

پیوست فنی یکپارچه سازی با اسنپ فود

(نسخه رستورانی 4.2.7)

تاریخچه‌ی نسخ:

نسخه	تنظیم کننده(ها)	توضیحات
3.2.2	مرضیه تاجیک مهران محمودی محمدرضا شعبانی	نسخه‌ی اولیه‌ی مستندها یکپارچه‌سازی در قالب دو فایل مجزا
4.0.0	سعید حاجی‌زاده	تجمیع مستندات سابق
4.0.1	سعید حاجی‌زاده	به‌روزرسانی بخش گردش کار سفارش و افزودن چرخه‌های جدید افزودن کدهای خطای مربوط به Authentication
4.1.0	مرضیه تاجیک	افزودن endpoint های مربوط به تاپینگ
4.2.0	محمدرضا شعبانی	افزودن endpoint های جدید مربوط به تنظیم موجودی، تنظیم قیمت، دریافت اطلاعات محصول و دریافت اطلاعات مستر و اعلام منسوخ شدن endpoint سابق اعلام موجودی
4.2.1	سعید حاجی‌زاده	اصلاح متن توضیحات webhook (تغییری در webhook رخ نداده است)
4.2.2	محمدرضا شعبانی	افزودن endpoint جدید: update_product_details اصلاح آدرس endpoint های get_master_products و get_products افزودن فیلد disableUntil به product_toggle
4.2.3	سعید حاجی‌زاده	افزودن بخش اولویت انتخاب vendor به‌روزرسانی اطلاعات مربوط به کوپن‌ها در webhook افزودن توضیحات کوپن در بخش webhook افزودن userCode و userAddressCode به webhook تکمیل کدهای خطای accept افزودن فیلد vat به محصولات در product
4.2.4	حمید بیانک	افزودن فیلدهای: originprice, productDiscountSFShare, productDiscountVendorShare, couponDiscountSfShareAmount, couponDiscountVendorShareAmount به وبهوک, آپدیت ریسپانس سمپل اندپوینت latest آپدیت ریسپانس سمپل اندپوینت groups افزودن و آپدیت تاپینگ ها در اندپوینت topping جداسازی نسخه رستورانی و فروشگاه‌های

4.2.5	حمید بیانک	- افزودن API دریافت اطلاعات محصولات - تغییر آدرس بیس API ها روی پروداکشن
4.2.6	حمید بیانک	- اضافه شدن فیلد productCode به API دریافت اطلاعات محصولات - افزودن API سینک محصولات - رفع محدودیت عدم اعمال آپدیت ها در صورت عدم وجود یک یا چند آیدی ولید در درخواست های آپدیت بالک - اضافه شدن برخی تغییرات و راهنمایی ها
4.2.7	حمید بیانک	- اضافه شدن فیلد categoryCode به API دریافت اطلاعات محصولات - افزودن API سینک گروه محصولات - بهبود و بروزرسانی برخی توضیحات

مستند حاضر شامل کلیه‌ی اطلاعات فنی است که در فرآیند یکپارچه‌سازی با سامانه‌ی اسنپ فود مورد نیاز خواهد بود. این مستند در سه بخش به مجموعه API ارتباطی، webhook و چرخش کار سفارش‌ها می‌پردازد.

API اتوماسیون فروشگاه‌ها

در این بخش مستندات استفاده از Vendor Automation API را مشاهده می‌کنید. این API امکان انجام اعمالی چون تایید و رد بر روی سفارش‌های ثبت‌شده، به‌روزرسانی محصولات و منوها را فراهم می‌کند که به طور کلی شامل موارد زیر می‌شود:

- Ack, Pick, Reject and Accept Orders
- Create and Update Product
- Create and Update Menus
- List Recent Orders

دقت کنید که در تمامی درخواست‌ها و پاسخ‌ها، واحد قیمت‌های ذکر شده تومان و واحد مدت‌زمان‌ها دقیقه است.

برای استفاده از این API، در ابتدا باید از سرور authorization ما Oauth2 Access Token دریافت کنید. پس از دریافت این Token می‌بایست آن را در Header تمامی درخواست‌ها به این صورت ارسال کنید:

```
Authorization: Bearer {Oauth2 Access Token}
```

1- نحوه‌ی دریافت OAuth2 Access Token

دریافت Token از سرور، از نوع Password Grant است و برای دریافت آن باید مشخصات خود را شامل client_id, client_secret, username, password که قبلاً دریافت کرده‌اید در درخواست ارسال کنید.

قالب درخواست دریافت Token به صورت زیر می‌باشد:

```
curl -XPOST -H 'Content-Type: application/x-www-form-urlencoded' -d 'grant_type=password&scope=automation&client_id=<<your_client_id>>&client_secret=<<your_client_secret>>&username=<<your_username>>&password=<<your_password>>' '<<authorization_server_endpoint>>'
```

دامنه‌ی مورد استفاده ([authorization_server_endpoint](#)) را بسته به این که قصد دارید Token را از سرور Staging یا Production دریافت کنید، تعیین نمایید:

- Staging Auth Server: <https://staging-auth.snappfood.dev/token>

- Production Auth Server: <https://auth.snappfood.ir/token>

توجه کنید که خطاهایی با کدهای 3001 تا 3010 در هنگام استفاده از هر یک از endpointها مربوط به معتبر نبودن token است.

2- آدرس پایهی درخواستها

برای کلیدی endpoint می‌بایست به یکی از دو سرور زیر درخواست خود را ارسال کنید. در مرحله‌ی تست از سرور staging و بر روی محصول اصلی از سرور production استفاده کنید.

- Staging API Server: <https://staging-api.snappfood.ir>
- Production API Server: <https://api.snappfood.ir>

3- اولویت انتخاب vendor

در کلیدی endpoint هایی که به عنوان پارامتر ورودی نیاز به مشخص شدن یک vendor دارند، آن vendor بر اساس اولویت زیر مشخص می‌شود:

۱- **vendorCode در Body درخواست:** این پارامتر با مقدار رشته‌ای، برای مشخص کردن یک vendor به کار می‌رود. این کد در سفارش‌هایی که توسط webhook ارسال می‌شوند نیز وجود دارد و مشخص کننده‌ی vendor مرتبط با آن سفارش است.

۲- **vendorId در Body درخواست:** این پارامتر با مقدار عددی، شناسه‌ی یک vendor است. برای مشخص کردن vendor می‌توانید از این پارامتر استفاده کنید. این پارامتر در صورتی بررسی می‌شود که vendorCode مشخص نشده باشد.

۳- **token:** هر حساب کاربری (client_id) برای یک vendor مشخص ساخته شده است. اگر در یک درخواست هیچ یک از پارامترهای vendorCode و vendorId مشخص نشده باشد، عملیات مربوط به vendor در نظر گرفته می‌شود که توکن حاضر برای حساب کاربری متناظر با آن vendor صادر شده است.

4- لیست endpointها

در ادامه، کلیدی endpoint به همراه پاسخ‌های ممکن آنها به تشریح آورده شده است.

1-4- تایید دریافت سفارش (ack)

پس از این که از طریق Webhook¹، از ثبت سفارش جدیدی خبردار شدید، لازم است این آدرس را فراخوانی کنید. پارامتر order_code برای هر سفارش، باید برابر با مقدار code در Webhook دریافت شده‌ی مربوط به آن سفارش باشد.

URL :

```
POST /va/v1/order/{order_code}/ack
```

Parameters :

- order_code (String) - required: Unique code which indicates acking order

Response :

- 204 Successfully acked
- 400 There was an error
 - status = 1023 Order not found

Response Sample:

```
[
  {
    "status": "1023",
    "title": "resource not found",
    "detail": "the resource you are inserted does not exist"
  }
]
```

2-4- باز کردن سفارش (pick)

هنگامی که اپراتور شما سفارشی را انتخاب کرده و آن را در اپلیکیشن شما مشاهده می کند (پیش از آن که سفارش را تایید یا رد کند)، باید با استفاده از این درخواست ما را خبردار کنید.

URL :

```
POST /va/v1/order/{order_code}/pick
```

Parameters :

- order_code (String) - required: Unique code which indicates picking order

¹ فصل بعد به تشریح webhook اختصاص دارد.

Response :

- 204 Successfully picked
- 400 There was an error
 - status = 1023 Order not found
 - status = 1059 Order was accepted before
 - status = 1060 Order was rejected before
 - status = 1068 Order was picked before

Response Sample :

```
[
  {
    "status": "1023",
    "title": "resource not found",
    "detail": "the resource you are inserted does not exist"
  }
]
```

3-4- تایید سفارش (accept)

در صورتی که اپراتور شما سفارش را تایید کند، باید این آدرس را با داده‌های مورد نیاز فراخوانی کنید.

URL :

```
POST /va/v1/order/{order_code}/accept
```

Parameters :

- order_code (String) - required: Unique code which indicates accepting order

Request Body:

- Type: application/json
- packingPrice (Integer): Price of packing the products
 - Can't be more than main order packing price
- delta (Integer): The difference between total price and given money from customer
- deliveryTime (Integer): Time to prepare and delivery since now, in minutes
 - It should be sent when the vendor delivers the product itself
- riderPickupTime (Integer): Time to prepare since now, in minutes.
 - It should be sent when Snapp Express delivers the product

Request Body Sample:

```
{
  "packingPrice": 0,
  "delta": 0,
  "deliveryTime": 0,
  "riderPickupTime": 0
}
```

Response :

- 204 Successfully accepted
- 400 There was an error
 - status = 1023 Order not found
 - status = 1059 Order was accepted before
 - status = 1060 Order was rejected before
 - status = 2178 New packing price is more than order packing price
 - status = 2110 Delivery or preparation time exceeds our limits
 - MAX deliveryTime =70 minutes
 - MAX riderPickupTime = preparationTime + vendorMaxPreparationTime
 - status = 2112 Delta surpass our limits
 - MAX_DELTA = 500
 - For confectioneries limit is 5000

Response Sample :

```
[
  {
    "status": "1023",
    "title": "resource not found",
    "detail": "the resource you are inserted does not exist"
  }
]
```

4-4- نیازمند تماس (reject)

در صورتی که انجام سفارش برای شما امکان پذیر نباشد، باید این آدرس را به همراه دلیل رد سفارش فراخوانی کنید.

URL :

```
POST /va/v1/order/{order_code}/reject
```

Parameters :

- order_code (String) - required: Unique code which indicates accepting order

Request Body:

- Type: application/json
- reasonId (Integer): ID of decline reason
 - You may find the ID through decline-reason endpoint
- comment(String): More detailed description on why the order is rejected
- nonExistentProducts (array): products which are not available
 - It should be filled when reasonId=139
 - Mentioned products will be deactivated for the vendor

- id and suggestedProductIds may be used by restaurants
 - use your own dataset product ids here

Request Body Sample:

```
{
  "reasonId": 0,
  "comment": "string",
  "nonExistentProducts" :
  [
    {
      "id": 458795654,
      "suggestedProductIds": [1324567855, 4568745451]
    }
  ]
}
```

Response :

- 204 Successfully rejected
- 400 There was an error
 - status = 1023 Order not found
 - status = 1059 Order was accepted before
 - status = 1060 Order was rejected before
 - status = 1061 Reason of rejection not found
 - status = 1062 nonExistentProducts are not in the order

Response Sample :

```
[
  {
    "status": "1023",
    "title": "resource not found",
    "detail": "the resource you are inserted does not exist"
  }
]
```

5-4- لیست دلایل رد سفارش (decline_reason)

لیست دلایل تعریف شده برای رد سفارش (متناسب با نوع وندور) از این آدرس قابل دریافت است. لازم است در بدنه‌ی هر درخواست رد سفارش، id یکی از این موارد را به عنوان reasonId ارسال کنید.

URL :

```
GET /va/v1/order/decline-reason/
```

Response :

- 200 Successful Response

Response Sample :

```
[
  [
    {
      "id": 113,
      "title": "رستوران پیک ندارد",
      "level": 1
    },
    {
      "id": 153,
      "title": "تاخیر در زمان ارسال",
      "level": 1
    },
    {
      "id": 154,
      "name": "تغییر هزینه پیک",
      "level": 2
    }
  ]
]
```

4-6- لیست سفارشهای اخیر (latest)

لیست سفارش های اخیر هر vendor را نمایش می دهد (به صورت پیش فرض از آغاز روز اخیر)

URL :

POST /va/v1/order/latest

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- from (Date): The date from which orders will be fetched
- to (Date): The date till which orders will be fetched

Request Body Sample:

```
{
  "vendorCode": "p55ke1",
  "from": "2020-05-01 00:00:00",
  "to": "2020-05-31 23:59:59"
}
```

Response :

- 200 Successful Response
- 400 Bad Request

Response Sample :

```
[
  {
    "code": "mdmkgm",
    "userCode": "668v86",
    "fullName": "محمد خلیلیان",
    "firstName": "محمد",
    "lastName": "خلیلیان",
    "userAddressCode": "45oonn",
    "phone": "+989191234567",
    "price": 96850,
    "comment": "",
    "statusCode": "42",
    "deliverAddress": "تهرانپارس، بلوار استقلال، جنب پارک پلیس",
    "orderDate": 1635324570,
    "latitude": 35.756447568341002,
    "longitude": 51.544230999473001,
    "deliveryPrice": 16000,
    "packingPrice": 0,
    "deliveryTime": 60,
    "preparationTime": 15,
    "taxCoeff": 0,
    "tax": 0,
    "vat": 0,
    "expeditionType": "DELIVERY",
    "discountType": "",
    "discountValue": 0,
    "newOrderDate": "2021/10/27 12:19:30",
    "orderCoupon": {
      "id": 432611,
      "title": "تخفیف 25%",
      "descriptions": "برای اولین سفارش از این رستوران (سقف تخفیف : 30 هزار تومان)",
      "conditionCode": "orders_of_vendor_count",
      "rewardCode": "total_discount",
      "rewardParams": {
        "sf_share": "0",
        "total": "25",
        "max_discount_amount": "30000"
      }
    },
    "orderPaymentTypeCode": "ONLINE",
    "preOrderTime": null,
    "vendorMaxPreparationTime": 15,
    "vendorCode": "p55w4e",
    "products": [
      {
        "title": "چلو جوجه کباب",
        "description": "یک سیخ جوجه کباب زعفرانی 250 گرمی، 400 گرم برنج اعلا ایرانی، کره حیوانی، به همراه دورچین: گوجه کبابی، لیمو، پیاز کبابی، فلفل دلمه کبابی، فلفل تند کبابی و نان داغ تنوری"
```

```

        "price": 80000,
        "toppingPrice": 0,
        "totalPrice": 80000,
        "originPrice": 80000,
        "productDiscountSFShare": 0,
        "productDiscountVendorShare": 0,
        "discount": 0,
        "productDiscount": 0,
        "productDiscountZFShare": 0,
        "quantity": 1,
        "containerPrice": 0,
        "containerPriceDiscount": 0,
        "vat": 0,
        "vmsFoodId": 10
    },
    {
        "title": "ماء الشعير قوطى هلو",
        "description": "میلی لیتر 300",
        "price": 9800,
        "toppingPrice": 0,
        "totalPrice": 9800,
        "originPrice": 9800,
        "productDiscountSFShare": 0,
        "productDiscountVendorShare": 0,
        "discount": 0,
        "productDiscount": 0,
        "productDiscountZFShare": 0,
        "quantity": 1,
        "containerPrice": 0,
        "containerPriceDiscount": 0,
        "vat": 0,
        "vmsFoodId": 27
    }
],
"couponDiscountSfShareAmount": 0,
"couponDiscountVendorShareAmount": 26950
}
    
```

7-4- ساخت و بهروزرسانی گروه محصولات (menu)

با استفاده از این آدرس می‌توانید منو جدیدی بسازید (create) یا اطلاعات یک منوی موجود را بهروزرسانی کنید (update). مقدار id در بدنه‌ی درخواست را برابر با id منو در سیستم خودتان قرار دهید. در صورتی که id ارسال شده در درخواست از پیش موجود باشد، این درخواست برای update منو در نظر گرفته خواهد شد.

URL :

POST /va/v1/menu/

Request Body:

Type: application/json

vendorCode(String): Code of vendor

- You may ignore vendorCode, so the vendor associated to the token will be used

menus (Array): List of menus to be created/updated

id (Integer): Menu ID in your dataset

orderIndex (Integer): Position of menu in vendor page

title (String): Menu Title

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "menus": [
    {
      "id": 8,
      "orderIndex": 9,
      "title": "دوغ"
    }
  ]
}
```

Response :

204	Successfully Updated Product Status
400	Bad Request

8-4- ساخت و بهروزرسانی محصولات (product)

با استفاده از این آدرس می‌توانید محصول جدیدی را بسازید (create) یا اطلاعات یک محصول موجود را بهروزرسانی کنید (update). مقدار productId در بدنه‌ی درخواست را برابر با id محصول در سیستم خودتان قرار

دهید. در صورتی که productId ارسال شده در درخواست، از پیش موجود باشد، این درخواست برای update محصول در نظر گرفته خواهد شد.

URL:

```
POST /va/v1/product/
```

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- products (Array): Product to be created/updated
- menuCategoryId (Integer): Menu ID in your dataset
- productId (Integer): Product ID in your dataset (Maximum 10 digits)
- title (String): Title of product
- position (Integer): Position of product in menu
- description (String): Detailed description of product
- price (Integer): Price of product
- containerPrice (Integer): Price of product container
- schedules (Array): Periods of time the product should be activated
- vat (Integer): Value added tax for the product in percent

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "products": [
    {
      "menuCategoryId": 8,
      "productId": 9,
      "title": "دوغ",
      "position": 1,
      "description": "دوغ گازدار گرما دیده",
      "price": 12000,
      "containerPrice": 1000,
      "schedules": [
        {
          "day": 1,
          "start": "10:00:00",
          "stop": "16:00:00"
        }
      ]
    },
    {
      "vat": 9
    }
  ]
}
```

Response :

- 204 Successful Response
- 400 Bad Request

9-4- به روز رسانی وضعیت محصول (product_toggle)

با استفاده از این آدرس می توانید وضعیت محصول را تغییر دهید. محصول در صورت disable شدن، تا پایان روز غیرفعال نشان داده می شود. این وضعیت در حالتی استفاده می شود که برای مثال محصول در آن روز ناموجود شده است.

همچنین محصول می تواند با تعیین مقدار active، فعال یا غیرفعال شود. محصول غیرفعال در منوی شما نشان داده نخواهد شد، مگر آنکه خودتان دوباره فعال کنید.

URL :

PUT /va/v1/product/

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- id (Integer): Product ID in your dataset
- active (Boolean): Activation status of product
 - If you deactivate a product by this parameter, it only will be activated using this endpoint
- disableUntil (Date): Disable a product until a specific date
 - In the below sample the product will be enabled on '2022-01-01 00:00:00'
 - This parameter is optional
- disable (Boolean): Activation status of product
 - If you deactivate a product by this parameter, it will be activated after "disableUntil", or on absence of "disableUntil" will be activated tomorrow

Request Body Sample

Restaurant:

```
{
  "vendorCode": "0rxxo",
  "products": [
    {
      "id": 8,
      "active": true,
      "disable": true,
      "disableUntil": "2022-01-01"
    }
  ]
}
```

```
}
```

Response :

- 204 Successfully Updated Product Status
- 400 Bad Request

10-4- به روزرسانی وضعیت گروه محصول (menu_toggle)

با استفاده از این آدرس می توانید وضعیت یک منوی موجود را به روزرسانی کنید. با تعیین مقدار active می توانید منو را برای مدتی نامحدود غیرفعال کرده و یا دوباره فعال کنید. در صورت غیرفعال بودن یک منو، محصولات مربوط به آن نمایش داده نخواهد شد.

URL :

```
PUT /va/v1/menu/
```

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- id (Integer): Menu ID in your dataset
- active (Boolean): Activation status of menu

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "menus": [
    {
      "id": 6,
      "active": false
    }
  ]
}
```

Response :

- 204 Successfully Updated Product Status
- 400 Bad Request

11-4- دریافت اطلاعات محصولات

با استفاده از این آدرس میتوانید اطلاعات محصولات خود شامل وضعیت محصول، قیمت و دیگر متعلقات محصول را دریافت کنید .

URL :

```
GET /va/v1/product/
```

Response :

- 200 Successful Response
- 400 Bad Request

Response Body Sample:

```
{
  "status": true,
  "data": {
    "vendorCode": "0q54rd",
    "products": [
      {
        "productId": 1280,
        "title": "پیتزا 2",
        "description": "تست 1",
        "position": 1,
        "vat": 9,
        "productCode": "BAdQ7OdV1y",
        "isActive": true,
        "isReviewed": true,
        "capacity": 10,
        "stock": null,
        "productType": [],
        "price": 500,
        "containerPrice": 200,
        "disabled": false,
        "disabledUntil": null,
        "schedules": [],
      }
    ]
  }
}
```

```

        "toppings": [
            {
                "vmsFoodId": 1270,
                "title": "پنیر گودا",
                "description": null,
                "active": false,
                "toppingGroupId": 206329,
                "price": 8000,
                "disabled": false,
                "disabledUntil": null
            }
        ],
        "menuCategory": {
            "vmsCategoryId": 153,
            "categoryCode": "qz4Pkjy2lkD"
            "title": "تست1",
            "isActive": true,
            "isDeleted": false,
            "isReviewed": true,
            "visibility": true
        }
    },
    {
        "productId": 1260,
        "title": "تست2",
        "description": "تست2",
        "position": 2,
        "vat": 9,
        "isActive": true,
        "isReviewed": true,
        "capacity": 10,
        "stock": null,
        "productType": [],
        "price": 500,
        "containerPrice": 100,
        "disabled": false,
        "disabledUntil": null,
        "schedules": [],
        "toppings": [
            {
                "vmsFoodId": 1200,
                "title": "پنیر گودا",
                "description": null,
                "active": false,
                "toppingGroupId": 206329,
                "price": 8000,
                "disabled": false,
                "disabledUntil": null
            }
        ],
        "menuCategory": {
            "vmsCategoryId": 154,
            "categoryCode": "qz4Pkjy2lkD"
            "title": "تست1",
            "isActive": true,
            "isDeleted": false,
    
```

```

        "isReviewed": true,
        "visibility": true
      }
    ]
  }
}

```

12-4- سینک آیدی گروه محصولات وندور با اسنپ فود

با استفاده از آدرس زیر می‌توانید آیدی گروه محصول وندور را به گروه محصول متناظر وندور در اسنپ فود جهت سینک شدن اختصاص دهید.

URL :

POST <https://api.snappfood.ir/va/v1/category/sync/categoryId>

Request Body:

- categoryId (Integer): Category ID in your dataset (Maximum 10 digits)
- categoryCode (String): Vendor product category code at Snappfood (Sync code)

Request Body Sample:

```

{
  "categories": [
    {
      "categoryId": 100,
      "categoryCode": "qz4P2lplwD"
    }
  ]
}

```

Response :

200	Successful Response
400	Bad Request

13-4- سینک آیدی محصولات وندور با اسنپ فود

با استفاده از آدرس زیر می‌توانید آیدی محصول وندور را به محصول متناظر وندور در اسنپ فود جهت سینک شدن اختصاص دهید

URL :

`POST https://api.snappfood.ir/va/v1/product/sync/productId`

Request Body:

- productId (Integer): Product ID in your dataset (Maximum 10 digits)
- productCode (String): Vendor Product code at Snappfood (Sync code)

Request Body Sample:

```
{
  "products": [
    {
      "productId": 12345,
      "productCode": "BAxxxxxV1y"
    }
  ]
}
```

Response :

200	Successful Response
400	Bad Request

14-4- نیاز به پشتیبانی فنی (support)

با استفاده از این آدرس می توانید نیاز خود به پشتیبانی فنی را اعلام کنید.

URL :

POST /va/v1/support/contact

Response :

- 200 Successfully registered the ticket
- 500 Error on registering ticket

15-4- دریافت لیست تاپینگ های وندور (get_toppings)

هر وندور می تواند تعدادی topping group داشته باشد. هر یک از این گروه ها شامل تعدادی تاپینگ است و همچنین تاپینگها و محصولات که با هم رابطه دارند ، این API لیست آنها را باز می گرداند.

URL:

GET /va/v1/topping/groups

Response:

- 200 Successful Response

Response Body:

- MinLimit (Integer): minimum number of toppings selected from group
- maxLimit (Integer): maximum number of toppings selected from group

Response Sample:

```
{
  "toppingGroups": [
    {
      "id": 929,
      "title": "پنیر",
      "minLimit": 0,
      "maxLimit": 1,
      "position": 1,
      "toppings": [
        {
          "id": 103,
          "title": "پنیر چدار",

```

```

        "price": 1000
      },
      {
        "id": 104,
        "title": "پنیر موزارلا",
        "price": 1000
      }
    ]
  },
  "productToppings": [
    {
      "productVariationId": 30796,
      "toppingId": 103
    },
    {
      "productVariationId": 30780,
      "toppingId": 104
    }
  ]
}

```

4-16- ساخت تاپینگ (create_topping)

از این endpoint برای ساختن تاپینگ استفاده می شود.

نکته : آیدی تاپینگ ها نباید با آیدی محصولات مشترک باشد.

URL:

POST /va/v1/topping/

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- Contains a single topping group info with its corresponding toppings
- toppingGroupId (Integer): The topping group will be updated to this. If you don't set it, a new topping group will be created.
- ToppingGroup (Object): Data to create a new topping:
 - title(String): Title of the topping group
 - min (Integer): Minimum number of toppings that user must select from this group.

- max (Integer): Maximum number of toppings that user must select from this group.
 - position (Integer): position of the topping
 - **You must provide either toppingGroupId, or toppingGroup object.**
- toppings (Array): An array of objects to create new toppings:
 - id (Integer): id of the topping
 - title (String): Topping title
 - price (Integer): Topping price
 - containerPrice(Integer): Container price of the topping

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "toppingGroup": {
    "title": "نان",
    "min": 0,
    "max": 1,
    "position": 1
  },
  "toppings": [
    {
      "id": 103,
      "title": "نان جو",
      "price": 1000,
      "containerPrice": 1000
    }
  ]
}
```

Response:

- 204 Successfully created toppings.
- 400 Bad Request

همچنین برای اضافه کردن تاپینگ به گروه های از قبل تعریف شده و یا آپدیت آنها میتوانید از درخواست زیر استفاده کنید.

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "toppingGroupId": 11222,
  "toppings": [
    {
      "id": 102,
      "title": "سس کچاپ",
      "price": 1000,
      "containerPrice": 500
    }
  ]
}
```

```
}
```

Response:

- 204 Successfully added toppings.
- 400 Bad Request

17-4- اتصال تاپینگ به محصول (product_topping)

با استفاده از این endpoint می‌توانید رابطه‌ی یک تاپینگ را با یک محصول مدیریت کنید.

URL:

```
POST /va/v1/topping/add-to-product
```

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- toppings (Array): An array of objects, each containing a productId and a toppingId
- toppingId (Integer): Id of the topping
- productId (Integer) : Id of the product

Request Body Sample:

```
{
  "vendorCode": "0rxxo",
  "toppings": [
    {
      "productId": 17,
      "toppingId": 101
    },
    {
      "productId": 16,
      "toppingId": 10000
    }
  ]
}
```

Response:

- 204 Successfully added toppings to products.
- 400 Topping or product with the provided id does not exist.

18-4- بروز رسانی اطلاعات محصولات (update_product_details)

با استفاده از این endpoint می‌توانید قیمت و سقف خرید محصولات را به‌روزرسانی کنید.

URL:

POST /va/v1/product/productDetails

Request Body:

- Type: application/json
- vendorCode(String): Code of vendor
 - You may ignore vendorCode, so the vendor associated to the token will be used
- products (Array): An array of objects, each containing a barcode or id
- id(integer) : Id of the product
- price(integer) : New price of the product
- capacity(integer) : New capacity on one order for the product

Request Body Sample:

```
{
  "vendorCode": "fx34Dm",
  "products": [
    {
      "id": 123,
      "price" : 1000,
      "capacity" : 10,
    }
  ]
}
```

Response:

- 200 Successfully updated
- 400 Request is not valid

Webhook

پس از ثبت سفارش جدید و رخ دادن هر تغییر قابل ملاحظه‌ای در وضعیت سفارش، ما با استفاده از این Webhook شما را باخبر می‌کنیم. اطلاعات سفارش در هر بار فراخوانی Webhook به طور کامل ارسال خواهد شد.

می‌توانید برای تمامی شعب خود از یک آدرس برای دریافت webhook استفاده کرده یا به ازای هر شعبه از فروشگاه، آدرس متفاوتی برای webhook داشته باشید.

امکان فرستادن HMAC Token در Webhook وجود دارد. این امکان اختیاری است و برای فعال‌سازی آن لازم است با پشتیبانی اسنپفود تماس بگیرید.

توجه: Content Type در این درخواست‌ها 'application/x-www-form-urlencoded' است. در ادامه‌ی مستند هم نمونه JSON ساخته شده از درخواست اصلی آمده است و هم Body به شکل x-www-form-urlencoded

Webhook Headers:

```
Authorization: 'HMAC snappfood:'+timestamp+':'+base64(hmac('sha256', 'POST' + timestamp, yourSecret))
User-Agent: 'SnappFood'
Content-Type: 'application/x-www-form-urlencoded'
```

Webhook Body:

- Type: application/x-www-form-urlencoded
- code (String): Order Code
- price (Integer): Order Total Price
- userCode(String): A unique code which identifies end user (customer)
- userAddressCode(String): A unique which identifies user address
- expeditionType (String): How the order will be delivered
 - [PICKUP, ZF_EXPRESS, MIARE, DELIVERY, PICK_MAN]
 - DELIVERY means restaurant own delivery
 - PICKUP means client will pick his order up
 - ZF_EXPRESS means order will be delivered by Zoodfood Express
- statusCode (Integer): Order Status Code
 - All status codes are explained in the next chapter
- orderDate (timestamp): Order Register Date
- fullName (String): Client Full Name
- firstName(String): Client First Name
- lastName (String): Client Last Name
- latitude (Float): Client Latitude
- longitude (Float): Client Longitude
- deliverAddress (String): Client Address
- comment (String): Client Detailed Description on Order
- phone (String): Client Phone Number

- tax (Integer): Calculated Tax Amount
- deliveryPrice (Integer): Delivery Price
- packingPrice (Integer): Packing Price
- deliveryTime (Integer): Time to prepare and delivery since now, in minutes
- preparationTime (Integer): Time to prepare since now, in minutes
- vat (float): Value Added Tax Coefficient
- taxCoeff (Integer): Value Added Tax Percent (= vat*100)
- discountType (String): Discount Type
 - [percentage, amount]
- discountValue (Integer): Price or Percent of Discount
- newOrderDate (dateTime): Order Register Date
- orderCoupon (Object): Order Coupon
 - It could be NULL or coupon object
 - rewardCode (String): Type of reward
 - [extra_item, total_discount, free_delivery_fee]
 - title(String): Persian definition of reward
 - description(String): Persian definition of condition
 - rewardParams(Map): A map of key-value parameters which define the reward.
They will be different for each rewardCode. You may find a complete description on next part.
- orderPaymentTypeCode (String): Order Payment Type
 - [CREDIT, CASH, ONLINE]
- preOrderTime : Custom pre-order time registered by the customer
- vendorMaxPreparationTime : The maximum addable time to the default preparation time.
- vendorCode (String): Vendor Code to Which Order Associated
- Products (Array): Array of Products in the Order
 - id (Integer): Product ID
 - quantity (Integer): Ordered Quantity of Product
 - originprice (Integer): Product Price
 - title (String): Product Title
 - productDiscountSFShare (Integer): Snappfood Discount Share Calculated Price
 - productDiscountVendorShare (Integer): Vendor Discount Share Calculated Price
 - vat (float): Value Added Tax Coefficient
 - barcode (String): Product Barcode
 - toppings (Array): Array of topping objects
- couponDiscountSfShareAmount (Integer) : Snappfood Coupon Discount Share Calculated Price.
- couponDiscountVendorShareAmount (Integer) : Vendor Coupon Discount Share Calculated Price.

Webhook JSON sample:

```
{
  "code": " ykj7g6vy ",
  "userCode": "0gw43n",
  "fullName": " حمید",
  "firstName": "حمید",
  "lastName": "بیانک",
  "userAddressCode": "5q5y1x",
  "phone": "+989991111111",
  "price": 1910,
  "comment": " تست اردر",
  "statusCode": "56",
  "deliverAddress": " تهران، زعفرانیه، ولی عصر، ولی الله فلاحی",
  "a": "قرار داده",
  "orderDate": 1714389167,
  "latitude": 35.804123071093002,
  "longitude": 51.419917821847001,
  "deliveryPrice": 500,
  "packingPrice": 200,
  "deliveryTime": 48,
  "preparationTime": 15,
  "taxCoeff": 10,
  "tax": 110,
  "vat": 0.10000000000000001,
  "expeditionType": "DELIVERY",
  "discountType": "",
  "discountValue": 0,
  "newOrderDate": "2024/04/29 14:42:47",
  "orderCoupon": null,
  "orderPaymentTypeCode": "CASH",
  "preOrderTime": null,
  "vendorMaxPreparationTime": 15,
  "vendorCode": "0q54rd",
  "products": [
    {
      "id": 101,
      "quantity": 1,
      "price": 500,
      "originPrice": 500,
      "productDiscountSFShare": 0,
      "productDiscountVendorShare": 0,
      "title": " پیتزای تستی",
      "discount": 0,
      "vat": 0.10000000000000001,
      "barcode": ""
    },
    {
      "id": 102,
      "quantity": 1,
      "price": 600,
      "originPrice": 600,
      "productDiscountSFShare": 0,
      "productDiscountVendorShare": 0,
      "title": " پیتزای تستی",
      "discount": 0,
      "vat": 0.10000000000000001,
      "barcode": ""
    }
  ],
  "couponDiscountSfShareAmount": 0,
  "couponDiscountVendorShareAmount": 0,
  "bikerName": "",
  "canBeDeliveredByOnDemandExpressTrip": false
}
```

Webhook Body 'application/x-www-form-urlencoded' Sample:

```
code=ykj7g6vy&userCode=0gw43n&fullName=%D8%AD%D9%85%DB%8C%D8%AF+%D8%A8%DB%8C%D8%A7%D9%86%DA%A9&firstName=%D8%AD%D9%85%DB%8C%D8%AF&lastName=%D8%A8%DB%8C%D8%A7%D9%86%DA%A9&userAddressCode=5q5y1x&phone=%2B989991111111&price=1910&comment=%D8%AA%D8%B3%D8%AA+%D8%A7%D8%B1%D8%AF%D8%B1+%DB%B1&statusCode=56&deliverAddress=%D8%AA%D9%87%D8%B1%D8%A7%D9%86%D8%8C+%D8%B2%D8%B9%D9%81%D8%B1%D8%A7%D9%86%DB%8C%D9%87%D8%8C+%D9%88%D9%84%DB%8C+%D8%B9%D8%B5%D8%B1%D8%8C+%D9%88%D9%84%DB%8C%E2%80%8C+%D8%A7%D9%84%D9%84%D9%87+%D9%81%D9%84%D8%A7%D8%AD%DB%8C+%D9%82%D8%A8%D9%84+%D8%A7%D8%B2+%D9%85%D9%82%D8%AF%D8%B3+%D8%A7%D8%B1%D8%AF%D8%A8%DB%8C%D9%84%DB%8C%D8%8C+%D9%BE%D9%84%D8%A7%DA%A9+2+%28%D8%B3%D8%A7%D8%AE%D8%A%D9%85%D8%A7%D9%86+%D8%A7%D8%B3%D9%86%D9%BE%29%D8%8C+%D8%BA%D8%B0%D8%A7+%D8%AF%D8%A7%D8%AE%D9%84+%D8%A8%D8%A7%DA%A9%D8%B3+a+%D9%82%D8%B1%D8%A7%D8%B1+%D8%AF%D8%A7%D8%AF%D9%87+%D8%B4%D9%88%D8%AF&orderDate=1714389167&latitude=35.804123071093&longitude=51.419917821847&deliveryPrice=500&packingPrice=200&deliveryTime=48&preparationTime=15&taxCoeff=10&tax=110&vat=0.1&expeditionType=DELIVERY&discountType=&discountValue=0&newOrderDate=2024%2F04%2F29+14%3A42%3A47&orderPaymentTypeCode=CASH&vendorMaxPreparationTime=15&vendorCode=0q54rd&products%5B0%5D%5Bid%5D=101&products%5B0%5D%5Bquantity%5D=1&products%5B0%5D%5Bprice%5D=500&products%5B0%5D%5BoriginPrice%5D=500&products%5B0%5D%5BproductDiscountSFShare%5D=0&products%5B0%5D%5BproductDiscountVendorShare%5D=0&products%5B0%5D%5Btitle%5D=+%D9%BE%DB%8C%D8%AA%D8%B2%D8%A7%DB%8C+%D8%AA%D8%B3%D8%AA%DB%8C+2&products%5B0%5D%5Bdiscount%5D=0&products%5B0%5D%5Bvat%5D=0.1&products%5B0%5D%5Bbarcode%5D=&products%5B1%5D%5Bid%5D=102&products%5B1%5D%5Bquantity%5D=1&products%5B1%5D%5Bprice%5D=600&products%5B1%5D%5BoriginPrice%5D=600&products%5B1%5D%5BproductDiscountSFShare%5D=0&products%5B1%5D%5BproductDiscountVendorShare%5D=0&products%5B1%5D%5Btitle%5D=+%D9%BE%DB%8C%D8%AA%D8%B2%D8%A7%DB%8C+%D8%AA%D8%B3%D8%AA%DB%8C+2&products%5B1%5D%5Bdiscount%5D=0&products%5B1%5D%5Bvat%5D=0.1&products%5B1%5D%5Bbarcode%5D=&couponDiscountSfShareAmount=0&couponDiscountVendorShareAmount=0&bikerName=&canBeDeliveredByOnDemandExpressTrip=0
```

Response :

- 200 or 204 Webhook Received Successfully
- All other response codes will be assumed as vendor service failure

1- مقادیر کوپن

مقادیر آرایه‌ی orderCoupon[rewardParams] بر اساس نوع پاداش (orderCoupon[rewardCode]) حالات متفاوتی خواهند داشت. در این بخش اجزای این map برای هر یک از حالات مختلف پاداش توصیف می‌شود.

1-1- آیتم اضافه (extra_item)

```
"rewardParams":{
  "item":[persian description like: نوشابه رایگان],
  "VmsFoodId":[product_Id of extra item on your dataset]
}
```

1-2- ارسال رایگان (free_delivery_fee)

```
"rewardParams":{
  "delivery_fee":1 ,
  "sf_share":[SnappFood Share of delivery cost in percent]
}
```

1-3- تخفیف روی کل سفارش (total_discount)

```
"rewardParams":{
  "total":[total percent of discount] ,
  "max_discount_amount":[maximum amount of discount in Tooman] ,
  "sf_share":[SnappFood Share of discount]}
}
```

گردش کار سفارش‌ها

فهرست Status Code هایی که توسط webhook برای فروشگاه‌ها ارسال می‌شود از این قرار است:

- 56: سفارش جدید
- 714: ارسال شده برای فروشگاه
- 61: دریافت شده توسط فروشگاه
- 713: باز شده توسط فروشگاه
- 42: سفارش تایید شده

- 51: سفارش نیاز به تماس شده (توسط وندور رد شده)
- 54: سفارش کنسل شده
- 71: نیازمند پرداخت اضافه

شرح کل فرآیند سفارش

- پس از ثبت سفارش و پرداخت مشتری در سامانه اسنپ فود ، سفارش برای اولین بار با کد 56 برای webhook فروشگاه فرستاده می شود.
- پس از ارسال موفق، سفارش به کد 714 تغییر یافته و برای فروشگاه ارسال می شود.
- فروشگاه پس از دریافت سفارش می بایست API شماره ۱ (ack) را فراخوانی کرده و پاسخ 204 بگیرد. پس از آن سفارش به کد 61 تغییر می کند.
- پس از باز شدن سفارش توسط فروشنده در نرم افزار فروشگاهی، نرم افزار فروشگاه می بایست API شماره ۲ (pick) را فراخوانی کند و پاسخ 204 بگیرد. سپس سفارش به کد 713 تغییر می کند.
- پس از بررسی سفارش توسط فروشنده یکی از دو حالت زیر اتفاق می افتد :

۱- تایید سفارش توسط فروشگاه

در این حالت نرم افزار فروشگاهی API شماره ۳ (accept) را فراخوانی می کند و پاسخ 204 می گیرد. سفارش به کد 42 تغییر می کند. پس از تایید سفارش، شما تا یک ساعت فرصت دارید که در صورت بروز مشکلی دوباره با استفاده از API شماره ۴ (reject) آن سفارش را نیازمند تماس کنید. در غیر این صورت فرآیند سفارش به پایان می رسد.

2- رد سفارش توسط فروشگاه (نیاز به تماس)

در این حالت نرم افزار فروشگاهی API شماره ۴ (reject) را فراخوانی می کند و سفارش به کد 51 تغییر می کند. برای فراخوانی reject لازم است تا نرم افزار فروشگاهی یکی از دلایل رد سفارش را (که با استفاده از API شماره 5 (decline_reason) قابل دریافت است) به عنوان علت رد سفارش مشخص کند.

بعد از این مرحله پشتیبانی اسنپ فود با تماس به فروشگاه و مشتری یکی از سه اقدام زیر را انجام خواهد داد:

الف- کنسل کردن سفارش : سفارش به کد 54 تغییر یافته و فرآیند این سفارش به اتمام می‌رسد.

ب- تغییر اقلام سفارش و ارسال مجدد : سفارش (با همان Order Code قبلی) با تغییراتی در محصولات دوباره با کد 56 به فروشگاه ارسال می‌شود.

ج- تغییر اقلام سفارش و تغییر مبلغ آن: در صورتی که سفارش پس از تغییر نیازمند پرداخت اضافه باشد، سفارش به کد 71 تغییر می‌کند. اگر مشتری در مدت زمان مشخص مبلغ اضافه را پرداخت کند سفارش با کد 56 به فروشگاه ارسال می‌شود. اگر این مهلت به اتمام برسد و مشتری مبلغ اضافه را پرداخت نکند سفارش لغو شده و با کد 54 برای فروشگاه ارسال می‌شود.

در صورت وجود هرگونه سوال لطفاً با اکانت زیراز طریق اسکایپ ارتباط بگیرید.

live:hamidbayanak_1