğeri ise normal opsiyon bilgisi için.

--- **productId** (long): İlgili opsiyon başlığına opsiyon olarak, menüde var olan bir ürün seçilebilir. Böyle bir durumda, opsiyon olarak seçilen ürünün id bilgisi bu alanda gelmektedir. Null gelebilir.

--- **parentObjectOptionItemID** (long): Parent-child olarak tanımlanan opsiyonlarda, child opsiyonun içerisinde, parent opsiyona ait objectOptionHeaderId bilgisini barındırır. Bir üst katmanında başka bir opsiyon bulunmayan tanımlamalar için -1 gelir.

--- **headerName** (string): ilgili opsiyon başlığının adını içeren alandır. (Örn: Porsiyon Seçimi)

--- **itemNames** (string): İlgili opsiyona ait isim bilgisini getirir. (Örn: 1.5 Porsiyon)

--- **primaryPrice**: (long): İlgili opsiyonun kuruş cinsinden fiyat bilgisini içeren alandır. Adet (quantity) ile çarpılarak hesaplanmış olarak getirilir. Adet olarak seçilmeyen, yani adet bilgisi 0 gelen seçimlerde ise, tekil fiyat gelmektedir.

--- **primaryPriceText** (string): İlgili opsiyonun TL cinsinden fiyat bilgisini içeren alandır. Adet (quantity) ile çarpılarak hesaplanmış olarak getirilir. Adet olarak seçilmeyen, yani adet bilgisi 0 gelen seçimlerde ise, tekil fiyat gelmektedir.

--- **primaryDiscountedPrice** (long): İlgili opsiyonun kuruş cinsinden fiyat bilgisini içeren alandır. Adet (quantity) ile çarpılarak hesaplanmış olarak getirilir. Adet olarak seçilmeyen, yani adet bilgisi 0 gelen seçimlerde ise, tekil fiyat gelmektedir.

--- **primaryDiscountedPriceText** (string): İlgili opsiyonun TL cinsinden fiyat bilgisini içeren alandır. Adet (quantity) ile çarpılarak hesaplanmış olarak getirilir. Adet olarak seçilmeyen, yani adet bilgisi 0 gelen seçimlerde ise, tekil fiyat gelmektedir.

--- **quantity** (int): Opsiyonun adetini belirten alandır. Adet olarak seçilemeyen opsiyonlar da bulunduğundan (Örn: İnce Hamur) bu gibi durumlarda 0 da gelebilmektedir.

--- **exluded** (bool): İlgili opsiyonun, çıkarılabilir bir malzemeyi temsil edip etmediği bilgisini barındırmaktadır. Eğer bu alanın değeri True ise, ilgili opsiyon, müşteri tarafından üründen çıkarılmak istendiği bilgisini belirtir. False ise, çıkarılacak malzeme olmadığı anlamına gelmektedir.

--- **subOptions** (object array): Bir opsiyona ait alt opsiyon bilgilerini içeren listedir. **options** alanı ile aynı sınıf yapısındadır ve parent-child olarak, sonsuz kere alt kademelere gidebilmektedir. Her ne kadar müşteri deneyimi konusunda problem yaşanmaması adına, genelde alt alta 4. Kademeden daha derin bir opsiyon yapıları kurgulamak tercih edilmese dahi, teknik olarak bir sınır belirtilmemektedir.

--- **optionType** (string): İlgili opsiyonun hangi tipte olduğu bilgisini belirtir. 3 farklı gelebilecek parametre mevcuttur. Bunlar; PRODUCT,INGREDIENT,NONE seçenekleridir. Kısaca açıklamak gerekirse

- **PRODUCT**: İlgili opsiyonun, menüde bulunan başka bir ürün olduğu anlamına gelmektedir. Böyle bir durumda ilgili nesnenin ProductId alanı da ilgili ürüne ait id bilgisini barındırmaktadır.
- **INGREDIENT**: İlgili opsiyonun, çıkarılacak bir malzeme anlamına geldiğini ifade etmektedir. Böyle bir durumda excluded alanı true gelecektir.
- **NONE**: İlgili opsiyonun, standart bir opsiyon seçeneği olduğu anlamına gelecektir.

-- **objectOptionHeaderIsMainHeader** (bool): İlgili opsiyonun, bağlı olduğu ürünün ana seçimi olup olmadığı bilgisini içerir.

- **payment** (object): İlgili siparişin, hangi ödeme yöntemi ile ödendiği-ödeneyeceği bilgisini barındıran modeldir.

-- **type** (object): İlgili siparişin ödeme tipinin detaylarını barındıran bilgidir.

--- **name** (string): Ödeme yönteminin, tip bilgisini içeren ve bir enum değere karışık gelen alandır.

--- **description** (string): İlgili ödeme yönteminin açıklamasıdır.

--- **hashedNameId** (int): İlgili ödeme yönteminin enum karşılığının Id bilgisidir.

--- **isOnlinePayment** (bool): İlgili ödeme tipinin, online bir ödeme tipi olup olmadığı bilgisini barındırmaktadır. True ise ilgili sipariş online olarak ödenmiştir. False ise, kapıda ödeme yöntemlerinden biri ile ödenecektir.

--- **simplifiedName** (string): Ödeme yöntemlerinden, eğer online bir ödeme yöntemi ile ödenmiş ise doğrudan "ONLINE_PAYMENT" olarak basitleştirilmiş ödeme adını içermektedir. Eğer ödeme yöntemi, kapıda ödeme yöntemlerinden biri ise, ilgili ödeme yönteminin enum karşılığını belirtir.

--- **simplifiedHashedNameId** (int): Ödeme yöntemlerinden, eğer online bir ödeme yöntemi ile ödenmiş ise doğrudan 1 olarak basitleştirilmiş ödeme id bilgisini içermektedir. Eğer ödeme yöntemi, kapıda ödeme yöntemlerinden biri ise, ilgili ödeme yönteminin Id bilgisi gelecektir.

- **extendedProperties** (object): Siparişin diğer detaylarını içeren bilgidir.

-- **orderNote** (string): Siparişe ait genel not bilgisini içeren alandır.

-- **saveGreen** (bool): True ise, "Yeşili Korum: Plastik Çatal Bıçak Göndermeyin" anlamına gelmektedir.

-- **contactlessDelivery** (bool): True ise "Temassız Teslimat" anlamına gelmektedir.

-- **ringDoorBell** (bool): False ise "Zili Çalmayınız!!!" anlamına gelmektedir.

- **createdAsMs** (long): İlgili siparişin, hangi tarihte verildiği bilgisini içeren, epoch timestamp formatındaki tarih bilgisidir.

2.2 Sipariş İptal Edildi (Order Canceled Webhook)

Bir sipariş, hangi sebeple olursa olsun, iptal edilmesi-reddedilmesi durumunda, entegratörü bilgilendirmek adına gönderilen webhook isteğidir. İlgili entegratör firmaya ait POST metoduna istek atılmaktadır. İstek atılan metod, basic authentication ile yetkilendirilmelidir. Diğer yetkilendirme yöntemleri, mevcut yapıda işlenememektedir.

Retry yapısı: Sipariş iptal edildi bilgisi iletilirken, 3 kez deneme yapılır. Her deneme arasında sırasıyla 10-20-30 saniye bulunmaktadır. 3. Denemede de işlem başarısız olur ise, tekrar sipariş iptali bilgisi aktarım denemesi gerçekleştirilmez.

Model

```
{
  "OrderId": 0,
  "StoreId": 0,
  "UserId": 0
}
```

- **OrderId** (long): Sipariş Id bilgisidir, Migros Yemek paneli üzerindeki sipariş numarası bilgisi ile aynıdır.
- **StoreId** (long): Siparişin iletildiği restoranın Id bilgisidir.
- **UserId** (long): Siparişi veren müşterinin Id bilgisidir.

2.3 Kurye Durumu Değişti (Delivery Status Changed Webhook)

Migros Yemek kuryesi tarafından iletilecek siparişlerde, kurye durumlarının değişmesi durumunda, restorani bilgilendirme adına gönderilen webhook isteğidir. İlgili entegratör firmaya ait POST metoduna istek atılmaktadır. İstek atılan metod, basic authentication ile yetkilendirilmelidir. Diğer yetkilendirme yöntemleri, mevcut yapıda işlenememektedir.

Retry yapısı: Sipariş iptal edildi bilgisi iletilirken, 3 kez deneme yapılır. Her deneme arasında sırasıyla 10-20-30 saniye bulunmaktadır. 3. Denemede de işlem başarısız olur ise, tekrar kurye durum değişikliği bilgisi aktarım denemesi gerçekleştirilmez.

Model

```
{
  "orderId": 0,
  "storeId": 0,
  "status": "string",
  "deliveryStatus": "string",
  "isCancelled": true,
  "deliveryProvider": "string",
  "courierName": "string"
}
```

- **orderId** (long): Sipariş Id bilgisidir, Migros Yemek paneli üzerindeki sipariş numarası bilgisi ile aynıdır.

- **storeId** (long): Siparişin iletildiği restoranın Id bilgisidir.

- **status** (string): Sipariş durumu bilgisidir.

- **deliveryStatus** (string): Kurye durumu bilgisidir. Şu bilgilerden biri gelebilir

1. ASSIGNED_FOR_DELIVERY: Kurye ataması yapılmıştır.
2. COURIER_APPROACHED: Kurye restorana 1 km yaklaştığında iletilmektedir.
3. COURIER_ARRIVED: Kurye restorana ulaştığında iletilmektedir.
4. IN_DELIVERY: Kurye yolda.
5. DELIVERED: Kurye paketi teslim etti.

- **isCancelled**(bool): Siparişin iptal olup olmadığı bilgisini getirir.

- **deliveryProvider**(string): Siparişi restoranın mu yoksa Migrosun kuryesinin mi götüreceğini belirten alandır.

“RESTAURANT ” ise restoran kendi kuryesi ile götürür. “MIGROS” ise Migros kuryesi teslimatı gerçekleştirecek anlamı taşımaktadır.

- **courierName** (string): Kurye atanmış ise, kuryenin ismini belirten alandır.

3 Mapping

3.1 GetOrderStatusList

Sipariş Durum Tiplerini getiren servis. GET Metodu olduğundan ötürü bir şifreleme işlemi söz konusu değildir.

Request

Metod	URL
GET	Mapping/GetOrderStatusList

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key

Response

Status	Response
200	<pre>{ "data": [{ "id": "string", "parameterKey": "string", "parameterValue": "string", "description": "string" }], "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>Data (object array): Sipariş işlemlerinde gönderilecek statü bilgilerini ve açıklamalarını getirmektedir. (Approved,Rejected,Prepared,Delivery,Completed)</p> <p>-id (string)</p> <p>-parameterKey (string):UpdateOrderStatus servisine istek atılırken kullanılacak "orderStatus" parametresine denk gelmektedir.</p> <p>-description (string): İlgili status bilgisinin açıklamasını içermektedir.</p>

3.2 V2/GetCancelReasons

Sipariş İptal Sebeplerini getiren servis

Request

Metod	URL
GET	Mapping/v2/GetCancelReasons

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key

Response

Status	Response
200	<pre>{ "data": [{ "reasonId": 0 "description": "string" }], "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>Data (object array): İptal sebeplerini içeren listedir.</p> <p>-reasonId (long) : V2/UpdateOrderStatus ve V2/CancelOrder metodlarında cancelReasonId alanına karşılık gelen alan.</p> <p>-description (string): İlgili iptal sebebinin açıklamasını içermektedir.</p>

3.3 GetCancelReasons

5 Mart 2024 tarihi itibariyle kullanım dışı olmuştur ve kaldırılacaktır. Bu servis yerine Mapping/v2/GetCancelReasons servisi kullanılmalıdır.

Sipariş İptal Sebeplerini getiren servis

Request

Metod	URL
GET	Mapping/GetCancelReasons

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key

Response

Status	Response
200	<pre>{ "data": [{ "name": "string", "description": "string" }], "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>Data (object array): İptal sebeplerini içeren listedir.</p> <p>-name (string) : UpdateOrderStatus ve CancelOrder metodlarında gönderilecek alan.</p> <p>-description (string): İlgili iptal sebebinin açıklamasını içermektedir.</p>

4 Menu

4.1 MenuByStoreGroupId

Gönderilen Zincir Id (StoreGroupId) bilgisine göre, ilgili zincire tanımlanmış tüm menü isimlerini ve id lerini getirir.

Request

Metod	URL
POST	Menu/MenuByStoreGroupId

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0 }</pre>	StoreGroupId: Zincir Id bilgisidir. Long tipindedir.

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "storeId": 0, "name": "string" }], "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>Data (object array): İlgili zincire tanımlanmış menülerin id ve isim bilgilerini içeren listedir.</p> <p>-id (long): İlgili menüye ait id bilgisidir</p> <p>-name (string) : Menünün ismidir.</p>

4.2 MenuDetails

Menü içerisindeki tüm menü başlıkları ve başlıklar altındaki ürünleri getirmektedir. Opsiyonlar ve çıkarılabilir malzemeler, bu servis içerisinde gelmemektedir.

Request

Metod	URL
POST	Menu/MenuDetails

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "id": 0, "storeGroupId": 0 }</pre>	Id: Menü'ye ait id bilgisidir. Long tipindedir. StoreGroupId: Zincir Id bilgisidir.Long tipindedir.

Response

Status	Response
200	<pre>{ "data": { "id": 0, "storeGroupId": 0, "totalProductCount": 0, "productCountWithImage": 0, "name": "string", "status": "string", "menuHeaderInfos": [{ "id": 0, "name": "string", "status": "string", "foodMenuItemDetailsDTOs": [{ "id": 0, "productId": 0, "productName": "string", "productDescription": "string", "imageUrls": [{ "urls": { "PRODUCT_LIST": "string", "PRODUCT_DETAIL": "string", "PRODUCT_HD": "string", "CART": "string" }, "productId": 0 }], "defaultPrimaryPrice": 0, "defaultPrimaryDiscountedPrice": 0, "status": "string" }] }] }, "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>Data: (object): Menüye ait detay bilgilerini içeren modeldir.</p> <ul style="list-style-type: none">- id (long): İlgili menüye ait id bilgisidir- storeGroupId (long) : Zincir Id bilgisidir.- totalProductCount (int): Mevcut menüdeki ürün sayısını belirtir.- productCountWithImage (int): Resmi olan ürün sayısını belirtir.- name (string): Menü ismini belirtir.- status (string): Menünün aktiflik durumunu belirtir.- menuHeaderInfos (object array): Menü içerisindeki menü başlıkları ile ilgili detay bilgileri barındıran entity'dir-- id (long): Menü başlığı Id bilgisidir.-- name (string): Menü başlığı adını belirtir.-- status (string): Menü başlığının aktiflik durumunu belirtir.

	<p>-- foodMenuItemDetailsDTOs (object array): İlgili menü başlığı altındaki ürünleri listeler.</p> <p>--- id (long): Menü başlığı-ürün eşleştirmesi ile oluşan ara tablo id bilgisidir.</p> <p>--- productId (long): Ürün Id bilgisidir.</p> <p>--- productName (string): Ürünün adı bilgisini içerir.</p> <p>--- productDescription (string): Ürünün açıklamasını içerir.</p> <p>--- defaultPrimaryPrice (long): Ürünün, kuruş cinsinden, üstü çizili fiyatıdır. 100.00 TL Olarak görüntülenmektedir.</p> <p>--- defaultPrimaryDiscountedPrice (long): Ürünün satış fiyatıdır. Üstü çizili fiyatı le aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.</p> <p>--- status (string): Ürünün -zincir genelinde- aktiflik durumunu bildirir.</p>
--	---

4.3 GetMenuItems

İlgili menü içerisindeki ürün listesini, başlık bilgileri olmadan listeler. Opsiyonlar ve çıkarılabilir malzemeler, bu servis içerisinde gelmemektedir.

Request

Metod	URL
POST	Menu/GetMenuItems

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "menuId": 0}</pre>	menuId: Menü'ye ait id bilgisidir. Long tipindedir.

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "menuId": 0, "productId": 0, "productName": "string", "defaultPrimaryPrice": 0, "defaultPrimaryDiscountedPrice": 0, "status": "string" }], "success": true, "errorMessage": null, "validationErrorMessages": null}</pre> <p>Data (object array): İlgili menü içerisinde yer alan productların listesidir.</p> <p>-id (long): Menü başlığı-ürün bağlantısını gerçekleştiren ara tabloya ait id bilgisidir.</p> <p>-menuId (long): Menü'nün Id bilgisidir.</p> <p>-productId (long) : Ürünün Id bilgisidir.</p> <p>-productName (string): Ürün adı bilgisini içerir.</p> <p>- defaultPrimaryPrice (long): Ürünün, kuruş cinsinden, üstü çizili fiyatıdır. 100.00 TL Olarak görüntülenmektedir.</p> <p>- defaultPrimaryDiscountedPrice (long): Ürünün satış fiyatıdır. Üstü çizili fiyatı le aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.</p> <p>- status (string): Ürünün -zincir genelinde- aktiflik durumunu bildirir.</p>

4.4 GetStoreMenusByStoreIds

Gönderilen Şube Id bilgisi ile, ilgili şubede aktif olarak kullanılan menü bilgilerini getiren servis.

Request

Metod	URL
POST	/Menu/GetStoreMenusByStoreIds

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeIds": [0]}</pre>	<p>storeIds: Şube Id bilgisini içerecek long tipindeki array bilgisidir.</p> <p>Örn: "storeId":[23640001231]</p>

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "storeId": 0, "menuId": 0, "storeGroupId": 0 }], "success": true, "errorMessage": null, "validationErrorMessages": null}</pre> <p>Data (object array): İlgili storeId ye atanmış menü'ye ait bilgileri içerir.</p> <p>-menuId (long): İlgili şubede aktif olarak kullanılan Menü'nün Id bilgisidir.</p> <p>-storeId (long) : Şubein Id bilgisidir.</p> <p>-storeGroupId (long): İlgili şubenin bağlı olduğu zincir tanımına ait id bilgisidir.</p>

4.5 GetAllOptionsByMenuId

Bir menü içerisindeki tüm ürünlere ait opsiyon bilgilerini getiren servistir.

Request

Metod	URL
POST	/Menu/GetAllOptionsByMenuId

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "menuId": 0 }</pre>	menuId: Menü'ye ait id bilgisidir. Long tipindedir.

Response

Status	Response
200	<pre>{ "data": [{ "ownerId": 0, "optionsInfo": { "objectOptionHeaderInfosV2": [{ "id": 0, "optionHeaderDTO": { "id": 0, "menuId": 0, "name": "string", "description": "string", "isInclude": true, "status": "string" }, "ownerId": 0, "minItemChoiceQuantity": 0, "maxItemChoiceQuantity": 0, "isMultipleChoice": true, "objectOptionItemInfosV2": [{ "id": 0, "optionItemDTO": { "id": 0, "name": "string", "description": "string", "objectId": 0, "objectType": "string", "status": "string" }, "primaryPrice": 0, "isExcludable": true, "status": "string" }], "status": "string" }] } }], "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>Data (object array): Menüde opsiyonu bulunan tüm ürünlere ait opsiyon ve opsiyon başlığı bilgilerini içeren listedir.</p> <p>- ownerId (long): Bu katmandaki ownerId bilgisi, aslında diğer servislerdeki ProductId bilgisine denk gelmektedir. Burada Owner olarak isimlendirilmesinin sebebi, bu modelin altındaki opsiyonların, bu ürüne aitliğini belirtmek içindir.</p> <p>- optionsInfo (object): OwnerId'ye ait opsiyon başlığı ve opsiyon listesi bilgilerini barındıran çatı entity bilgisidir.</p> <p>-- objectOptionHeaderInfosV2 (object array): İlgili OwnerId'de tanımlı opsiyon başlıklarının listesini barındırmaktadır.</p> <p>--- id (long) : Opsiyon başlığı-ürün arasındaki bağlantının sağlandığı ara tablo id bilgisidir.</p>

--- optionHeaderDTO (object) : İlgili opsiyon başlığına ait bilgileri içermektedir.

--- optionHeaderDto.id (long): Opsiyon başlığına ait id bilgisidir.

--- optionHeaderDto.name (string): Opsiyon başlığı adı bilgisini içerir.

--- optionHeaderDto.description (string): Opsiyon başlığı açıklaması bilgisini içerir.

--- optionHeaderDto.isInclude (bool): Menü içerisindeki ürünleri temsil eden opsiyon başlığı bilgisidir. True ise, ürünün içerisindeki “Malzemeler”i belirten başlıktır, false ise haricen tanımlanan başlıktır.

--- status (string): Opsiyon başlığına ait aktiflik bilgisini içermektedir.

-- ownerId (long): İlk baştaki ownerId bilgisi ile aynıdır.

-- minItemChoiceQuantity (long): İlgili opsiyon başlığı altında tanımlanan opsiyonlardan, tek seferde minimum kaç adet alabileceği bilgisini içerir. 0 olması durumunda, kullanıcıyı opsiyon tercih etmeme hakkına sahip olur, 1 olması durumunda en az 1 adet opsiyon seçimi yapması beklenmektedir.

-- maxItemChoiceQuantity (long): İlgili opsiyon başlığı altında tanımlanan opsiyonlardan, tek seferde maksimum kaç adet alabileceği bilgisini içerir.

-- isMultipleChoice (bool): İlgili opsiyon başlığı altında, kullanıcı tek seferde birden fazla tercih yapabilip yapamayacağının bilgisidir.

-- objectOptionItemInfosV2 (object array): Opsiyon başlığı altında tanımlanmış opsiyonların listesini tutmaktadır.

--- id (long): İlgili opsiyonun, owner product ile bağlantısını sağlayan ara tabloya ait id bilgisidir.

--- optionItemDTO (object): Opsiyon bilgisini barındıran modeldir.

--- optionItemDTO.id (long): Opsiyona ait id bilgisidir.

--- optionItemDTO.name (string): Opsiyonun adını içeren bilgidir.

--- optionItemDTO.description (string): Opsiyonun açıklamasını içeren bilgidir.

--- optionItemDTO. objectId (long): Eğer ilgili opsiyon, aslında başka bir ürüne ait tanım ise, buradaki objectId, diğer ürüne ait productId bilgisini içermektedir.

--- optionItemDTO.objectType (string): İlgili opsiyonun, Ürün mü, İçerisindeki Malzeme mi yoksa diğer bir opsiyon mu olup olmadığını belirten bilgidir.
Alabileceği değerler: PRODUCT, INGREDIENT, NONE

--- optionItemDTO.status (string): İlgili opsiyonun aktiflik bilgisini içerir.

--- primaryPrice (long): Kuruş cinsinden ilgili opsiyonun fiyat bilgisidir.

--- isExcludable (bool): İlgili opsiyonun, çıkarılabilir bir malzeme olup olmadığı bilgisini içerir. True ise çıkarılabilir malzeme, false ise diğer bir opsiyon türü demektir.

4.6 UpdateProductStatusByStoreId

Bir ürünün aktifliğini, ilgili şube için düzenlemek için kullanılacak servistir.

Request

Metod	URL
POST	/Menu/UpdateProductStatusByStoreId

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "productId": 0, "status": "string" }</pre>	<p>storeId: Şube'ye ait id bilgisidir. Long tipindedir.</p> <p>productId: Ürüne ait Id bilgisidir. Long tipindedir.</p> <p>status: Ürünün, şubedeki aktiflik değerini düzenleyen değerdir. String tipindedir.</p> <p>Alabileceği değerler</p> <p>->ACTIVE</p> <p>->PASSIVE</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null "validationErrorMessages": null }</pre>

4.7 UpdateBatchPrice

Bir menüdeki ürün fiyatlarını güncellemek için kullanılan servistir. Yalnızca, güncellenmek istenen ürünlerin gönderilmesi yeterlidir.

Request

Metod	URL
POST	/Menu/UpdateBatchPrice

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeGroupId": 0, "menuId": 0, "prices": [{ "menuItemId": 0, "productId": 0, "defaultPrimaryPrice": 0, "defaultPrimaryDiscountedPrice": 0 }] }</pre>	<p>storeGroupId: Zincir/Marka id bilgisidir</p> <p>menuId: İlgili güncelleştirmelerin hangi menüde-babtta yapılacağı bilgisini içerir.</p> <p>Prices: Fiyat listesini içeren listedir</p> <ul style="list-style-type: none">- menuItemId (long): Ürün ile menü arasındaki ara id bilgisidir.- productId (long) ürünün id bilgisidir.- defaultPrimaryPrice (long): Fiyat bilgisidir.- defaultPrimaryDiscountedPrice (long): Ürünün fiyat bilgisidir. <p>NOT: Buradaki her iki fiyat bilgisi de aynı gönderilmelidir.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null "validatonErrorMessages": null }</pre>

4.8 GetMenuDetailsByStoreAndStoreGroupId

İlgili restorandaki menü detayları ve menüde bulunan ürünlerin restorandaki durumlarını getiren servis.

Request

Metod	URL
POST	/Menu/ GetMenuDetailsByStoreAndStoreGroupId

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "storeGroupId": 0 }</pre>	storeId: Restoran Id bilgisidir. storeGroupId: Zincir/Marka id bilgisidir

Response

Status	Response
200	<pre>{ "data": { "id": 0, "storeGroupId": 0, "totalProductCount": 0, "productCountWithImage": 0, "name": "string", "createdAt": 0, "creatorId": 0, "lastUpdatedAt": 0, "lastUpdaterId": 0, "status": "string", "menuHeaderInfos": [{ "id": 0, "menuId": 0, "name": "string", "priority": 0, "createdAt": 0, "creatorId": 0, "lastUpdatedAt": 0, "lastUpdaterId": 0, "status": "string", "foodMenuItemDetailsDTOs": [{ "id": 0, "menuId": 0, "productId": 0, "productName": "string", "productDescription": "string", "imageUrls": [</pre>

```

        "urls": {
            "PRODUCT_LIST": "string",
            "PRODUCT_DETAIL": "string",
            "PRODUCT_HD": "string",
            "CART": "string"
        },
        "productId": 0
    }
],
"priority": 0,
"defaultPrimaryPrice": 0,
"defaultPrimaryDiscountedPrice": 0,
"createdAt": 0,
"creatorId": 0,
"lastUpdatedAt": 0,
"lastUpdaterId": 0,
"status": "string"
}
}
]
},
"success": true,
"errorMessage": {
    "errorCode": "string",
    "errorTitle": "string",
    "errorDetail": "string",
    "isSystemError": true
},
"validationErrorMessages": [
    {
        "field": "string",
        "cause": "string",
        "errorDetail": "string"
    }
]
}

```

Data: (object): Menüye ait detay bilgilerini içeren modeldir.

- id (long): İlgili menüye ait id bilgisidir
- storeGroupld (long) : Zincir Id bilgisidir.
- totalProductCount (int): Mevcut menüdeki ürün sayısını belirtir.
- productCountWithImage (int): Resmi olan ürün sayısını belirtir.
- name (string): Menü ismini belirtir.
- status (string): Menü'nün aktiflik durumunu belirtir.
- menuHeaderInfos (object array): Menü içerisindeki menü başlıkları ile ilgili detay bilgileri barındıran entity'dir
- id (long): Menü başlığı Id bilgisidir.
- name (string): Menü başlığı adını belirtir.
- status (string): Menü başlığının aktiflik durumunu belirtir.
- foodMenuitemDetailsDTOs (object array): İlgili menü başlığı altındaki ürünleri listeler.
- id (long): Menü başlığı-ürün eşleştirmesi ile oluşan ara tablo id bilgisidir.
- productId (long): Ürün Id bilgisidir.

	<p>--- productName (string): Ürünün adı bilgisini içerir.</p> <p>--- productDescription (string): Ürünün açıklamasını içerir.</p> <p>--- defaultPrimaryPrice (long): Ürünün, kuruş cinsinden, üstü çizili fiyatıdır. 100.00 TL Olarak görüntülenmektedir.</p> <p>--- defaultPrimaryDiscountedPrice (long): Ürünün satış fiyatıdır. Üstü çizili fiyatı le aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.</p> <p>--- status (string): Ürünün ilgili şubedeki satışa açık olup olmadığı bilgisini içerir.</p>
--	---

5 Order

5.1 PendingOrdersWithStores

Onay bekleyen siparişleri sorgulamak için kullanılır.

Request

Metod	URL
POST	/Order/PendingOrdersWithStores

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeIds": [0], "offset": 0, "limit": 0 }</pre>	<p>storeIds: Sorgulanmak istenilen şubeye ait Id bilgisinin iletileceği alandır. Long Array tipindedir.Örnek Kullanım: "storeIds":[293110003201]</p> <p>limit: Maksimum kaç adet sipariş sorgulanacağı bilgisini içeren alandır. Long tipindedir. Minimum değer=1, maksimum değer=20 dir.</p> <p>Offset: Belirtilen sayı kadarki siparişleri pas geçerek, bir sonraki limit adedindeki siparişlerin sorgulanmasını sağlamaktadır, sayfalama içindir.</p> <p>İlk 20 siparişi sorgulama örneği</p> <pre>"storeIds": [293110003201], "offset": 0, "limit": 20 }</pre> <p>İkinci 20 siparişi sorgulama örneği</p> <pre>"storeIds": [293110003201], "offset": 20, "limit": 20 }</pre>

Response

Status	Response
200	<pre>{ "data": [{ "storeId": 0, "pendingOrderDetailsDTOS": [{ "id": 0, "userId": 0, "storeName": "string", "storeGroupName": "string", "phoneNumber": "string", "description": "string", "totalPrice": 0, "totalPriceText": "string", "discountedPrice": 0, "discountedPriceText": "string", "restaurantDiscountedPrice": 0, "restaurantDiscountedPriceText": "string", "migrosDiscountedPrice": 0, "migrosDiscountedPriceText": "string", "products": [{ "id": 0, "productId": 0, "name": "string", "price": 0, "priceText": "string", "amount": 0, "note": "string", "options": [{ "headerName": "string", "itemNames": "string", "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [{ "headerName": "string", "itemNames": "string", "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [] }] }] }] }] }], "contactlessDelivery": true, "ringDoorbell": true, }</pre>

```

        "saveGreen": true,
        "orderNote": "string",
        "customerFullName": "string",
        "address": "string",
        "addressDirection": "string",
        "status": "string",
        "paymentType": "string",
        "paymentTypeDescription": "string",
        "orderApproval": {
            "createdAt": 0,
            "expireAt": 0
        },
        "deliveryLocation": {
            "latitude": 0,
            "longitude": 0
        },
        "deliveryProvider": "string",
        "courierName": "string"
    }
}
],
"success": true,
"errorMessage": null
"validatonErrorMessages": null
}

```

Data (object array): Onay bekleyen siparişler ve siparişe ait storeld bilgilerini içeren liste bilgisidir.

- storeld (long): İlgili modeldeki siparişlerin sahip olduğu şube bilgisine ait id bilgisidir.

- pendingOrderDetailsDTOS (object array): Siparişleri içeren liste bilgisidir.

-- id (long): Siparişin id bilgisidir. Aynı zamanda sipariş numarası olarak kullanılmaktadır.

-- userId (long): Siparişi veren kullanıcının id bilgisidir. Ters akış (webhook) için oluşturulan modelde bu bilgi Customer modelinin içerisindeki id bilgisine denk gelmektedir.

-- storeName (string): Şiparişin geldiği şube adı bilgisidir.

-- storeGroupName (string): Siparişin geldiği şubenin zincir adı bilgisidir.

-- phoneNumber (string): Siparişi veren müşteriye ait telefon bilgisidir.

-- description (string): Sipariş içeriğine ait özet başlık bilgisini barındırır.

-- totalPrice (long): Siparişin kuruş cinsinden toplam fiyat bilgisidir.

-- totalPriceText (string): Siparişin TL cinsinden toplam fiyat bilgisidir.

-- discountedPrice (long): Siparişin, indirimler sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.

-- discountedPriceText (string): Siparişin, indirimler sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.

-- restaurantDiscountedPrice (long): Siparişin, Restoranın yaptığı indirim sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.

-- restaurantDiscountedPriceText (string): Siparişin, Restoranın yaptığı indirim sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.

-- migrosDiscountedPrice (long): Siparişin, Migros Yemek indrimi sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.

-- migrosDiscountedPriceText (string): Siparişin, Migros Yemek indrimi sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.

-- products (array object): Sipariş içerisindeki ürünlerin listesidir.

--- id (long): Sipariş bilgisi ile ürün bilgisinin bağlantısını sağlayan aratablonun id bilgisidir.

--- productId (long): İlgili ürünün id bilgisidir.

--- name (string): Ürün adı bilgisidir.

--- price (long): Ürünün kuruş cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

--- priceText (string): Ürünün TL cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

--- amount (int): İlgili ürünün adet miktarı bilgisidir.

--- note (string): İlgili ürüne özel not bilgisidir.

--- options (object array): İlgili ürünün alt opsiyonlarını, çıkarılabilir malzemeleri ve ek ürün gibi özelliklerini içeren liste bilgisidir.

---- headerName (string): İlgili opsiyonun, opsiyon başlığına ait bilgisidir.

---- itemNames (string): Opsiyonun adı

---- objectOptionItemId (long): İlgili opsiyonun, ilgili opsiyon başlığı ile ilişkisinin sağlandığı ara tablo id bilgisidir.

---- optionItemId (long): Opsiyona ait id bilgisidir. Bu id, bir menüdeki aynı isimdeki diğer opsiyonlar ile aynı id bilgisine sahip olmaktadır. Örneğin, Ketçap isimli bir opsiyon olduğunu farzederseniz, bu opsiyonun n tane ürüne tanımlanması farketmeksizin, buradaki id bilgisi ortak olacaktır.

---- productId (long): İlgili opsiyon, menü içerisinde tanımlanmış bir ürün ise, product id bilgisi de gelmektedir. ProductId bilgisi, ürün id bilgisidir.

---- primaryDefaultPrice (long): Opsiyonun, kuruş cinsinden, üstü çizili fiyatıdır. ~~100.00~~ TL Olarak görüntülenmektedir.

---- primaryDefaultDiscountedPrice (long): Opsiyonun kuruş cinsinden satış fiyatıdır. Üstü çizili fiyatı ile aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.

---- primaryPriceText (string): Opsiyonun TL cinsinden fiyatıdır. Quantity alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, quantity alanına bölünmesi gerekmektedir.

---- quantity (int): Opsiyonun adet miktarıdır.

---- excluded (bool): Çıkarılacak bir opsiyon olup olmadığı bilgisidir. Örneğin, bu değer true ise, ilgili opsiyonun çıkarılması için gönderilen bir bilgi olduğunu belirtir. False ise, çıkarılacak bir malzeme değil, diğer türlerde bir opsiyon bilgisidir.

---- optionType (string): İlgili opsiyonun tip bilgisini vermektedir. "PRODUCT" gelirse, ilgili opsiyon menüde tanımlanmış başka bir ürünü temsil eder. "INGREDIENT" gelirse, çıkarılacak bir malzeme bilgisi temsil eder. "NONE" gelirse, diğer tiplerdeki tüm opsiyonları temsil eder (ek malzeme/ürün, promosyon seçimi, porsiyon seçimi vb.)

---- subOptions (object array): İlgili opsiyonun alt opsiyonlarını barındıran listedir. Options modeli ile tipi aynıdır. En alt katman için bir sınır bulunmamaktadır.

-- contactlessDelivery (bool): True ise "Temassız Teslimat İstiyorum" bilgisini getirmektedir.

-- ringDoorbell (bool): False ise, "Lütfen Zili Çalmayınız!" ifadesini temsil eden bilgidir.

-- saveGreen (string): True ise "Plastik çatal bıçak istemiyorum!" ifadesini temsil eden bilgidir.

-- orderNote (string): Kullanıcının sipariş özelindeki not bilgisini içerir.

-- customerFullName (string): Müşterinin ad-soyad bilgisini içerir. Ad-soyad zorunluluğu bulunmadığı için bu bilgi boş gelebilmektedir.

-- address (string): Siparişin teslim edileceği adres bilgisidir.

-- addressDirection (string): Adres tarifi bilgisidir.

-- status (string): Siparişin durumunu içeren bilgidir.

-- paymentType (string): Siparişin hangi ödeme yöntemi ile alındığının bilgisidir.

-- paymentTypeDescription (string): İlgili ödeme yönteminin açıklamasıdır.

-- orderApproval (object): siparişin hangi tarihte oluşturulduğunu ve onaylanmaması durumunda hangi tarih ve saatte otomatik iptal edileceği bilgisini içeren epoch timestamp (long) formatındaki bilgileri içeren modeldir.

-- deliveryLocation (object): Siparişin gideceği adresin konum bilgilerini içeren modeldir.

--- latitude (long): Konumun enlem bilgisidir.

--- longitude (long): Konumun boylam bilgisidir.

-- deliveryProvider (string): Teslimatın kim tarafından yapılacağına dair bilgi içerir. "**RESTAURANT**" ise teslimattan restoran kuryesi sorumludur. "**MIGROS**" olarak belirtilmiş ise, teslimat Migros Yemek Teslimatı kuryeleri tarafından yapılacaktır.

-- courierName (string): Eğer deliveryProvider alanı "**MIGROS**" gelmiş ise, yani siparişin teslimatından sorumlu kurye Migros Yemek Teslimatı kuryesi ise, kurye atanması durumunda bu alanda ilgili kuryenin ad ve soyad bilgisi gösterilecektir. Teslimatın, restoran sorumluluğunda olduğu durumlarda bu alan null yada boş string dönmektedir.

5.2 ActiveOrdersWithStores

Onaylanmış ve henüz teslim aşamasına getirilmemiş siparişlerin listelemesinde kullanılır.

Request

Metod	URL
POST	/Order/ActiveOrdersWithStores

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeIds": [0], "offset": 0, "limit": 0 }</pre>	<p>storeIds: Sorgulanmak istenilen şubeye ait Id bilgisinin iletileceği alandır. Long Array tipindedir.Örnek Kullanım: "storeIds":[293110003201]</p> <p>limit: Maksimum kaç adet sipariş sorgulanacağı bilgisini içeren alandır. Long tipindedir. Minimum değer=1, maksimum değer=20 dir.</p> <p>Offset: Belirtilen sayı kadarki siparişleri pas geçerek, bir sonraki limit adedindeki siparişlerin sorgulanmasını sağlamaktadır, sayfalama içindir.</p> <p>İlk 20 siparişi sorgulama örneği</p> <pre>{ "storeIds": [293110003201], "offset": 0, "limit": 20 }</pre> <p>İkinci 20 siparişi sorgulama örneği</p> <pre>{ "storeIds": [293110003201], "offset": 20, "limit": 20 }</pre>

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeIds": [0], "offset": 0, "limit": 0 }</pre>	<p>storeIds: Sorgulanmak istenilen şubeye ait Id bilgisinin iletileceği alandır. Long Array tipindedir.Örnek Kullanım: “storeIds”:[293110003201]</p> <p>limit: Maksimum kaç adet sipariş sorgulanacağı bilgisini içeren alandır. Long tipindedir. Minimum değer=1, maksimum değer=20 dir.</p> <p>Offset: Belirtilen sayı kadarki siparişleri pas geçerek, bir sonraki limit adedindeki siparişlerin sorgulanmasını sağlamaktadır, sayfalama içindir.</p> <p>İlk 20 siparişi sorgulama örneği</p> <pre>"storeIds": [293110003201], "offset": 0, "limit": 20 }</pre> <p>İkinci 20 siparişi sorgulama örneği</p> <pre>"storeIds": [293110003201], "offset": 20, "limit": 20 }</pre>

Response

Status	Response
200	<pre>{ "data": [{ "activeOrderDetailsDTOS": [{ "id": 0, "userId": 0, "storeName": "string", "storeGroupName": "string", "phoneNumber": "string", "description": "string", "totalPrice": 0, "totalPriceText": "string", "discountedPrice": 0, "discountedPriceText": "string", "restaurantDiscountedPrice": 0, "restaurantDiscountedPriceText": "string", "migrosDiscountedPrice": 0, "migrosDiscountedPriceText": "string", "products": [{ "id": 0, "productId": 0, "name": "string", "price": 0, "priceText": "string", "amount": 0, "note": "string", "options": [{ "headerName": "string", "itemNames": "string", "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [{ "headerName": "string", "itemNames": "string", "objectOptionHeaderPriority": 0, "objectOptionItemPriority": 0, "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [] }] }] }] }] }], "contactlessDelivery": true, }</pre>


```

        "ringDoorbell": true,
        "saveGreen": true,
        "orderNote": "string",
        "customerFullName": "string",
        "address": "string",
        "addressDirection": "string",
        "createdAt": 0,
        "status": "string",
        "paymentType": "string",
        "paymentTypeDescription": "string",
        "deliveryLocation": {
            "latitude": 0,
            "longitude": 0
        },
        "deliveryProvider": "string",
        "courierName": "string"
    }
}
],
"success": true,
"errorMessage": null
"validatonErrorMessages": null
}

```

Data (object array): Onaylanmış ve henüz teslim edilmemiş siparişleri storeld bilgileri ile birlikte içeren liste.

- storeld (long): İlgili modelindeki siparişlerin sahip olduğu şube bilgisine ait id bilgisidir.

- activeOrderDetailsDTOS (object array): Onaylanmış ve henüz teslim edilmemiş, siparişleri içeren liste bilgisidir.

-- id (long): Siparişin id bilgisidir. Aynı zamanda sipariş numarası olarak kullanılmaktadır.

-- userId (long): Siparişi veren kullanıcının id bilgisidir. Ters akış (webhook) için oluşturulan modelde bu bilgi Customer modelinin içerisindeki id bilgisine denk gelmektedir.

-- storeName (string): Şiparişin geldiği şube adı bilgisidir.

-- storeGroupName (string): Siparişin geldiği şubenin zincir adı bilgisidir.

-- phoneNumber (string): Siparişi veren müşteriye ait telefon bilgisidir.

-- description (string): Sipariş içeriğine ait özet başlık bilgisini barındırır.

-- totalPrice (long): Siparişin kuruş cinsinden toplam fiyat bilgisidir.

-- totalPriceText (string): Siparişin TL cinsinden toplam fiyat bilgisidir.

-- discountedPrice (long): Siparişin, indirimler sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.

-- discountedPriceText (string): Siparişin, indirimler sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.

-- restaurantDiscountedPrice (long): Siparişin, Restoranın yaptığı indirim sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.

-- restaurantDiscountedPriceText (string): Siparişin, Restoranın yaptığı indirim sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.

-- migrosDiscountedPrice (long): Siparişin, Migros Yemek indirim sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.

-- migrosDiscountedPriceText (string): Siparişin, Migros Yemek indirimini sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.

-- products (array object): Sipariş içerisindeki ürünlerin listesidir.

--- id (long): Sipariş bilgisi ile ürün bilgisinin bağlantısını sağlayan aratablonun id bilgisidir.

--- productId (long): İlgili ürünün id bilgisidir.

--- name (string): Ürün adı bilgisidir.

--- price (long): Ürünün kuruş cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

--- priceText (string): Ürünün TL cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

--- amount (int): İlgili ürünün adet miktarı bilgisidir.

--- note (string): İlgili ürüne özel not bilgisidir.

--- options (object array): İlgili ürünün alt opsiyonlarını, çıkarılabilir malzemeleri ve ek ürün gibi özelliklerini içeren liste bilgisidir.

---- headerName (string): İlgili opsiyonun, opsiyon başlığına ait bilgidir.

---- itemNames (string): Opsiyonun adı

---- objectOptionItemId (long): İlgili opsiyonun, ilgili opsiyon başlığı ile ilişkisinin sağlandığı ara tablo id bilgisidir.

---- optionItemId (long): Opsiyona ait id bilgisidir. Bu id, bir menüdeki aynı isimdeki diğer opsiyonlar ile aynı id bilgisine sahip olmaktadır. Örneğin, Ketçap isimli bir opsiyon olduğunu farzederseniz, bu opsiyonun n tane ürüne tanımlanması farketmeksizin, buradaki id bilgisi ortak olacaktır.

---- productId (long): İlgili opsiyon, menü içerisinde tanımlanmış bir ürün ise, product id bilgisi de gelmektedir. ProductId bilgisi, ürün id bilgisidir.

---- primaryDefaultPrice (long): Opsiyonun, kuruş cinsinden, üstü çizili fiyatıdır. ~~100.00 TL~~ Olarak görüntülenmektedir.

---- primaryDefaultDiscountedPrice (long): Opsiyonun kuruş cinsinden satış fiyatıdır. Üstü çizili fiyatı ile aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.

---- primaryPriceText (string): Opsiyonun TL cinsinden fiyatıdır. Quantity alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, quantity alanına bölünmesi gerekmektedir.

---- quantity (int): Opsiyonun adet miktarıdır.

---- excluded (bool): Çıkarılacak bir opsiyon olup olmadığı bilgisidir. Örneğin, bu değer true ise, ilgili opsiyonun çıkarılması için gönderilen bir bilgi olduğunu belirtir. False ise, çıkarılacak bir malzeme değil, diğer türlerde bir opsiyon bilgisidir.

---- optionType (string): İlgili opsiyonun tip bilgisini vermektedir. "PRODUCT" gelirse, ilgili opsiyon menüde tanımlanmış başka bir ürünü temsil eder. "INGREDIENT" gelirse, çıkarılacak bir malzeme bilgisi temsil eder. "NONE" gelirse, diğer tiplerdeki tüm opsiyonları temsil eder (ek malzeme/ürün, promosyon seçimi, porsiyon seçimi vb.)

---- subOptions (object array): İlgili opsiyonun alt opsiyonlarını barındıran listedir. Options modeli ile tipi aynıdır. En alt katman için bir sınır bulunmamaktadır.

-- contactlessDelivery (bool): True ise “Temassız Teslimat İstiyorum” bilgisini getirmektedir.

-- ringDoorbell (bool): False ise, “Lütfen Zili Çalmayınız!” ifadesini temsil eden bilgidir.

-- saveGreen (string): True ise “Plastik çatal bıçak istemiyorum!” ifadesini temsil eden bilgidir.

-- orderNote (string): Kullanıcının sipariş özelindeki not bilgisini içerir.

-- customerFullName (string): Müşterinin ad-soyad bilgisini içerir. Ad-soyad zorunluluğu bulunmadığı için bu bilgi boş gelebilmektedir.

-- address (string): Siparişin teslim edileceği adres bilgisidir.

-- addressDirection (string): Adres tarifi bilgisidir.

-- createdAt (long): siparişin hangi tarihte verildiği bilgisini içeren epoch timestamp tipindeki tarih bilgisidir.

-- status (string): Siparişin durumunu içeren bilgidir.

-- paymentType (string): Siparişin hangi ödeme yöntemi ile alındığının bilgisidir.

-- paymentTypeDescription (string): İlgili ödeme yönteminin açıklamasıdır.

-- deliveryLocation (object): Siparişin gideceği adresin konum bilgilerini içeren modeldir.

--- latitude (long): Konumun enlem bilgisidir.

--- longitude (long): Konumun boylam bilgisidir.

-- deliveryProvider (string): Teslimatın kim tarafından yapılacağına dair bilgi içerir. “**RESTAURANT**” ise teslimattan restoran kuryesi sorumludur. “**MIGROS**” olarak belirtilmiş ise, teslimat Migros Yemek Teslimatı kuryeleri tarafından yapılacaktır.

-- courierName (string): Eğer deliveryProvider alanı “**MIGROS**” gelmiş ise, yani siparişin teslimatından sorumlu kurye Migros Yemek Teslimatı kuryesi ise, kurye atanması durumunda bu alanda ilgili kuryenin ad ve soyad bilgisi gösterilecektir. Teslimatın, restoran sorumluluğunda olduğu durumlarda bu alan null yada boş string dönmektedir.

5.3 GetPassiveOrders

Bir şubeye ait geçmiş siparişleri getiren servis.

Request

Metod	URL
POST	/Order/GetPassiveOrders

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "offset": 0, "limit": 0 }</pre>	<p>storeId: Sorgulanmak istenilen şubeye ait Id bilgisinin iletileceği alandır. Long tipindedir.</p> <p>limit: Maksimum kaç adet sipariş sorgulanacağı bilgisini içeren alandır. Long tipindedir. Minimum değer=1, maksimum değer=20 dir.</p> <p>Offset: Belirtilen sayı kadarki siparişleri pas geçerek, bir sonraki limit adedindeki siparişlerin sorgulanmasını sağlamaktadır, sayfalama içindir.</p> <p>İlk 20 siparişi sorgulama örneği</p> <pre>"storeId": 293110003201, "offset": 0, "limit": 20 }</pre> <p>İkinci 20 siparişi sorgulama örneği</p> <pre>"storeId": 293110003201, "offset": 20, "limit": 20 }</pre>

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "userId": 0, "phoneNumber": "string", "description": "string", "totalPrice": 0, "totalPriceText": "string", "discountedPrice": 0, "discountedPriceText": "string", "restaurantDiscountedPrice": 0, "restaurantDiscountedPriceText": "string", "migrosDiscountedPrice": 0, "migrosDiscountedPriceText": "string", "products": [{ "id": 0, "productId": 0, "name": "string", "price": 0, "priceText": "string", "amount": 0, "note": "string", "options": [{ "headerName": "string", "itemNames": "string", "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [{ "headerName": "string", "itemNames": "string", "objectOptionItemId": 0, "optionItemId": 0, "productId": 0, "parentObjectOptionItemId": 0, "primaryDiscountedPrice": 0, "primaryPrice": 0, "primaryDiscountedPriceText": "string", "primaryPriceText": "string", "quantity": 0, "excluded": true, "optionType": "string", "subOptions": [] }] }] }] }], "contactlessDelivery": true, "saveGreen": true, "orderNote": "string", "customerFullName": "string", "address": "string", "addressDirection": "string", "createdAt": 0, }</pre>

```
{
  "status": "string",
  "cancelled": true,
  "cancelReason": "string",
  "ringDoorbell": true,
  "orderApproval": {
    "createdAt": 0,
    "expireAt": 0
  },
  "paymentType": "string",
  "paymentTypeDescription": "string",
  "deliveryLocation": {
    "latitude": 0,
    "longitude": 0
  },
  "deliveryProvider": "string",
  "courierName": "string"
}
],
"success": true,
"errorMessage": null
"validationErrorMessages": null
}
```

Data (object array): İlgili şubeye ait geçmiş sipariş listesini içerir.

- id (long): Siparişin id bilgisidir. Aynı zamanda sipariş numarası olarak kullanılmaktadır.
- userId (long): Siparişi veren kullanıcının id bilgisidir. Ters akış (webhook) için oluşturulan modelde bu bilgi Customer modelinin içerisindeki id bilgisine denk gelmektedir.
- storeName (string): Şiparişin geldiği şube adı bilgisidir.
- storeGroupName (string): Siparişin geldiği şubenin zincir adı bilgisidir.
- phoneNumber (string): Siparişi veren müşteriye ait telefon bilgisidir.
- description (string): Sipariş içeriğine ait özet başlık bilgisini barındırır.
- totalPrice (long): Siparişin kuruş cinsinden toplam fiyat bilgisidir.
- totalPriceText (string): Siparişin TL cinsinden toplam fiyat bilgisidir.
- discountedPrice (long): Siparişin, indirimler sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.
- discountedPriceText (string): Siparişin, indirimler sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. İndirimlerden sonraki nihai halidir.
- restaurantDiscountedPrice (long): Siparişin, Restoranın yaptığı indirim sonrasındaki kuruş cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.
- restaurantDiscountedPriceText (string): Siparişin, Restoranın yaptığı indirim sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Restorandır**.
- migrosDiscountedPrice (long): Siparişin, Migros Yemek indrimi sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.
- migrosDiscountedPriceText (string): Siparişin, Migros Yemek indrimi sonrasındaki TL cinsinden toplam indirimli fiyat bilgisidir. Buradaki indirimin sponsoru **Migros Yemektir**.
- products (array object): Sipariş içerisindeki ürünlerin listesidir.
- id (long): Sipariş bilgisi ile ürün bilgisinin bağlantısını sağlayan aratablonun id bilgisidir.
- productId (long): İlgili ürünün id bilgisidir.
- name (string): Ürün adı bilgisidir.

-- price (long): Ürünün kuruş cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

-- priceText (string): Ürünün TL cinsinden fiyat bilgisidir. Amount alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, amount alanına bölünmesi gerekmektedir.

-- amount (int): İlgili ürünün adet miktarı bilgisidir.

-- note (string): İlgili ürüne özel not bilgisidir.

-- options (object array): İlgili ürünün alt opsiyonlarını, çıkarılabilir malzemeleri ve ek ürün gibi özelliklerini içeren liste bilgisidir.

--- headerName (string): İlgili opsiyonun, opsiyon başlığına ait bilgisidir.

--- itemNames (string): Opsiyonun adı

--- objectOptionItemId (long): İlgili opsiyonun, ilgili opsiyon başlığı ile ilişkisinin sağlandığı ara tablo id bilgisidir.

--- optionItemId (long): Opsiyona ait id bilgisidir. Bu id, bir menüdeki aynı isimdeki diğer opsiyonlar ile aynı id bilgisine sahip olmaktadır. Örneğin, Ketçap isimli bir opsiyon olduğunu farzederek, bu opsiyonun n tane ürüne tanımlanması farketmeksizin, buradaki id bilgisi ortak olacaktır.

--- productId (long): İlgili opsiyon, menü içerisinde tanımlanmış bir ürün ise, product id bilgisi de gelmektedir. ProductId bilgisi, ürün id bilgisidir.

--- primaryDefaultPrice (long): Opsiyonun, kuruş cinsinden, üstü çizili fiyatıdır. ~~100.00 TL~~ Olarak görüntülenmektedir.

--- primaryDefaultDiscountedPrice (long): Opsiyonun kuruş cinsinden satış fiyatıdır. Üstü çizili fiyatı le aynı olabilir. İki fiyat aynı tutarı temsil ediyor ise, mobil uygulamada tek fiyat bilgisi gösterilir.

--- primaryPriceText (string): Opsiyonun TL cinsinden fiyatıdır. Quantity alanı ile çarpıldıktan sonra elde edilen miktardır. Birim fiyatna erişilebilmesi için, quantity alanına bölünmesi gerekmektedir.

--- quantity (int): Opsiyonun adet miktarıdır.

--- excluded (bool): Çıkarılacak bir opsiyon olup olmadığı bilgisidir. Örneğin, bu değer true ise, ilgili opsiyonun çıkarılması için gönderilen bir bilgi olduğunu belirtir. False ise, çıkarılacak bir malzeme değil, diğer türlerde bir opsiyon bilgisidir.

--- optionType (string): İlgili opsiyonun tip bilgisini vermektedir. "PRODUCT" gelirse, ilgili opsiyon menüde tanımlanmış başka bir ürünü temsil eder. "INGREDIENT" gelirse, çıkarılacak bir malzeme bilgisi temsil eder. "NONE" gelirse, diğer tiplerdeki tüm opsiyonları temsil eder (ek malzeme/ürün, promosyon seçimi, porsiyon seçimi vb.)

--- subOptions (object array): İlgili opsiyonun alt opsiyonlarını barındıran listedir. Options modeli ile tipi aynıdır. En alt katman için bir sınır bulunmamaktadır.

- contactlessDelivery (bool): True ise "Temassız Teslimat İstiyorum" bilgisini getirmektedir.

- ringDoorbell (bool): False ise, "Lütfen Zili Çalmayınız!" ifadesini temsil eden bilgidir.

- saveGreen (string): True ise "Plastik çatal bıçak istemiyorum!" ifadesini temsil eden bilgidir.

- orderNote (string): Kullanıcının sipariş özelindeki not bilgisini içerir.

- customerFullName (string): Müşterinin ad-soyad bilgisini içerir. Ad-soyad zorunluluğu bulunmadığı için bu bilgi boş gelebilmektedir.
- address (string): Siparişin teslim edileceği adres bilgisidir.
- addressDirection (string): Adres tarifi bilgisidir.
- status (string): Siparişin durumunu içeren bilgisidir.
- paymentType (string): Siparişin hangi ödeme yöntemi ile alındığının bilgisidir.
- paymentTypeDescription (string): İlgili ödeme yönteminin açıklamasıdır.
- cancelled (bool): İlgili siparişin, iptal edilen bir sipariş olup olmadığı bilgisini verir. True ise bu sipariş iptal ya da reddedilmiş bir sipariş olarak gösterilmektedir.
- cancelReason (string): Eğer sipariş iptal edilmiş ise, hangi sebepten iptal edildiğine ait bilgiyi getirir.
- orderApproval (object): siparişin hangi tarihte oluşturulduğunu ve onaylanmaması durumunda hangi tarih ve saatte otomatik iptal edileceği bilgisini içeren epoch timestamp (long) formatındaki bilgileri içeren modeldir.
- deliveryLocation (object): Siparişin gideceği adresin konum bilgilerini içeren modeldir.
- latitude (long): Konumun enlem bilgisidir.
- longitude (long): Konumun boylam bilgisidir.
- deliveryProvider (string): Teslimatın kim tarafından yapılacağına dair bilgi içerir. **"RESTAURANT"** ise teslimattan restoran kuryesi sorumludur. **"MIGROS"** olarak belirtilmiş ise, teslimat Migros Yemek Teslimatı kuryeleri tarafından yapılacaktır.
- courierName (string): Eğer deliveryProvider alanı **"MIGROS"** gelmiş ise, yani siparişin teslimatından sorumlu kurye Migros Yemek Teslimatı kuryesi ise, kurye atanması durumunda bu alanda ilgili kuryenin ad ve soyad bilgisi gösterilecektir. Teslimatın, restoran sorumluluğunda olduğu durumlarda bu alan null yada boş string dönmektedir.

5.4 V2/UpdateOrderStatus

Siparişi Onaylama, Reddetme, Hazırlandı olarak işaretleme, Kuryeye Verildi olarak işaretleme ve Teslim edildi olarak işaretleme amacı ile kullanılan servistir.

Request

Metod	URL
POST	/Order/v2/UpdateOrderStatus

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "orderId": 0, "storeId": 0, "orderStatus": "string", "cancelReasonId": 0 }</pre>	<p>orderId: Siparişin id bilgisidir. Migros Yemek Panelindeki sipariş numarası bilgisi ile aynı bilgisidir.</p> <p>storeId: Siparişi alan şubenin id bilgisidir.</p> <p>orderStatus: Yapılmak istenen işlemin Enum karşılığının kontrol edildiği alandır. Buraya gönderilebilir parametreler şunlardır.</p> <p>Approved -> Onaylama Rejected -> Reddetme Prepared -> Hazırlandı Delivery -> Kuryeye Teslim Edildi (Yola Çıktı) Completed -> Müşteriye Teslim edildi.</p> <p>Siparişin akışı itibarıyla, siparişler ilk önce onaya düşmektedir. Bir sipariş, sipariş oluşturulma tarihinden sonraki 10 dakika içerisinde onaylanmaz ise otomatik iptal edilmektedir.</p> <p>Siparişi onaylamak için yapılacak istek örneği aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Approved" }</pre> <p>Siparişi onaylamak için yapılacak istek örneği Henüz onay aşamasında olan bir siparişi kabul etmemek-reddetmek için kullanılacak istek örneği aşağıdaki gibidir. Reddedilen sipariş, işlem göremez ve geçmiş siparişlere düşer.</p>

		<p>cancelReasonId alanına, sipariş iptal sebeplerini getiren listedeki reasonId alanına karşılık gelen değer gönderilmelidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Rejected", "cancelReasonId": 23 }</pre> <p>Onaylanan bir siparişi, “Hazırlandı” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Prepared" }</pre> <p>“Hazırlandı” durumundaki bir siparişi, “Kuryeye Teslim Edildi” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Delivery" }</pre> <p>“Kuryeye Teslim Edildi” durumundaki bir siparişi, “Müşteriye Teslim Edildi” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Completed" }</pre>
--	--	---

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>- success (bool) : İlgili sipariş durumu güncelleme işlemi başarılı ise, bu alan true dönecektir. Başarısız olması durumunda false dönecektir.</p>

5.5 V2/CancelOrder

Onaylanmış bir siparişi iptal etmek için kullanılan servistir. Henüz onaylanmamış ve ya hazırlanmış siparişler iptal edilemezler.

Request

Metod	URL
POST	/Order/v2/CancelOrder

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "orderId": 0, "userId": 0, "notifyUser" : true, "cancelReasonId":0 }</pre>	<p>orderId: Siparişin id bilgisidir. Migros Yemek Panelindeki sipariş numarası bilgisi ile aynı bilgisidir.</p> <p>userId: Siparişi veren müşterinin Id bilgisidir.</p> <p>notifyUser: Bu alan kaldırılacaktır. Varsayılan olarak true iletebilirsiniz.</p> <p>cancelReasonId: İptal sebebi alanıdır. Mapping/v2/GetCancelReasons servisinden dönen iptal sebebi bilgilerinden seçili olanın reasonId bilgisini bu parametreye geçmeniz gerekmektedir.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>- success (bool) : İlgili sipariş iptal etme işlemi başarılı ise, bu alan true dönecektir. Başarısız olması durumunda false dönecektir.</p>

5.6 GetSummarizedOrder

Siparişe ait sipariş durumu, kurye durumu, atanmış kurye bilgisi ve sipariş iptal olup olmadığına dair özet bir bilgi dönen apidir.

Request

Metod	URL
POST	/Order/GetSummarizedOrder

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "orderId": 0, "userId": 0}</pre>	<p>orderId: Siparişin id bilgisidir. Migros Yemek Panelindeki sipariş numarası bilgisi ile aynı bilgisidir.</p> <p>userId: Siparişi veren müşterinin Id bilgisidir.</p>

Response

Status	Response
200	<pre>{ "data": { "id": 0, "status": "string", "isCancelled": true, "deliveryProvider": "string", "courier": { "id": 0, "name": "string" }, "courierAssigned": true }, "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- data (object): Siparişin özet bilgisini içeren nesne.</p> <p>-- id (long): Sipariş Id bilgisidir.</p> <p>-- status (string): Sipariş durumunu getirir. NEW_PENDING -> Onay bekleyen sipariş COLLECTING -> Hazırlanıyor COLLECTION_APPROVED -> Hazırlandı IN_DELIVERY -> Yolda DELIVERED -> Teslim Edildi CANCELLED -> İptal Edildi</p>

	<p>-- deliveryProvider (string): Teslimatı kimin yapacağını belirten alandır. "MIGROS" ise, teslimat Migros kuryesi ile, "RESTAURANT" ise, teslimat restoranın kendi kuryesi ile sağlanacaktır.</p> <p>-- courierAssigned (bool): Eğer teslimat tipi "MIGROS" ise, kuryenin atanıp atanmadığı bilgisini içerecektir. Atanmış kurye bilgileri, courier içerisinde verilecektir. Restoranın teslimatını yapacağı senaryoda bu alan varsayılan olarak false gelmektedir.</p> <p>-- courier (object): Atanmış ise, Migros kuryesine ait bilgileri barındırır.</p> <p>--- id (long): Migros Yemek Teslimatı kuryesinin id bilgisidir.</p> <p>--- name (string): Teslimatı yapacak kuryenin adıdır.</p> <p>- success (bool) : İlgili sipariş iptal etme işlemi başarılı ise, bu alan true dönecektir. Başarısız olması durumunda false dönecektir.</p>
--	--

5.7 UpdateOrderStatus

5 Mart 2024 tarihi itibarıyla kullanım dışı olmuştur ve kaldırılacaktır. Bu servis yerine Order/v2/UpdateOrderStatus servisi kullanılmalıdır.

Siparişi Onaylama, Reddetme, Hazırlandı olarak işaretleme, Kuryeye Verildi olarak işaretleme ve Teslim edildi olarak işaretleme amacı ile kullanılan servistir.

Request

Metod	URL
POST	/Order/UpdateOrderStatus

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "orderId": 0, "storeId": 0, "orderStatus": "string", "cancelReason": "string" }</pre>	<p>orderId: Siparişin id bilgisidir. Migros Yemek Panelindeki sipariş numarası bilgisi ile aynı bilgidir.</p> <p>storeId: Siparişi alan şubenin id bilgidir.</p> <p>orderStatus: Yapılmak istenen işlemin Enum karşılığının kontrol edildiği alandır. Buraya gönderilebilir parametreler şunlardır.</p> <p>Approved -> Onaylama Rejected -> Reddetme Prepared -> Hazırlandı Delivery -> Kuryeye Teslim Edildi (Yola Çıktı) Completed -> Müşteriye Teslim edildi.</p> <p>Siparişin akışı itibarıyla, siparişler ilk önce onaya düşmektedir. Bir sipariş, sipariş oluşturulma tarihinden sonraki 10 dakika içerisinde onaylanmaz ise otomatik iptal edilmektedir.</p> <p>Siparişi onaylamak için yapılacak istek örneği aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Approved" }</pre> <p>Siparişi onaylamak için yapılacak istek örneği Henüz onay aşamasında olan bir siparişi kabul etmemek-reddetmek için kullanılacak istek örneği aşağıdaki gibidir. Reddedilen sipariş, işlem göremez ve geçmiş siparişlere düşer.</p> <p>CANCEL REASON NAME alanına, sipariş iptal</p>

		<p>sebeplerini getiren listedeki <u>name</u> alanına karşılık gelen değer gönderilmelidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Approved", "cancelReason": "CANCEL_REASON_NAME" }</pre> <p>Onaylanan bir siparişi, “Hazırlandı” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Prepared" }</pre> <p>“Hazırlandı” durumundaki bir siparişi, “Kuryeye Teslim Edildi” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Delivery" }</pre> <p>“Kuryeye Teslim Edildi” durumundaki bir siparişi, “Müşteriye Teslim Edildi” olarak işaretlemek için gönderilecek örnek istek aşağıdaki gibidir.</p> <pre>{ "orderId": 10002123, "storeId": 230000102136, "orderStatus": "Completed" }</pre>
--	--	--

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>- success (bool) : İlgili sipariş durumu güncelleme işlemi başarılı ise, bu alan true dönecektir. Başarısız olması durumunda false dönecektir.</p>

5.8 — CancelOrder

5 Mart 2024 tarihi itibariyle kullanım dışı olmuştur ve kaldırılacaktır. Bu servis yerine Order/v2/CancelOrder servisi kullanılmalıdır.

Onaylanmış bir siparişi iptal etmek için kullanılan servistir. Henüz onaylanmamış ve ya hazırlanmış siparişler iptal edilemezler.

Request

Metod	URL
POST	/Order/CancelOrder

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "orderId": 0, "userId": 0, "notifyUser": true "cancelReason": "string" }</pre>	<p>orderId: Siparişin id bilgisidir. Migros Yemek Panelindeki sipariş numarası bilgisi ile aynı bilgisidir.</p> <p>userId: Siparişi veren müşterinin Id bilgisidir.</p> <p>notifyUser: Bu alan kaldırılacaktır. Varsayılan olarak true iletebilirsiniz.</p> <p>cancelReason: İptal sebebi alanıdır. Mapping/GetCancelReasons servisinden dönen iptal sebebi bilgilerinden seçili olanın name bilgisini bu parametreye geçmeniz gerekmektedir. Bu parametrenin doğru biçimde geçilmemesi halinde sipariş iptal sebebi TECHNICAL_PROBLEM (teknik problem) olarak varsayılan değer ataması yapılmaktadır.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>- success (bool) : İlgili sipariş iptal etme işlemi başarılı ise, bu alan true dönecektir. Başarısız olması durumunda false dönecektir.</p>

6 Payment Methods

6.1 GetPaymentMethodsByStoreId

Bir şubeye ait ödeme yöntemi bilgileri ve bu ödeme yöntemlerine ait aktiflik durumunu getiren servis. Tüm şubeler için, tüm ödeme yöntemleri bilgileri gelmekle birlikte, ödeme yöntemlerinin aktifliği şubeden şubeye değişiklik göstermektedir.

Request

Metod	URL
POST	/PaymentMethod/GetPaymentMethodsByStoreId

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0 }</pre>	storeId: Sorgulanmak istenilen şubeye ait Id bilgisinin iletileceği alandır. Long tipindedir.

Response

Status	Response
200	<pre>{ "data": { "onlinePayments": [{ "hashedNameId": 0, "name": "string", "title": "string", "description": "string", "status": true }], "onDeliveryPayments": [{ "hashedNameId": 0, "name": "string", "title": "string", "description": "string", "status": true }] }, "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>Data (object): Ödeme tiplerini online ve kapıda olarak iki ayrı listede dönen master modeldir.</p> <p>-onlinePayments (object array): Online ödeme yöntemleri bilgilerini içeren listedir.</p> <p>-- hashedNameId (long) : Ödeme yönteminin enum karşılığı olan id bilgisidir. Webhook modelinde paymentType.hashedNameId alanına karşılık gelir.</p>

	<p>-- name (string): Ödeme yönteminin enum tipindeki string karşılığıdır. Sipariş verilerinde paymentType alanına karşılık gelmektedir. (Webhook modelinde paymentType.name alanına karşılık gelir.)</p> <p>-- title (string) : Ödeme yönteminin başlık bilgisidir.</p> <p>-- description (string) : Ödeme bilgisinin açıklama bilgisidir.</p> <p>-- status (bool) : İlgili ödeme yönteminin, istek atılan şubedeki aktiflik durumunu belirtir. True ise ilgili ödeme yöntemi, ilgili restoranda aktiftir.</p> <p>- onDeliveryPayments (object array) : Kapıda ödeme yöntemlerinin listesini belirtir. İçerik olarak onlinePayments metodu ile aynı modele sahiptir.</p>
--	--

6.2 UpdatePaymentMethodStatus

İlgili restoran için ödeme yöntemlerini güncellemek için kullanılacak servistir.

Request

Metod	URL
POST	/PaymentMethod/UpdatePaymentMethodStatus

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "updatePaymentStatus": [{ "name": "string", "status": true }]}</pre>	<p>storeId: Ödeme yöntemi güncellenmek istenen restorana ait storeId bilgisidir.</p> <p>updatePaymentStatus: Güncelleştirilmek istenen ödeme yöntemlerini liste olarak alan modeldir.</p> <p>Name: Ödeme yöntemlerindeki Name alanına karşılık gelen bilgidir.</p> <p>status: İlgili ödeme yönteminin, hangi duruma getirilmek istendiğine ait bilgidir. True gönderilirse, ilgili ödeme yöntemi açılır. false gönderilirse, ilgili ödeme yöntemi kapatılır.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- success (bool) : True ise Ödeme yöntemleri güncelleme işlemi başarılı anlamına gelmektedir.</p>

7 Store

7.1 GetStoreDetail

Header'daki api key bilgisi ile şubeye ait detay bilgileri getiren servistir. Bu servis, gönderilen api key ile eşleşen bir restoran bulunması durumunda, restoran entegrasyonu inaktif olsa dahi, veri dönmektedir. Restoranın satışa açık olup olmadığını, entegrasyonunun aktif olup olmadığını, çalışma saatleri içinde olup olmadığı gibi bir çok bilgiyi barındırmaktadır.

Request

Metod	URL
GET	/Store/GetStoreDetail

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key

Response

Status	Response
200	<pre>{ "data": { "integrationIsActive": true, "id": 0, "warehouseId": 0, "storeGroupId": 0, "prepareDuration": 0, "embeddedCuisine": "string", "inWorkingHours": true, "storeName": "string", "active": true }, "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>- data (object array) : Restorana ait bilgileri içeren nesne.</p> <p>-- integrationIsActive (boolean): Restoranın entegrasyonunun aktif mi pasif mi olduğu bilgisini içeren alan. True ise entegrasyon aktif, pasif ise entegrasyon pasiftir.</p> <p>-- id (long) : Restorana ait id bilgisidir. Bazı metodların request body'lerinde yer alan StoreId alanına karşılık gelmektedir.</p> <p>-- warehouseId (long): StoreId nin son 6 hanesine karşılık gelmektedir.</p> <p>-- storeGroupId (long): Restoranın bağlı olduğu zincir bilgisine ait id bilgisidir.</p> <p>-- prepareDuration (long): Restoranın ürün hazırlık süresini getiren bilgidir. Dakika bilgisi gelmektedir.</p>

	<p>-- embeddedCuisine (string): Restoranın, hangi mutfak tiplerine ait ürünleri satışa sunduğu bilgisini belirtir.</p> <p>-- inWorkingHours (boolean): Restoranın çalışma saatleri içerisinde olup olmadığını belirten alan. True ise çalışma saatleri içerisinde, False ise çalışma saatleri dışında.</p> <p>-- storeName (string): Restoran adı bilgisini içeren alan.</p> <p>-- active (boolean): Restoranın satışa açık olup olmadığını bilgisini belirtir. True ise restoran satışa açıktır, false ise kapalıdır.</p> <p>NOT: Bir restoranın, Migros Yemek uygulamasında açık görünebilmesi için, active alanı ile birlikte inWorkingHours alanı da true olmalıdır. Restoran açık ancak çalışma saatleri dışındaysa sipariş alamaz.</p>
--	---

7.2 GetDefinedActiveRestaurantApiKeys

Entegratöre verilen özel secret key bilgisi ile, ilgili entegratörde aktif olan tüm restoranlara ait api key, zincir id ve restoran id bilgilerini dönen servis. **Şifreleme yapılmasına gerek yoktur.**

Request

Metod	URL
POST	/Store/GetDefinedActiveRestaurantApiKeys

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "secretKey": "string"}</pre>	secretKey: Entegratöre verilen ve şifrelemede kullanılacak bilgidir. Secret key bilgisi, test ve canlı ortamlarda değişiklik göstermektedir.

Response

Status	Response
200	<pre>{ "data": [{ "apiKey": "string", "posFirmId": "string", "isActive": true, "isPosFirmActive": true, "restaurantStoreGroups": [{ "storeGroupId": 0, "restaurantStores": [{ "storeId": 0] }] }]], "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- data (object array) : Aktif olarak ilgili entegratör ile entegrasyonu kullanan restoranlara ait zincir id, restoran id ve api key bilgilerini dönen servis. Entegrasyonları pasif olan işletmelere ait key bilgileri bu api içerisinde dönmemektedir.</p> <p>-- apiKey (string): Restorana ait api key bilgisidir.</p> <p>-- posFirmId (string): Entegratörün id bilgisidir.</p>

	<p>-- isActive (bool): ilgili api key bilgisinin aktifliğini belirten alandır.</p> <p>-- isPosFirmActive (string): ilgili entegratörün aktiflik durumunu belirten alandır.</p> <p>-- restaurantStoreGroups (object array): ilgili api keye bağlı zincir id bilgisini içeren metod. Normal şartlar altında bu array içerisinde yalnızca 1 adet zincir id bilgisi görüntülenmelidir.</p> <p>--- storeGroupId (long): ilgili restoranın bağlı bulunduğu zincire ait id bilgisidir.</p> <p>--- restaurantStores (object array): ilgili api key ve zincir altındaki şubeye ait id bilgisini içeren modeldir. İçerisinde yalnızca 1 adet storeId gelmelidir.</p> <p>--- storeId (long): Restoran(Şube) Id bilgisidir</p>
--	--

7.3 GetStoreGroups

İlgili api key bilgisine bağlı zincir bilgisini getiren servis.

Request

Metod	URL
GET	/Store/GetStoreGroups

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "name": "string", "description": "string", "logoPath": "string", "coverPhotoPath": "string" }], "success": true, "errorMessage": null, "validatonErrorMessages": null }</pre> <p>- data (object array) : Header'da gönderilen api keye bağlı zincir bilgilerini getiren servis. Array içerisinde 1 adet zincir bilgisi gelmektedir.</p> <p>-- id (long): Api Keyin bağlı olduğu restoranın zincir bilgilerini içerir.</p> <p>-- name (string): Zincir adı bilgisini içerir.</p> <p>-- logoPath (string): Zincir logosunun url bilgisini içerir</p> <p>-- coverPhotoPath (string): Zincirin header fotoğrafına ait link bilgisini içerir.</p>

7.4 GetStores

İlgili api key ve zincir id bilgisine göre restoran-şube bilgilerini getiren servis

Request

Metod	URL
POST	/Store/GetStores

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeGroupId": 0}</pre>	storeGroupId: Zincir id bilgisidir.

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "cityId": 0, "warehouseId": 0, "storeGroupId": 0, "prepareDuration": 0, "embeddedCuisine": "string", "inWorkingHours": true, "storeName": "string", "active": true }], "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- data (object array) : Header'da gönderilen api keye ve body de gönderilen zincir id bilgilerine göre sorgulanan restoran-şube bilgilerini içeren modeldir.</p> <p>-- id (long): Restoran-şube id bilgisidir.</p> <p>-- cityId (long): Restoranın bulunduğu şehre ait id bilgisidir.</p> <p>-- warehouseId (long): Restorana ait iç servislerde kullanılmakta olan id bilgisidir. RestoranId bilgisinin son 6 rakamına eşittir.</p> <p>-- storeGroupId (long): Zincire ait id bilgisidir.</p> <p>-- prepareDuration (long): Restorana ait genel sipariş hazırlama süresi bilgisidir.</p> <p>-- embeddedCuisine (string): Restoranın hizmet verdiği mutfak bilgilerini içerir.</p>

	<p>-- inWorkingHours (bool): Restoranın, servis isteğinin yapıldığı saatlerde çalışma saatlerinin içerisinde olup olmadığı bilgisini verir.</p> <p>-- storeName (string): Restoran-şube adı bilgisidir.</p> <p>-- active (bool): Restoranın satışa açık ya da kapalı olduğu bilgisidir.</p>
--	---

7.5 ActivateStore

Restoranı satışa açmak için kullanılan servis. Son güncellemeler ile birlikte, bu servis, restoran üzerinde geçici kapatma var ise bunu kaldırmaktadır.

Request

Metod	URL
POST	/Store/ActivateStore

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "warehouseId": 0}</pre>	<p>storeId: Restoran Id bilgisidir.</p> <p>warehouseId: Restorana bağlı iç id bilgisidir. storeId bilgisinin son 6 hanesine tekabül etmektedir.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- success (bool) : True ise restoranı satışa açma işlemi başarılı demektir.</p>

7.6 DeActivateStore

Restoranı satışa kapatmak için kullanılan servis. Son güncellemeler ile birlikte, bu servis, restoranı geçici olarak kapatmaktadır. Mevcut durumda bu süre 1 aydır. ActivateStore isteği geldiğinde bu geçici kapatma kaldırılır.

Request

Metod	URL
POST	/Store/DeActivateStore

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "warehouseId": 0}</pre>	<p>storeId: Restoran Id bilgisidir.</p> <p>warehouseId: Restorana bağlı iç id bilgisidir. storeId bilgisinin son 6 hanesine tekabül etmektedir.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>- success (bool) : True ise restoranı satışa kapatma işlemi başarılı demektir.</p>

7.7 GetStoreViewStatus

Restoranın satışa açık olup olmadığı bilgisini dönen servis. Eğer, restoran geçici olarak belli bir saat kapatılmış ise, açılacağı tarihi de dönmektedir.

Request

Metod	URL
POST	/Store/GetStoreViewStatus

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "storeGroupId": 0 }</pre>	storeId: Restoran Id bilgisidir. storeGroupId: Zincir/Marka Id bilgisidir.

Response

Status	Response
200	<pre>{ "data": { "status": "string", "openingTime": 0 }, "success": true, "errorMessage": { "errorCode": "string", "errorTitle": "string", "errorDetail": "string", "isSystemError": true }, "validationErrorMessages": [{ "field": "string", "cause": "string", "errorDetail": "string" }] }</pre> <p>- data(object): Restoran durumu bilgilerini içeren nesnedir.</p> <p>-- status (string): Restoranın durum bilgisini içermektedir.</p> <p>OPEN-> Restoran satışa açıktır.</p> <p>TEMPORARY_CLOSE-> Restoran geçici olarak kapalıdır.</p> <p>STORE_DRAFT->Restoran tanımlanmış ancak henüz tanım işlemleri tamamlanmamış ve satışa kapalıdır.</p> <p>STORE_PASSIVE->Restoran pasif durumdadır. Bu durumda iken yalnızca Migros Yemek Operasyon ekibi tarafından aktif edilebilir.</p>

	-- openingTime (long-nullable): Eğer restoran geçici olarak kapatılmış ise, açılacağı tarih epoch timestamp tipinde bu alanda gelir. Eğer, restoranda geçici kapatma yok ise bu alan null gelecektir.
--	---

7.8 AddStoreOffDate

Restoranı geçici olarak kapatmak için kullanılan servis. 1 saat, 4 saat ya da bir sonraki günün mesai başlangıç saatine kadar kapatmak için kullanılır.

Request

Metod	URL
POST	/Store/AddStoreOffDate

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "storeGroupId": 0, "storeOffDateOption": "string"}</pre>	<p>storeId: Restoran Id bilgisidir.</p> <p>storeGroupId: Zincir/Marka Id bilgisidir.</p> <p>storeOffDateOption: Restoranın ne kadar süreliğine kapatılacağı bilgisini tutar.</p> <p>Gönderilebilir bilgiler</p> <ul style="list-style-type: none">-ONE_HOUR-> 1 saat-FOUR_HOUR-> 4 saat-NEXT_WORK_HOUR -> Bir sonraki günün çalışma saatine kadar.

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validationErrorMessages": null}</pre> <p>- success (bool) : True ise restoran geçici kapatma işlemi başarılıdır.</p>

7.9 RemoveStoreOffDate

Restorandaki geçici kapatma işlemini kaldırarak, restoranı satışa açmak için kullanılır. Geçici kapatma kaydını siler, restoran aktif ise, satışa açılır.

Request

Metod	URL
POST	/Store/RemoveStoreOffDate

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "storeGroupId": 0}</pre>	storeId: Restoran Id bilgisidir. storeGroupId: Zincir/Marka Id bilgisidir.

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validationErrorMessages": null}</pre> <p>- success (bool) : True ise restoranda bulunan geçici kapatma durumu kaldırılmış ve satışa açılmış anlamı taşır.</p>

8 WorkingHours

8.1 GetStoreTimeSlot

Bir restorana-şubeye ait çalışma saatlerini getiren servis.

Request

Metod	URL
POST	/WorkingHour/GetStoreTimeSlot

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0}</pre>	storeId: Restoran Id bilgisidir.

Response

Status	Response
200	<pre>{ "data": [{ "storeId": 0, "startDay": "string", "startHour": "string", "endHour": "string", "active": true }], "success": true, "errorMessage": null, "validationErrorMessages": null}</pre> <p>Data (object array): Çalışma saatlerini, gün bazında içeren alandır.</p> <ul style="list-style-type: none">- storeId (long): Restoran Id bilgisidir.- startDay (string): İlgili tarihin başlangıç gününü enum olarak içerir. Gelebilecek değerler: PAZARTESİ, SALI, ÇARŞAMBA, PERŞEMBE, CUMA, CUMARTESİ, PAZAR- startHour (string): Çalışmanın başlayacağı saati belirtir, string olarak buçuklu saatler gelebilir. Örneğin; 10:00, 10:30, 11:00, 11:30, 12:00, 12:30, 13:00- endHour (string): Çalışmanın biteceği saati belirtir, string olarak buçuklu saatler gelebilir. Örneğin; 10:00, 10:30, 11:00, 11:30, 12:00, 12:30, 13:00- active (bool): İlgili çalışma gününde, restoranın açık mı kapalı mı olacağı bilgisini içerir.

8.2 UpsertStoreTimeSlot

Çalışma günlerini ve saatlerini güncellemek için kullanılan servis.

Request

Metod	URL
POST	/WorkingHour/UpsertStoreTimeSlot

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "timeSlotIds": [{ "startDay": "string", "startHour": "string", "endHour": "string", "active": true }] }</pre>	<p>storeId: Restoran Id bilgisidir.</p> <p>timeSlotIds: Çalışma günleri ve saatlerini içerir. StartDay bilgisi, bu liste içerisinde unique olmalıdır. Aynı gün bilgisinden 2 adet olması durumunda servis hata döner.</p> <p>startHour: Çalışmanın başlayacağı saati belirtir. hh:mm formatında saat bulunur. 11:00 gibi.</p> <p>endHour: Başlangıç saati ile aynı input yapısına sahiptir. Çalışmanın sonlanacağı saati belirtir.</p> <p>active: İlgili çalışma gününde restoranın aktiflik bilgisini tutar. True gönderilmesi durumunda restoran ilgili günde satışa açık olur, False göndermemiz durumunda restoran satışa kapatılır.</p>

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <p>- success (bool) : True ise restoranı satışa kapatma işlemi başarılı demektir.</p>

9 OptionItem

9.1 ActivateOptionItemStatusByStore

Bir opsiyonun ya da çıkarılabilir malzemenin, restoran bazında statüsünü aktifleştirmek için kullanılır.

Request

Metod	URL
POST	/OptionItem/ActivateOptionItemStatusByStore

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "optionItemId": 0}</pre>	storeId: Restoran Id bilgisidir. optionItemId: opsiyonun id bilgisidir

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>-success (bool) true gelmesi durumunda aktifleştirme işlemi başarılı, false gelmesi durumunda başarısızdır.</p>

9.2 DeActivateOptionItemStatusByStore

Bir opsiyonun ya da çıkarılabilir malzemenin, restoran bazında statüsünü pasif hale getirmek için kullanılır.

Request

Metod	URL
POST	/OptionItem/DeActivateOptionItemStatusByStore

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "optionItemId": 0}</pre>	storeId: Restoran Id bilgisidir. optionItemId: opsiyonun id bilgisidir

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>-success (bool) true gelmesi durumunda pasifleştirme işlemi başarılı, false gelmesi durumunda başarısızdır.</p>

10 Receipt

10.1 AddOrderReceipt

Siparişe ait fiş/fatura görselini yüklemek için kullanılacak apidir. Desteklenen dosya tipleri: jpeg,png,pdf.

Metod	URL
POST	/Receipt/AddOrderReceipt

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Header	Content-Type	multipart/form-data
Form Data	value	<p>Şifrelenmiş body text bilgisinin gönderileği alandır. Aşağıdaki modeldeki json bilgisi encrypt edildikten sonra value alanı ile gönderilmelidir.</p> <pre>{ "orderId": 0, "storeId": 0, "receiptDate": "2024-07-30",}</pre> <p>orderId: Sipariş Id bilgisidir. Long tipindedir. storeId: Restoran Id bilgisidir. Long tipindedir. receiptDate: Fatura/Fiş kesilme tarihidir. yyyy-MM-dd formatında gönderilmelidir. String tipindedir.</p>
Form Data	file	file: Fiş görseli. Kabul edilen uzantılar: jpeg,jpg,png,pdf

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": null, "validatonErrorMessages": null}</pre> <p>-success (bool) true gelmesi durumunda pasifleştirme işlemi başarılı, false gelmesi durumunda başarısızdır.</p>

10.2 GetReceipts

Gönderilen sipariş id'lerine ait fatura/fiş bilgilerine erişmek için kullanılacak api.

Request

Metod	URL
POST	/Receipt/GetReceipts

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "orderIds": [1,2,3] }</pre>	<p>storeId: Restoran Id bilgisidir.</p> <p>orderId: Sipariş Id'lerini içeren dizi-liste bilgisidir. OrderId'ler long tipindedir.</p>

Response

Status	Response
200	<pre>{ "data": [{ "id": 0, "orderId": 0, "storeId": 0, "url": "string", "receiptDate": 0 }], "success": true, "errorMessage": null, "validationErrorMessages": null }</pre> <ul style="list-style-type: none">- data (object array): Sorgulanan orderId lere ait fatura/fiş görsellerini dönen servistir. Eğer gönderilen orderId'lere ait yüklenmiş fatura/giş görseli yoksa, boş array dönmektedir.- id (long): Yüklenen fatura/fiş bilgisine ait Id bilgisidir.- orderId (long): Sipariş Id bilgisidir.- url (string): Yüklenen fatura/fiş görseline ait link bilgisini içerir.- receiptDate (long): Fatura/Fiş görseline ait, fiş kesilme tarihi bilgisini içerir. epoch timestamp formatındaki tarih bilgisidir.- success (bool) true gelmesi durumunda pasifleştirme işlemi başarılı, false gelmesi durumunda başarısızdır.

11 StoreArea

11.1 GetStoreAreas

Restorana tanımlanmış bölgelerin isim ve id bilgilerini dönen servistir.

Request

Metod	URL
POST	/StoreArea/GetStoreAreas

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "storeGroupId": 0 }</pre>	storeId: Restoran Id bilgisidir. storeGroupId: Zincir/Marka Id bilgisidir.

Response

Status	Response
200	<pre>{ "data": { "name": "string", "hexapolygonId": 0, "activeness": true }, "success": true, "errorMessage": { "errorCode": "string", "errorTitle": "string", "errorDetail": "string", "isSystemError": true }, "validatonErrorMessages": [{ "field": "string", "cause": "string", "errorDetail": "string" }] }</pre> <p>- data(object array): Restoran bölge bilgilerini içeren listedir.</p> <p>-- name (string): Bölgenin isim bilgisidir.</p> <p>-- hexapolygonId (long):Bölge Id bilgisidir. Bölgeyi açmak ve kapamakta kullanılan endpointlerde kullanılır.</p> <p>-- activeness (bool): Bölgenin aktif olup olmadığı bilgisini getirir.</p> <p>- success (bool) : true gelmesi durumunda api başarılı dönmüştür.</p>

11.2 ActivateArea

Restorana tanımlanmış bölgeyi aktif etmekte kullanılan endpoint

Request

Metod	URL
POST	/StoreArea/ActivateArea

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "hexapolygonId": 0 }</pre>	storeId: Restoran Id bilgisidir. hexapolygonId: Bölge Id bilgisidir.

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": { "errorCode": "string", "errorTitle": "string", "errorDetail": "string", "isSystemError": true }, "validationErrorMessages": [{ "field": "string", "cause": "string", "errorDetail": "string" }] }</pre> <p>- success (bool) : true gelmesi durumunda api başarılı dönmüştür.</p>

11.3 DeActivateArea

Restorana tanımlanmış bölgeyi pasif duruma getirmek için kullanılan endpoint

Request

Metod	URL
POST	/StoreArea/DeActivateArea

Parametreler

Type	Params	Values
Header	XApiKey	Restoran Api Key
Body	<pre>{ "storeId": 0, "hexapolygonId": 0 }</pre>	storeId: Restoran Id bilgisidir. hexapolygonId: Bölge Id bilgisidir.

Response

Status	Response
200	<pre>{ "success": true, "errorMessage": { "errorCode": "string", "errorTitle": "string", "errorDetail": "string", "isSystemError": true }, "validatonErrorMessages": [{ "field": "string", "cause": "string", "errorDetail": "string" }] }</pre> <p>- success (bool) : true gelmesi durumunda api başarılı dönmüştür.</p>