

API Urban Grocers

Warehouses

Warehouses - Almacén: "Big World"

La interacción se realiza a través de SOAP

POST

/big-world/wsdl

Parámetro

Campo	Tipo	Descripción
deliveryTime	number	Plazo de entrega previsto
products	Array	Lista de productos
id	number	Identificador del producto de la tabla product_model
quantity	number	Cantidad de producto

- Ejemplo de solicitud [#parameter-examples-Warehouses-BigWorld-2_0_0-0]

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://w
  <soap:Body>
    <tns:checkSupply>
      <tns:products>
        <tns:products>
          <id>1</id>
          <quantity>1</quantity>
        </tns:products>
        <tns:products>
          <id>4</id>
          <quantity>1</quantity>
```

```

        </tns:products>
      </tns:products>
      <tns:deliveryTime>10</tns:deliveryTime>
    </tns:checkSupply>
  </soap:Body>
</soap:Envelope>

```

- Respuesta: Ejemplo de respuesta [#success-examples-Warehouses-BigWorld-2_0_0-0]

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tns="WebServ
  <soap:Body>
    <tns:checkSupplyResponse>
      <tns:name>Big World</tns:name>
      <tns:workingHours>
        <start>5</start>
        <end>20</end>
      </tns:workingHours>
      <tns:price>9.5</tns:price>
      <tns:isWarehouseWorking>true</tns:isWarehouseWorking>
      <tns:isItemsAvailable>false</tns:isItemsAvailable>
      <tns:percentOfAvailableItems>33</tns:percentOfAvailableItems>
      <tns:availableProducts>
        <tns:availableProducts>
          <id>1</id>
          <quantity>1</quantity>
        </tns:availableProducts>
      </tns:availableProducts>
    </tns:checkSupplyResponse>
  </soap:Body>
</soap:Envelope>

```

Warehouses - Almacén: "Everything You Need"

POST

```
/everything-you-need/v1/calculate
```

- Ejemplos de encabezados [#header-examples-Warehouses-EverythingYouNeed-1_0_0-0]

```
{  
  "Content-Type": "application/json"  
}
```

Parámetro

Campo	Tipo	Descripción
deliveryTime	number	Plazo de entrega previsto
products	Array	Lista de productos
id	number	Identificador del producto de la tabla product_model
quantity	number	Cantidad de producto

- Ejemplo de solicitud [#parameter-examples-Warehouses-EverythingYouNeed-1_0_0-0]

```
{  
  "deliveryTime": 9,  
  "products": [  
    {  
      "id": 1,  
      "quantity": 1  
    },  
    {  
      "id": 4,  
      "quantity": 3  
    }  
  ]  
}
```

- Respuesta: Ejemplo de respuesta [#success-examples-Warehouses-EverythingYouNeed-1_0_0-0]

```
HTTP/1.1 200 OK  
{  
  "name": "Everything You Need",
```

```
"workingHours": { "start": 8, "end": 23 },
"price": 10,
"availableProducts": [
  { "id": 1, "quantity": 2 }
],
"isWarehouseWorking": true,
"isItemsAvailable": true,
"percentOfAvailableItems": 80
}
```

Warehouses - Almacén: "Food City"

Interacción a través de XML

POST

```
/food-city/calculate.xml
```

- Ejemplos de encabezados [#header-examples-Warehouses-Food_City-2_0_0-0]

```
{
  "Content-Type": "application/xml"
}
```

Parámetro

Campo	Tipo	Descripción
deliveryTime	number	Plazo de entrega previsto
products	Array	Lista de productos
id	number	Identificador del producto de la tabla product_model
quantity	number	Cantidad de producto

- Ejemplo de solicitud [#parameter-examples-Warehouses-Food_City-2_0_0-0]

```
<?xml version="1.0" encoding="UTF-8"?>
<InputModel>
```

```
<deliveryTime>9</deliveryTime>
<product id="5" quantity="2" />
<product id="4" quantity="1" />
</InputModel>
```

- Respuesta: Ejemplo de respuesta [#success-examples-Warehouses-Food_City-2_0_0-0]

```
<response name="Food City" price="8" isWarehouseWorking="true" isItemsAvailable="true" pe
  <working_hours>
    <start>8</start>
    <end>21</end>
  </working_hours>
  <available_products>
    <product id="5" quantity="6"/>
    <product id="4" quantity="12"/>
  </available_products>
</response>
```

Warehouses - Comprobar la cantidad de productos en stock

La versión SOAP de este punto final se llama - [SOAP] Comprobar la cantidad de productos en stock

POST

```
/api/v1/warehouses/amount
```

- [JSON] Ejemplos de encabezados [#header-examples-Warehouses-AmountWarehouses-1_0_0-0]
- [XML] Ejemplos de encabezados [#header-examples-Warehouses-AmountWarehouses-1_0_0-1]

```
{
  "Content-Type": "application/json"
}
```

```
{
  "Content-Type": "application/xml"
}
```

Parámetro

Campo	Tipo	Descripción
ids	number[]	Una matriz de identificadores de productos (después de id en la tabla product_model).
dataType opcional	string	<p>Formato de los datos de entrada. Puede tomar valores:</p> <ul style="list-style-type: none">"json" - El cuerpo de la solicitud se espera en formato JSON"xml" - El cuerpo de la solicitud se espera en formato XML <p>Valor por defecto: json</p>

- [JSON] Comprobamos la cantidad de productos [#parameter-examples-Warehouses-AmountWarehouses-1_0_0-0]
- [XML] Comprobamos la cantidad de productos [#parameter-examples-Warehouses-AmountWarehouses-1_0_0-1]

```
{
  "ids": [
    1,
    4,
    44
  ]
}
```

```
<root>
  <id>1</id>
  <id>4</id>
  <id>44</id>
</root>
```

- Respuesta: En qué almacén, qué hay disponible y cuánto [#success-examples-Warehouses-AmountWarehouses-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "Everything You Need": {
    "Refresco Sprite": 9,
    "Gourmet Palomitas Kernels": 6
  },
  "Food City": {
    "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": 3,
    "Refresco Sprite": 12
  },
}
```

```
"Big World": {
  "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": 1
},
"Fresh food": {
  "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": 3,
  "Refresco Sprite": 12
}
}
```

Warehouses - Comprobar la disponibilidad de productos en los almacenes

POST

```
/api/v1/warehouses/check
```

- Ejemplos de encabezados [#header-examples-Warehouses-CheckWarehouses-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
products	number[]	Una matriz de identificadores de productos and their quantities (Campos ``id` en la tabla product_model).

- Comprobamos la disponibilidad de productos [#parameter-examples-Warehouses-CheckWarehouses-1_0_0-0]

```
{
  "products": [
    {
      "id": 5,
      "quantity": 1
    },
    {
      "id": 4,
      "quantity": 5
    }
  ]
}
```

```
]
}
```

- Respuesta: En qué almacén, qué hay disponible y cuánto [#success-examples-Warehouses-CheckWarehouses-1_0_0-0]

HTTP/1.1 200 OK

```
{
  "Everything You Need": {
    "Refresco Sprite": true,
    "Gourmet Palomitas Kernels": true,
    "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": false
  },
  "Food City": {
    "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": true,
    "Refresco Sprite": true,
    "Gourmet Palomitas Kernels": false
  },
  "Big World": {
    "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": true,
    "Refresco Sprite": false,
    "Gourmet Palomitas Kernels": false
  },
  "Fresh food": {
    "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free": true,
    "Refresco Sprite": true,
    "Gourmet Palomitas Kernels": false
  }
}
```

Warehouses - Obtener la lista de almacenes

GET

/api/v1/warehouses

- Respuesta: Obtener la lista de almacenes con éxito [#success-examples-Warehouses-GetWarehouses-1_0_0-0]

HTTP/1.1 200 OK

```
[
  {
    "name": "Everything You Need",
```



```

        "workingHours": {
          "start": 7,
          "end": 23
        }
      },
    {
      "name": "Fresh food",
      "workingHours": {
        "start": 8,
        "end": 23
      }
    },
    {
      "name": "Food City",
      "workingHours": {
        "start": 8,
        "end": 21
      }
    },
    {
      "name": "Big World",
      "workingHours": {
        "start": 5,
        "end": 20
      }
    }
  ]

```

Warehouses - Wharehouse: "Fresh Food"

POST

```
/fresh-food/v2.0.1/ct
```

- Ejemplos de encabezados [#header-examples-Warehouses-FreshFood-2_0_1-0]

```

{
  "Content-Type": "application/json"
}

```

Parámetro

Campo	Tipo	Descripción
deliveryTime	number	Plazo de entrega previsto
products	Array	Lista de productos
id	number	Identificador del producto de la tabla product_model
quantity	number	Cantidad de producto

- Ejemplo de solicitud [#parameter-examples-Warehouses-FreshFood-2_0_1-0]

```
{
  "deliveryTime": 9,
  "products": [
    {
      "id": 1,
      "quantity": 1
    },
    {
      "id": 4,
      "quantity": 3
    }
  ]
}
```

- Respuesta: Ejemplo de respuesta [#success-examples-Warehouses-FreshFood-2_0_1-0]

```
HTTP/1.1 200 OK
{
  "name": "Fresh food",
  "workingHours": { "start": 8, "end": 23 },
  "price": 10,
  "availableProducts": [
    { "id": 1, "quantity": 2 }
  ],
  "isWarehouseWorking": true,
  "isItemsAvailable": true,
}
```

```
"percentOfAvailableItems": 80
```

```
}
```

Warehouses - [SOAP] Comprobar la cantidad de productos en stock

POST

```
/api/wsd1
```

Parámetro

Campo	Tipo	Descripción
ids	number[]	Una matriz de identificadores de productos (después de id en la tabla product_model).

- [XML] Comprobamos la cantidad de productos [#parameter-examples-Warehouses-AmountWarehousesSOAP-1_0_0-0]

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://w
  <soap:Body>
    <Request xmlns="WebServices.MainWsd1">
      <ids>1</ids>
      <ids>4</ids>
      <ids>44</ids>
    </Request>
  </soap:Body>
</soap:Envelope>
```

- Respuesta: En qué almacén, qué hay disponible y cuánto [#success-examples-Warehouses-AmountWarehousesSOAP-1_0_0-0]

```
HTTP/1.1 200 OK
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tns="WebServ
  <soap:Body>
    <Response>
      <name>Everything You Need</name>
      <products>
        <name>Refresco Sprite</name>
        <quantity>9</quantity>
```

```

    </products>
  </products>
    <name>Gourmet Palomitas Kernels</name>
    <quantity>6</quantity>
  </products>
</Response>
<Response>
  <name>Food City</name>
  <products>
    <name>Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free</name>
    <quantity>3</quantity>
  </products>
  <products>
    <name>Refresco Sprite</name>
    <quantity>12</quantity>
  </products>
</Response>
<Response>
  <name>Big World</name>
  <products>
    <name>Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free</name>
    <quantity>1</quantity>
  </products>
</Response>
<Response>
  <name>Fresh food</name>
  <products>
    <name>Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free</name>
    <quantity>3</quantity>
  </products>
  <products>
    <name>Refresco Sprite</name>
    <quantity>12</quantity>
  </products>
</Response>
</soap:Body>
</soap:Envelope>

```

Couriers - Comprobar los costes de envío

POST

```
/api/v1/couriers/check
```

- Ejemplos de encabezados [#header-examples-Couriers-CheckCouriers-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
products	number[]	Una matriz de identificadores de productos and their quantities (después de id en la tabla product_model).
deliveryTime	number	Plazo de entrega previsto (from 0 to 23)

- Comprobamos los costes de los diferentes servicios [#parameter-examples-Couriers-CheckCouriers-1_0_0-0]

```
{
  "products": [
    {
      "id": 1,
      "quantity": 3
    },
    {
      "id": 4,
      "quantity": 1
    },
    {
      "id": 9,
      "quantity": 3
    }
  ],
  "deliveryTime": 7
}
```

- Respuesta: Cuánto costaría cada uno de los servicios [#success-examples-Couriers-CheckCouriers-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "Fast Delivery": {
    "deliveryPrice": 0
  },
  "Speedy": {
    "deliveryPrice": 99
  },
  "Food Service": {
    "deliveryPrice": 0
  },
  "Order and Go": {
    "deliveryPrice": 99
  }
}
```

Couriers - Entrega: "Fast Delivery"

Error en los datos de entrada. La interacción se realiza mediante XML

POST

```
/fast-delivery/v3.1.1/calculate-delivery.xml
```

- Ejemplos de encabezados [#header-examples-Couriers-FastDelivery-3_1_1-0]

```
{
  "Content-Type": "application/xml"
}
```

Parámetro

Campo	Tipo	Descripción
productsCount	number	Número de productos en el pedido
productsWeight	number	Peso de los productos
deliveryTime	number	Plazo de entrega previsto

- Ejemplo de solicitud [#parameter-examples-Couriers-FastDelivery-3_1_1-0]

```
<InputModel>
  <productsCount>2</productsCount>
  <productsWeight>5.115</productsWeight>
  <deliveryTime>20</deliveryTime>
</InputModel>
```

- Respuesta: Ejemplo de respuesta [#success-examples-Couriers-FastDelivery-3_1_1-0]

```
HTTP/1.1 200 OK
<response name="Fast Delivery" isItPossibleToDeliver="true" hostDeliveryCost="43" clientD
  <toBeDeliveredTime>
    <min>25</min>
    <max>30</max>
  </toBeDeliveredTime>
</response>
```

Couriers - Entrega: "Food Service"

Interacción a través de SOAP

POST

/food-service/wSDL

Parámetro

Campo	Tipo	Descripción
productsCount	number	Número de productos en el pedido
productsWeight	number	Peso de los productos
deliveryTime	number	Plazo de entrega previsto

- Ejemplo de solicitud [#parameter-examples-Couriers-FoodService-1_0_0-0]

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://w
```

```
<soap:Body>
  <Request xmlns="WebServices.CourierWsd1">
    <productsCount>3</productsCount>
    <productsWeight>1.433</productsWeight>
    <deliveryTime>7</deliveryTime>
  </Request>
</soap:Body>
</soap:Envelope>
```

- Respuesta: Ejemplo de respuesta [#success-examples-Couriers-FoodService-1_0_0-0]

```
HTTP/1.1 200 OK
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:tns="WebServ
  <soap:Body>
    <Response xmlns="WebServices.CourierWsd1">
      <name>Food Service</name>
      <isItPossibleToDeliver>true</isItPossibleToDeliver>
      <hostDeliveryCost>29</hostDeliveryCost>
      <toBeDeliveredTime>
        <min>25</min>
        <max>30</max>
      </toBeDeliveredTime>
      <clientDeliveryCost>0</clientDeliveryCost>
    </Response>
  </soap:Body>
</soap:Envelope>
```

Couriers - Entrega: "Order and Go"

POST

```
/order-and-go/v1/delivery
```

- Ejemplos de encabezados [#header-examples-Couriers-OrderAndGo-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```


Parámetro

Campo	Tipo	Descripción
productsCount	number	Número de productos en el pedido
productsWeight	number	Peso de los productos
deliveryTime	number	Plazo de entrega previsto

- Ejemplo de solicitud [#parameter-examples-Couriers-OrderAndGo-1_0_0-0]

```
{
  "deliveryTime": 9,
  "productsCount": 10,
  "productsWeight": 11
}
```

- Respuesta: Ejemplo de respuesta [#success-examples-Couriers-OrderAndGo-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "name": "Order and Go",
  "clientDeliveryCost": 10,
  "toBeDeliveredTime": { "min": 10, "max": 20 },
  "hostDeliveryCost": 23,
  "isItPossibleToDeliver": true
}
```

Couriers - Entrega: "Speedy"

POST

```
/speedy/v1/calculate
```

- Ejemplos de encabezados [#header-examples-Couriers-Speedy-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
productsCount	number	Número de productos en el pedido
productsWeight	number	Peso de los productos
deliveryTime	number	Plazo de entrega previsto

- Ejemplo de solicitud [#parameter-examples-Couriers-Speedy-1_0_0-0]

```
{
  "deliveryTime": 9,
  "productsCount": 10,
  "productsWeight": 11
}
```

- Respuesta: Ejemplo de respuesta [#success-examples-Couriers-Speedy-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "name": "Speedy",
  "clientDeliveryCost": 10,
  "toBeDeliveredTime": { "min": 10, "max": 20 },
  "hostDeliveryCost": 23,
  "isItPossibleToDeliver": true
}
```

Couriers - Obtenga la lista de entregas

GET

```
/api/v1/couriers
```

- Respuesta: Recibir las entregas con éxito [#success-examples-Couriers-GetCouriers-1_0_0-0]

```
HTTP/1.1 200 OK
[
  {
```

```
[
  {
    "name": "Order and Go",
    "workingHours": {
      "start": 8,
      "end": 22
    }
  },
  {
    "name": "Speedy",
    "workingHours": {
      "start": 8,
      "end": 22
    }
  },
  {
    "name": "Fast Delivery",
    "workingHours": {
      "start": 7,
      "end": 21
    }
  },
  {
    "name": "Food Service",
    "workingHours": {
      "start": 6,
      "end": 20
    }
  }
]
```

Main.User

Main.User - Creación de cuenta

POST

/api/v1/users

- Ejemplos de encabezados [#header-examples-Main_User-CreateUser-1_0_0-0]

```
{  
  "Content-Type": "application/json"  
}
```

Parámetro

Campo	Tipo	Descripción
firstName	string	Nombre de usuario/a, escrito en el campo firstName de la tabla Users
phone	string	El número de teléfono del/de la usuario/a se escribe en el campo phone de la tabla Users
address	string	La dirección del/de la usuario/a se escribe en el campo address de la tabla Users
email opcional	string	Dirección de correo electrónico, escrita en el campo email de la tabla Users
comment opcional	string	Comentario de usuario/a, escrito en el campo comment de la tabla Users

- Conjunto mínimo de datos [#parameter-examples-Main_User-CreateUser-1_0_0-0]
- Información completa sobre el/la usuario/a [#parameter-examples-Main_User-CreateUser-1_0_0-1]

```
{  
  "firstName": "Max",  
  "phone": "+10005553535",  
  "address": "8042 Lancaster Ave.Hamburg, NY"  
}
```

```
{  
  "firstName": "Max",  
  "email": "max@example.com",  
  "phone": "+10005553535",  
  "comment": "Cuidado con el perro",  
  "address": "8042 Lancaster Ave.Hamburg, NY"  
}
```

- La cuenta de usuario se ha creado correctamente [#success-examples-Main_User-CreateUser-1_0_0-0]

HTTP/1.1 201 Creado

```
{
  authToken: 'jknnFApafP4awfAIFfafam2fma'
}
```

- Error en los datos de entrada #1 [#error-examples-Main_User-CreateUser-1_0_0-0]
- Error en los datos de entrada #2 [#error-examples-Main_User-CreateUser-1_0_0-1]
- Error en los datos de entrada #3 [#error-examples-Main_User-CreateUser-1_0_0-2]
- Error en los datos de entrada #4 [#error-examples-Main_User-CreateUser-1_0_0-3]

HTTP/1.1 400 Bad request.

```
{
  "code": 400,
  "message": "No se han aprobado todos los parámetros requeridos. Parámetros requeridos"
}
```

HTTP/1.1 400 Bad request.

```
{
  "code": 400,
  "message": "Has introducido un nombre de usuario no válido. El nombre solo puede cont"
}
```

HTTP/1.1 400 Bad request.

```
{
  "code": 400,
  "message": "Has introducido un número de teléfono de usuario no válido. El número de"
}
```

HTTP/1.1 400 Bad request.

```
{
  "code": 400,
  "message": "Has introducido una dirección no válida. La dirección solo puede contener"
}
```

Main.Products

Main.Products - Búsqueda de cestas por producto

Punto final para buscar cestas que contengan los productos especificados

POST

```
/api/v1/products/kits
```

- Ejemplos de encabezados [#header-examples-Main_Products-GetKitsByProducts-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
ids	number[]	Una matriz de identificadores de productos from the table product_model.

- Buscar cestas que contengan el producto 1 o 2 o 3: [#parameter-examples-Main_Products-GetKitsByProducts-1_0_0-0]

```
{
  "ids": [
    1,
    2,
    3
  ]
}
```

- Respuesta: Recibir las cestas con éxito [#success-examples-Main_Products-GetKitsByProducts-1_0_0-0]

```
HTTP/1.1 200 OK
[
  {
    "id": 1,
    "name": "Para pícnicos",
    "productsList": [
```

```
{
  "id": 1,
  "quantity": 1
},
{
  "id": 2,
  "quantity": 1
},
{
  "id": 3,
  "quantity": 1
},
{
  "id": 4,
  "quantity": 1
},
{
  "id": 5,
  "quantity": 1
},
{
  "id": 6,
  "quantity": 1
},
{
  "id": 7,
  "quantity": 1
},
{
  "id": 8,
  "quantity": 1
},
{
  "id": 9,
  "quantity": 1
},
{
  "id": 10,
  "quantity": 1
},
{
  "id": 11,
  "quantity": 1
},
},
```

```
{
  "id": 12,
  "quantity": 1
},
{
  "id": 13,
  "quantity": 1
},
{
  "id": 14,
  "quantity": 1
},
{
  "id": 15,
  "quantity": 1
},
{
  "id": 16,
  "quantity": 1
},
{
  "id": 17,
  "quantity": 1
},
{
  "id": 18,
  "quantity": 1
},
{
  "id": 19,
  "quantity": 1
},
{
  "id": 20,
  "quantity": 1
},
{
  "id": 21,
  "quantity": 1
},
{
  "id": 22,
  "quantity": 1
},
```



```
{
  "id": 23,
  "quantity": 1
},
{
  "id": 24,
  "quantity": 1
},
{
  "id": 25,
  "quantity": 1
},
{
  "id": 26,
  "quantity": 1
},
{
  "id": 27,
  "quantity": 1
},
{
  "id": 28,
  "quantity": 1
},
{
  "id": 29,
  "quantity": 1
},
{
  "id": 30,
  "quantity": 1
},
{
  "id": 31,
  "quantity": 1
},
{
  "id": 32,
  "quantity": 1
},
{
  "id": 33,
  "quantity": 1
},
```

```
{
  {
    "id": 34,
    "quantity": 1
  },
  {
    "id": 35,
    "quantity": 1
  },
  {
    "id": 36,
    "quantity": 1
  },
  {
    "id": 37,
    "quantity": 1
  },
  {
    "id": 38,
    "quantity": 1
  },
  {
    "id": 39,
    "quantity": 1
  }
],
"productsCount": 39
},
{
  "id": 2,
  "name": "Para películas y series",
  "productsList": [
    {
      "id": 7,
      "quantity": 1
    },
    {
      "id": 8,
      "quantity": 1
    },
    {
      "id": 9,
      "quantity": 1
    },
  ],
  {
```

```
      "id": 40,  
      "quantity": 1  
    },  
    {  
      "id": 41,  
      "quantity": 1  
    },  
    {  
      "id": 42,  
      "quantity": 1  
    },  
    {  
      "id": 43,  
      "quantity": 1  
    },  
    {  
      "id": 44,  
      "quantity": 1  
    },  
    {  
      "id": 45,  
      "quantity": 1  
    },  
    {  
      "id": 46,  
      "quantity": 1  
    },  
    {  
      "id": 47,  
      "quantity": 1  
    },  
    {  
      "id": 2,  
      "quantity": 1  
    },  
    {  
      "id": 3,  
      "quantity": 1  
    },  
    {  
      "id": 4,  
      "quantity": 1  
    },  
    {
```

```
      "id": 48,  
      "quantity": 1  
    },  
    {  
      "id": 49,  
      "quantity": 1  
    },  
    {  
      "id": 50,  
      "quantity": 1  
    },  
    {  
      "id": 51,  
      "quantity": 1  
    },  
    {  
      "id": 52,  
      "quantity": 1  
    },  
    {  
      "id": 53,  
      "quantity": 1  
    },  
    {  
      "id": 54,  
      "quantity": 1  
    },  
    {  
      "id": 55,  
      "quantity": 1  
    },  
    {  
      "id": 56,  
      "quantity": 1  
    },  
    {  
      "id": 57,  
      "quantity": 1  
    },  
    {  
      "id": 58,  
      "quantity": 1  
    },  
    {
```

```
        "id": 59,
        "quantity": 1
    },
    {
        "id": 60,
        "quantity": 1
    }
],
"productsCount": 27
}
```

```
]
```

Main.Products - Cambio del precio del producto

PUT

```
/api/v1/products/:id
```

- Ejemplos de encabezados [#header-examples-Main_Products-UpdateProduct-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
id	number	El id de la cesta en la tabla product_model. Se pasa en la URL
price	opcional string	price del producto. Se pasa en el texto de la solicitud.

- Cambio del producto /api/v1/products/7 [#parameter-examples-Main_Products-UpdateProduct-1_0_0-0]

```
{
  "price": 175
}
```

- Respuesta: El kit ha sido eliminado con éxito. [#success-examples-Main_Products-UpdateProduct-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "ok": true
}
```

- Error: Producto no encontrado [#error-examples-Main_Products-UpdateProduct-1_0_0-0]

```
HTTP/1.1 404 Not found.
{
  "code": 404,
  "message": "Not found"
}
```

Main.Kits

Main.Kits - Añadir productos a la cesta:

POST

```
/api/v1/kits/:id/products
```

- Ejemplos de encabezados [#header-examples-Main_Kits-AddProductsToKit-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
id	number	El id de cesta en la tabla kit_model. Se pasa en la URL
productsList	Array	Una lista de productos que se agregarán al kit. La lista contiene los ID de los artículos y sus cantidades. Debe enviarse en el cuerpo de la solicitud.

- Ejemplo de cuerpo de solicitud para agregar artículos al kit [#parameter-examples-Main_Kits-AddProductsToKit-1_0_0-0]

```
{
  "productsList": [
    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 6,
      "quantity": 2
    }
  ]
}
```

- Respuesta: Completar con éxito la cesta de productos [#success-examples-Main_Kits-AddProductsToKit-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "id": 2,
  "name": "Mi conjunto del fin de semana",
  "productsList": [
    {
      "id": 1,
      "name": "Caviar rojo",
      "price": 45,
      "weight": 5,
      "units": "kg",
      "quantity": 2
    },
    {
      "id": 5,
      "name": "Baguette",
      "price": 15,
      "weight": 1,
      "units": "kg",
      "quantity": 2
    }
  ],
  "productsCount": 4
}
```

- Error: No se han encontrado conjuntos coincidentes que coincidan [#error-examples-Main_Kits-AddProductsToKit-1_0_0-0]

```
HTTP/1.1 404 Not found.
{
  "code": 404,
  "message": "Not found"
}
```

Main.Kits - Cambio de la cesta

PUT

/api/v1/kits/:id

- Ejemplos de encabezados [#header-examples-Main_Kits-UpdateKit-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
id	number	El id de cesta en la tabla kit_model. Se pasa en la URL
name <small>opcional</small>	string	name de la cesta. Se pasa en el texto de la solicitud.
productsList <small>opcional</small>	Array	productsList Una lista de productos que se agregarán al kit. La lista contiene los ID de los artículos y sus cantidades. Debe enviarse en el cuerpo de la solicitud.

- Cambio de la cesta /api/v1/kits/7 [#parameter-examples-Main_Kits-UpdateKit-1_0_0-0]

```
{
  "name": "Mi cesta modificada",
  "productsList": [
    {
      "id": 1,
      "quantity": 4
    }
  ]
}
```



```

    },
    {
      "id": 5,
      "quantity": 2
    },
    {
      "id": 3,
      "quantity": 1
    },
    {
      "id": 4,
      "quantity": 1
    }
  ]
}

```

- Respuesta: El kit ha sido eliminado con éxito. [#success-examples-Main_Kits-UpdateKit-1_0_0-0]

```

HTTP/1.1 200 OK
{
  "ok": true
}

```

- Error: No se han encontrado conjuntos coincidentes que coincidan [#error-examples-Main_Kits-UpdateKit-1_0_0-0]

```

HTTP/1.1 404 Not found.
{
  "code": 404,
  "message": "Not found"
}

```

Main.Kits - Crear un kit

Punto final para crear una cesta de una tarjeta específica O de usuario.

- Es obligatorio pasar el encabezado Authorisation O el parámetro cardId, para crear la cesta
- Si se recibe una solicitud con un encabezado Authorisation que contenga el authToken de un/a usuario/a en particular - se creará la cesta de este/a usuario/a.
- Si se recibe el parámetro cardId, se creará una cesta dentro de la tarjeta correspondiente
- Si no se pasa ninguno de los parámetros, se devolverá un error.

- Cuando se pasan ambos parámetros, Authorization es la prioridad

POST

/api/v1/kits

Header

Campo		Tipo	Descripción
Authorization	opcional	string	Encabezado de autorización en formato Bearer {authToken}. Cuando se pasa - se devuelven todos las cestas creadas por el/la usuario/a.
Content-Type	opcional	string	Valor por defecto: application/json

- Ejemplos de encabezados [#header-examples-Main_Kits-CreateKit-1_0_0-0]
- Recepción de las cestas creadas por el/la usuario/a [#header-examples-Main_Kits-CreateKit-1_0_0-1]

```
{
  "Content-Type": "application/json"
}
```

```
{
  "Content-Type": "application/json",
  "Authorization": "Bearer jknnFApafP4awfAIFfafam2fma"
}
```

Parámetro

Campo		Tipo	Descripción
cardId	opcional	number	El id de la tarjeta en la tabla card_model. Cuando se pasa - se creará un conjunto incluido en esta tarjeta
name		string	El nombre de la cesta, que será escrito en el campo correspondiente de la tabla kit_model.

- Respuesta: El conjunto ha sido creado con éxito [#success-examples-Main_Kits-CreateKit-1_0_0-0]

```
HTTP/1.1 201 Creado
{
```

```
"name": "Mi conjunto",
"card": {
  "id": 1,
  "name": "Para la situación"
},
"productsList": null,
"id": 7,
"productsCount": 0
}
```

- Error: No se ha transmitido ninguno de los parámetros [#error-examples-Main_Kits-CreateKit-1_0_0-0]
- Error: Validación del nombre [#error-examples-Main_Kits-CreateKit-1_0_0-1]

```
HTTP/1.1 400 Bad request.
{
  "code": 400,
  "message": "No se han aprobado todos los parámetros requeridos"
}
```

```
HTTP/1.1 400 Bad request.
{
  "code": 400,
  "message": "El nombre debe contener sólo letras latino, un espacio y un guión. De
}
```

Main.Kits - Eliminación del equipo

DELETE

```
/api/v1/kits/:id
```

Parámetro

Campo	Tipo	Descripción
id	number	El id de cesta en la tabla kit_model. Se pasa en la URL

- Eliminación de un kit con id=7 [#parameter-examples-Main_Kits-RemoveKit-1_0_0-0]

```
/api/v1/kits/7
```

- Respuesta: El kit ha sido eliminado con éxito. [#success-examples-Main_Kits-RemoveKit-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "ok": true
}
```

Main.Kits - Recibir todas las cestas

Punto final para recuperar todas las cestas dentro de una tarjeta específica O las cestas creadas por el/la usuario/a.

- Es obligatorio pasar el encabezado Authorisation O el parámetro cardId, para recibir las cestas
- Si se recibe una solicitud con un encabezado Authorisation que contenga el authToken de un/a usuario/a en particular - se devolverán las cestas creadas por ese/a usuario/a.
- Si se recibe el parámetro de consulta cardId, se devolverán las cestas relativas a la tarjeta.
- Si no se pasa ninguno de los parámetros, se devolverá un error.
- Cuando se pasan ambos parámetros, Authorization es la prioridad

GET

/api/v1/kits

Header

Campo	Tipo	Descripción
Authorization <small>opcional</small>	string	Encabezado de autorización en formato Bearer {authToken}. Cuando se pasa - se devuelven todos las cestas creadas por el/la usuario/a.

- Recepción de las cestas creadas por el/la usuario/a [#header-examples-Main_Kits-GetKitsList-1_0_0-0]

```
{
  "Authorization": "Bearer jknnFApafP4awfAIFfafam2fma"
}
```

Parámetro

Campo	Tipo	Descripción
cardId	opcional	number
		El id de la tarjeta en la tabla card_model. Cuando se pasa - se devuelven todos las cestas incluidas en la tarjeta.

- Recibir las cestas dentro de una tarjeta with id = 1 [#parameter-examples-Main_Kits-GetKitsList-1_0_0-0]

```
/api/v1/kits?cardId=1
```

- Respuesta: Recibir las cestas con éxito [#success-examples-Main_Kits-GetKitsList-1_0_0-0]

```
HTTP/1.1 200 OK
[
  {
    "id": 1,
    "name": "Para pícnicos",
    "productsList": [
      {
        "id": 1,
        "name": "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free",
        "price": 149,
        "weight": 473,
        "units": "ml",
        "quantity": 1
      },
      {
        "id": 2,
        "name": "Refresco Mountain Dew",
        "price": 89,
        "weight": 1,
        "units": "l",
        "quantity": 1
      },
      {
        "id": 3,
        "name": "Refresco Pepsi",
        "price": 109,
        "weight": 1,
        "units": "l",
        "quantity": 1
      },
      {
        "id": 4,
```

```
    "name": "Refresco Sprite",
    "price": 79,
    "weight": 900,
    "units": "ml",
    "quantity": 1
  },
  {
    "id": 5,
    "name": "Jugo Fruit Power: Lichi",
    "price": 349,
    "weight": 900,
    "units": "ml",
    "quantity": 1
  },
  {
    "id": 6,
    "name": "Baguette",
    "price": 69,
    "weight": 160,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 7,
    "name": "Papas fritas: Saladas Clásicas",
    "price": 119,
    "weight": 150,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 8,
    "name": "Papas fritas: Paprika",
    "price": 199,
    "weight": 150,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 9,
    "name": "Papas Fritas: Chile",
    "price": 209,
    "weight": 100,
    "units": "g",
```

```
    "quantity": 1
  },
  {
    "id": 10,
    "name": "Pita",
    "price": 99,
    "weight": 300,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 11,
    "name": "Pan multigrano",
    "price": 149,
    "weight": 400,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 12,
    "name": "Pan de ajo",
    "price": 75,
    "weight": 225,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 13,
    "name": "Mini chapata",
    "price": 45,
    "weight": 107,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 14,
    "name": "Brioche",
    "price": 79,
    "weight": 79,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 15,
```

```
    "name": "Salchichas vienesas",
    "price": 179,
    "weight": 350,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 16,
    "name": "Salchichas de pollo",
    "price": 173,
    "weight": 200,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 17,
    "name": "Chorizo",
    "price": 269,
    "weight": 420,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 18,
    "name": "Salchichón",
    "price": 175,
    "weight": 464,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 19,
    "name": "Salami",
    "price": 99,
    "weight": 190,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 20,
    "name": "Bacon",
    "price": 269,
    "weight": 500,
    "units": "g",
```



```
        "quantity": 1
    },
    {
        "id": 21,
        "name": "Pastrami",
        "price": 143,
        "weight": 200,
        "units": "g",
        "quantity": 1
    },
    {
        "id": 22,
        "name": "Pepperoni",
        "price": 329,
        "weight": 500,
        "units": "g",
        "quantity": 1
    },
    {
        "id": 23,
        "name": "Anti-Bacterial Hand Sanitizer",
        "price": 139,
        "weight": 50,
        "units": "ml",
        "quantity": 1
    },
    {
        "id": 24,
        "name": "Paper Napkin - Large",
        "price": 99,
        "weight": 60,
        "units": "pzas",
        "quantity": 1
    },
    {
        "id": 25,
        "name": "Anti Bacterial Disinfectant Sanitising Wipes - Alcohol Free",
        "price": 199,
        "weight": 48,
        "units": "pzas",
        "quantity": 1
    },
    {
        "id": 26,
```

```
    "name": "SMART Mosquito Repellent",
    "price": 349,
    "weight": 100,
    "units": "ml",
    "quantity": 1
  },
  {
    "id": 27,
    "name": "Manzana verde",
    "price": 169,
    "weight": 500,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 28,
    "name": "Orange",
    "price": 146,
    "weight": 500,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 29,
    "name": "Cherry",
    "price": 249,
    "weight": 500,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 30,
    "name": "Raspberry",
    "price": 239,
    "weight": 250,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 31,
    "name": "Blackberry",
    "price": 125,
    "weight": 200,
    "units": "g",
```

```
    "quantity": 1
  },
  {
    "id": 32,
    "name": "Lima",
    "price": 59,
    "weight": 1,
    "units": "pzas",
    "quantity": 1
  },
  {
    "id": 33,
    "name": "Red Grapes",
    "price": 269,
    "weight": 500,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 34,
    "name": "Green Grapes",
    "price": 179,
    "weight": 500,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 35,
    "name": "Banana",
    "price": 99,
    "weight": 1000,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 36,
    "name": "Pear",
    "price": 199,
    "weight": 650,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 37,
```

```
        "name": "Blueberry",
        "price": 199,
        "weight": 125,
        "units": "g",
        "quantity": 1
    },
    {
        "id": 38,
        "name": "Pineapple",
        "price": 385,
        "weight": 1,
        "units": "pzas",
        "quantity": 1
    },
    {
        "id": 39,
        "name": "Pear",
        "price": 145,
        "weight": 800,
        "units": "g",
        "quantity": 1
    }
],
"productsCount": 39
},
{
    "id": 2,
    "name": "Para películas y series",
    "productsList": [
        {
            "id": 2,
            "name": "Refresco Mountain Dew",
            "price": 89,
            "weight": 1,
            "units": "l",
            "quantity": 1
        },
        {
            "id": 3,
            "name": "Refresco Pepsi",
            "price": 109,
            "weight": 1,
            "units": "l",
            "quantity": 1
        }
    ]
}
```

```
    },
    {
      "id": 4,
      "name": "Refresco Sprite",
      "price": 79,
      "weight": 900,
      "units": "ml",
      "quantity": 1
    },
    {
      "id": 7,
      "name": "Papas fritas: Saladas Clásicas",
      "price": 119,
      "weight": 150,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 8,
      "name": "Papas fritas: Paprika",
      "price": 199,
      "weight": 150,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 9,
      "name": "Papas Fritas: Chile",
      "price": 209,
      "weight": 100,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 43,
      "name": "Classic Hummus Dip & Spread",
      "price": 99,
      "weight": 150,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 44,
      "name": "Gourmet Palomitas Kernels",
```

```
    "price": 109,  
    "weight": 60,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 45,  
    "name": "Caramel Palomitas",  
    "price": 99,  
    "weight": 150,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 46,  
    "name": "Fruit Beer Non-Alcoholic",  
    "price": 249,  
    "weight": 250,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 47,  
    "name": "Stella Artois Beer Non-Alcoholic",  
    "price": 179,  
    "weight": 60,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 48,  
    "name": "100% Belgian Dark Chocolate - with Apricot",  
    "price": 89,  
    "weight": 45,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 49,  
    "name": "100% Belgian Dark Chocolate - Raisins",  
    "price": 69,  
    "weight": 450,  
    "units": "ml",  
    "quantity": 1  
  }
```

```
},
{
  "id": 50,
  "name": "100% Belgian Milk Chocolate - Sea Salt",
  "price": 109,
  "weight": 450,
  "units": "ml",
  "quantity": 1
},
{
  "id": 54,
  "name": "Dairy Rich Chocolate Bar - Fruit & Nut",
  "price": 139,
  "weight": 100,
  "units": "g",
  "quantity": 1
},
{
  "id": 55,
  "name": "Peanut Butter Nutty Cookies",
  "price": 119,
  "weight": 97,
  "units": "g",
  "quantity": 1
},
{
  "id": 56,
  "name": "Cookies - Orange Splash",
  "price": 89,
  "weight": 85,
  "units": "g",
  "quantity": 1
},
{
  "id": 57,
  "name": "Choco Nut Cookies",
  "price": 159,
  "weight": 160,
  "units": "g",
  "quantity": 1
},
{
  "id": 58,
  "name": "Milk Cookies",
```

```
    "price": 239,  
    "weight": 100,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 59,  
    "name": "Sandwich Biscuits",  
    "price": 119,  
    "weight": 100,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 60,  
    "name": "Chocobakes Choc Filled Cookies",  
    "price": 139,  
    "weight": 80,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 61,  
    "name": "Baguette French Recipe",  
    "price": 179,  
    "weight": 171,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 62,  
    "name": "Queso Croissant",  
    "price": 119,  
    "weight": 50,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 63,  
    "name": "French Almond Croissant",  
    "price": 119,  
    "weight": 228,  
    "units": "g",  
    "quantity": 1  
  }
```



```
    },
    {
      "id": 64,
      "name": "Chocolate Croissant",
      "price": 149,
      "weight": 168,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 65,
      "name": "Smoked Salmon Croissant",
      "price": 59,
      "weight": 100,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 66,
      "name": "Belgian Chocolate Cake Mix",
      "price": 219,
      "weight": 250,
      "units": "g",
      "quantity": 1
    }
  ],
  "productsCount": 27
},
{
  "id": 3,
  "name": "Sabores de París",
  "productsList": [
    {
      "id": 54,
      "name": "Dairy Rich Chocolate Bar - Fruit & Nut",
      "price": 139,
      "weight": 100,
      "units": "g",
      "quantity": 1
    },
    {
      "id": 58,
      "name": "Milk Cookies",
      "price": 239,
```

```
    "weight": 100,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 67,  
    "name": "Baguette French Recipe",  
    "price": 89,  
    "weight": 200,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 68,  
    "name": "Queso Croissant",  
    "price": 79,  
    "weight": 75,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 69,  
    "name": "French Almond Croissant",  
    "price": 129,  
    "weight": 120,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 70,  
    "name": "Chocolate Croissant",  
    "price": 104,  
    "weight": 96,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 71,  
    "name": "Smoked Salmon Croissant",  
    "price": 119,  
    "weight": 90,  
    "units": "g",  
    "quantity": 1  
  },  
}
```

```
{
  "id": 72,
  "name": "Belgian Chocolate Cake Mix",
  "price": 359,
  "weight": 125,
  "units": "g",
  "quantity": 1
},
{
  "id": 73,
  "name": "Queso para untar: Queso crema",
  "price": 79,
  "weight": 130,
  "units": "g",
  "quantity": 1
},
{
  "id": 74,
  "name": "Queso Slices",
  "price": 239,
  "weight": 400,
  "units": "g",
  "quantity": 1
},
{
  "id": 75,
  "name": "Queso Spread - Roasted Garlic",
  "price": 129,
  "weight": 200,
  "units": "g",
  "quantity": 1
},
{
  "id": 76,
  "name": "Spicy Queso Straw",
  "price": 229,
  "weight": 250,
  "units": "g",
  "quantity": 1
},
{
  "id": 77,
  "name": "Rebanadas de queso procesado",
  "price": 220,
```

```
    "weight": 250,  
    "units": "",  
    "quantity": 1  
  },  
  {  
    "id": 78,  
    "name": "Cubos de queso procesado",  
    "price": 399,  
    "weight": 200,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 79,  
    "name": "Soyabean Tempeh Cubes",  
    "price": 209,  
    "weight": 300,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 82,  
    "name": "Center Filled Dark Chocolate Cookies",  
    "price": 129,  
    "weight": 100,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 83,  
    "name": "Enrobed Cinnamon Milk Chocolate Dipped Cookies",  
    "price": 225,  
    "weight": 100,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 84,  
    "name": "Oreo & Crème Frozen Dessert",  
    "price": 339,  
    "weight": 300,  
    "units": "ml",  
    "quantity": 1  
  },  
}
```

```

    {
      "id": 85,
      "name": "Chocorich Eclairs Chocolate - Assorted",
      "price": 429,
      "weight": 200,
      "units": "g",
      "quantity": 1
    }
  ],
  "productsCount": 19
}
]

```

- Error: No se ha transmitido ninguno de los parámetros [#error-examples-Main_Kits-GetKitsList-1_0_0-0]
- Error: No se han encontrado conjuntos coincidentes que coincidan [#error-examples-Main_Kits-GetKitsList-1_0_0-1]

HTTP/1.1 400 Bad request.

```

{
  "code": 400,
  "message": "No se han aprobado todos los parámetros requeridos"
}

```

HTTP/1.1 404 Not found.

```

{
  "code": 404,
  "message": "Not found"
}

```

Main.Kits - Recibir una cesta por el nombre

GET

```

/api/v1/kits/search

```

Parámetro

Campo	Tipo	Descripción
name	string	name de la cesta en la tabla kit_model.

- Recibir una cesta con un nombre "Sabores de París" [#parameter-examples-Main_Kits-GetKitByName-1_0_0-0]

```
/api/v1/kits/search?name=Sabores%20de%20París
```

- Respuesta: El conjunto ha sido recibido con éxito. [#success-examples-Main_Kits-GetKitByName-1_0_0-0]

HTTP/1.1 200 OK

```
{
  "id": 3,
  "name": "Sabores de París",
  "productsList": [
    {
      "id": 54,
      "name": "Dairy Rich Chocolate Bar - Fruit & Nut",
      "price": 139,
      "weight": 100,
      "units": "g",
      "quantity": 6
    },
    {
      "id": 58,
      "name": "Milk Cookies",
      "price": 239,
      "weight": 100,
      "units": "g",
      "quantity": 8
    },
    {
      "id": 67,
      "name": "Baguette French Recipe",
      "price": 89,
      "weight": 200,
      "units": "g",
      "quantity": 4
    },
    {
      "id": 68,
      "name": "Queso Croissant",
      "price": 79,
      "weight": 75,
      "units": "g",
      "quantity": 1
    }
  ]
}
```

```
    "id": 69,  
    "name": "French Almond Croissant",  
    "price": 129,  
    "weight": 120,  
    "units": "g",  
    "quantity": 7  
  },  
  {  
    "id": 70,  
    "name": "Chocolate Croissant",  
    "price": 104,  
    "weight": 96,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 71,  
    "name": "Smoked Salmon Croissant",  
    "price": 119,  
    "weight": 90,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 72,  
    "name": "Belgian Chocolate Cake Mix",  
    "price": 359,  
    "weight": 125,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 73,  
    "name": "Queso para untar: Queso crema",  
    "price": 79,  
    "weight": 130,  
    "units": "g",  
    "quantity": 1  
  },  
  {  
    "id": 74,  
    "name": "Queso Slices",  
    "price": 239,  
    "weight": 400,
```

```
    "units": "g",
    "quantity": 1
  },
  {
    "id": 75,
    "name": "Queso Spread - Roasted Garlic",
    "price": 129,
    "weight": 200,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 76,
    "name": "Spicy Queso Straw",
    "price": 229,
    "weight": 250,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 77,
    "name": "Rebanadas de queso procesado",
    "price": 220,
    "weight": 250,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 78,
    "name": "Cubos de queso procesado",
    "price": 399,
    "weight": 200,
    "units": "g",
    "quantity": 1
  },
  {
    "id": 79,
    "name": "Soyabean Tempeh Cubes",
    "price": 209,
    "weight": 300,
    "units": "g",
    "quantity": 1
  },
  {

```



```

        "id": 82,
        "name": "Center Filled Dark Chocolate Cookies",
        "price": 129,
        "weight": 100,
        "units": "g",
        "quantity": 1
    },
    {
        "id": 83,
        "name": "Enrobed Cinnamon Milk Chocolate Dipped Cookies",
        "price": 225,
        "weight": 100,
        "units": "g",
        "quantity": 1
    },
    {
        "id": 84,
        "name": "Oreo & Crème Frozen Dessert",
        "price": 339,
        "weight": 300,
        "units": "ml",
        "quantity": 1
    },
    {
        "id": 85,
        "name": "Chocorich Eclairs Chocolate - Assorted",
        "price": 429,
        "weight": 200,
        "units": "g",
        "quantity": 1
    }
],
"productsCount": 19
}

```

- Error: No se han encontrado conjuntos coincidentes que coincidan [#error-examples-Main_Kits-GetKitByName-1_0_0-0]

HTTP/1.1 404 Not found.

```

{
    "code": 404,

```

```
"message": "Not found"
```

```
}
```

Main.Basket

Main.Basket - Añadir productos a la cesta

PUT

```
/api/v1/orders/:id
```

- Ejemplos de encabezados [#header-examples-Main_Basket-UpdateBasket-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
id	number	id de la cesta en la tabla order_model. Se pasa en la URL
productsList	Array	Una lista de productos que se agregarán al kit. La lista contiene los ID de los artículos y sus cantidades. Debe enviarse en el cuerpo de la solicitud.

- Añadir productos a la cesta [#parameter-examples-Main_Basket-UpdateBasket-1_0_0-0]

```
{
  "productsList": [
    {
      "id": 1,
      "quantity": 4
    },
    {
      "id": 5,
      "quantity": 2
    }
  ]
}
```

```

    },
    {
      "id": 3,
      "quantity": 1
    },
    {
      "id": 4,
      "quantity": 1
    }
  ]
}

```

- Respuesta: Los artículos se han añadido a la cesta con éxito [#success-examples-Main_Basket-UpdateBasket-1_0_0-0]

```

HTTP/1.1 200 OK
{
  "productsList": [
    {
      "id": 1,
      "quantity": 10
    },
    {
      "id": 5,
      "quantity": 10
    },
    {
      "id": 3,
      "quantity": 9
    },
    {
      "id": 4,
      "quantity": 5
    }
  ],
  "status": 0,
  "deliveryPriceOur": 30,
  "deliveryTime": "25~30",
  "courierService": "Fast Delivery",
  "deliveryPrice": 0,
  "wareHouse": "Fresh food",
  "userId": 1,
  "id": 5,
  "productsCost": 75,

```

```
}
  "finalCCost": 174
}
```

- Error: Cesta no encontrada [#error-examples-Main_Basket-UpdateBasket-1_0_0-0]
- Error: No hay un almacén para procesar tu pedido [#error-examples-Main_Basket-UpdateBasket-1_0_0-1]

```
HTTP/1.1 404 Not found.
{
  "code": 404,
  "message": "Not found"
}
```

```
HTTP/1.1 409 Conflict.
{
  "code": 409,
  "message": "No hay un almacén para procesar tu pedido"
}
```

Main.Basket - Crear una cesta

POST

```
/api/v1/orders
```

- Ejemplos de encabezados [#header-examples-Main_Basket-CreateBasket-1_0_0-0]

```
{
  "Content-Type": "application/json"
}
```

Parámetro

Campo	Tipo	Descripción
productsList	Array	Una lista de productos que se agregarán al kit. La lista contiene los ID de los artículos y sus cantidades. Debe enviarse en el cuerpo de la solicitud.
deliveryTime opcional	number	Plazo de entrega previsto. Si no se pasa ningún valor, el sistema utiliza la hora actual

- Crear una cesta [#parameter-examples-Main_Basket-CreateBasket-1_0_0-0]

```
{
  "productsList": [
    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 5,
      "quantity": 2
    },
    {
      "id": 3,
      "quantity": 1
    }
  ]
}
```

- Respuesta: La cesta fue creada con éxito [#success-examples-Main_Basket-CreateBasket-1_0_0-0]

HTTP/1.1 200 OK

```
{
  "productsList": [
    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 5,
      "quantity": 2
    },
    {
      "id": 3,
      "quantity": 1
    }
  ],
  "status": 0,
  "deliveryPriceOur": 30,
  "deliveryTime": "25~30",
  "courierService": "Fast Delivery",
  "deliveryPrice": 99,
  "wareHouse": "Fresh food",
  "userId": 1,
  "id": 5,
}
```

```
"productsCost": 75,  
"finalCCost": 174  
}
```

- Error: No se han aprobado todos los parámetros requeridos [#error-examples-Main_Basket-CreateBasket-1_0_0-0]
- Error: No hay un almacén para procesar tu pedido [#error-examples-Main_Basket-CreateBasket-1_0_0-1]

HTTP/1.1 400 Bad request.

```
{  
  "code": 400,  
  "message": "No se han aprobado todos los parámetros requeridos"  
}
```

HTTP/1.1 409 Conflict.

```
{  
  "code": 409,  
  "message": "No hay un almacén para procesar tu pedido"  
}
```

Main.Basket - Eliminar la cesta

DELETE

/api/v1/orders/:id

Parámetro

Campo	Tipo	Descripción
id	number	id de la cesta en la tabla order_model. Se pasa en la URL

- Respuesta: Eliminar una cesta con éxito [#success-examples-Main_Basket-RemoveBasket-1_0_0-0]

HTTP/1.1 200 OK

```
{  
  "ok": true  
}
```

Main.Basket - Hacer el pedido

PUT

/api/v1/orders/:id/complete

Parámetro

Campo	Tipo	Descripción
id	number	id de la cesta en la tabla order_model. Se pasa en la URL

- Respuesta: Los artículos se han añadido a la cesta con éxito [#success-examples-Main_Basket-CompleteBasket-1_0_0-0]

```
HTTP/1.1 200 OK
{
  "ok": true,
}
```

- Error: El authToken no ha sido transmitido [#error-examples-Main_Basket-CompleteBasket-1_0_0-0]

```
HTTP/1.1 401 Unauthorized.
{
  "code": 401,
  "message": "No autorizado",
}
```

Main.Basket - Recibir los productos de la cesta

GET

/api/v1/orders/:id

Header

Campo	Tipo	Descripción
Authorization opcional	string	Encabezado de autorización en formato Bearer {authToken}.

- Ejemplo de autorización [#header-examples-Main_Basket-GetBasket-1_0_0-0]

```
{
  "Authorization": "Bearer jknnFApafP4awfAIFfafam2fma"
}
```

Parámetro

Campo	Tipo	Descripción
id	string	id de la cesta en la tabla order_model. Se pasa por el parámetro de URL

- Recibir la cesta [#parameter-examples-Main_Basket-GetBasket-1_0_0-0]

```
/api/v1/orders/6
```

- Respuesta: Los productos de la cesta recibidos con éxito [#success-examples-Main_Basket-GetBasket-1_0_0-0]

```
HTTP/1.1 200 OK
[
  {
    "id": 1,
    "name": "Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free",
    "price": 149,
    "weight": 473,
    "units": "ml",
    "quantity": 3
  },
  {
    "id": 4,
    "name": "Refresco Sprite",
    "price": 79,
    "weight": 900,
    "units": "ml",
    "quantity": 4
  }
]
```

- Error: El authToken no ha sido transmitido [#error-examples-Main_Basket-GetBasket-1_0_0-0]
- Error: Cesta no encontrada [#error-examples-Main_Basket-GetBasket-1_0_0-1]

```
HTTP/1.1 401 Unauthorized.
{
  "code": 401,
```



```
"message": "No autorizado",
}
```

HTTP/1.1 404 Not found.

```
{
  "code": 404,
  "message": "Not found"
}
```

Main.Orders

Main.Orders - Recepción de pedidos de los usuarios

GET

/api/v1/orders

Header

Campo	Tipo	Descripción
Authorization <small>opcional</small>	string	Encabezado de autorización en formato Bearer {authToken}.

- Recibir los pedidos de los/las usuarios/as [#header-examples-Main_Orders-GetOrders-1_0_0-0]

```
{
  "Authorization": "Bearer jknnFApafP4awfAIFfafam2fma"
}
```

- Respuesta: Todos los pedidos del/de la usuario/a recibidos con éxito [#success-examples-Main_Orders-GetOrders-1_0_0-0]

HTTP/1.1 200 OK

```
[
  {
    "id": 6,
    "userId": 1,
    "productsList": [
```

```

    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 5,
      "quantity": 2
    },
    {
      "id": 3,
      "quantity": 1
    }
  ],
  "deliveryPrice": 0,
  "deliveryPriceOur": 30,
  "deliveryTime": "25~30",
  "wareHouse": "Fresh food",
  "courierService": "Fast Delivery",
  "status": 1
}
]

```

- Error: El authToken no ha sido transmitido [#error-examples-Main_Orders-GetOrders-1_0_0-0]

HTTP/1.1 401 Unauthorized.

```

{
  "code": 401,
  "message": " ",
}

```

Utils

Utils - Logs de mensajeros/almacenes del servidor

Se trata de una herramienta para obtener las últimas líneas del log mixto de todos los servidores secundarios (mensajeros y almacenes). The count parameter means the number of lines to be read and returned in the response

GET

/api/logs/secondary

Parámetro

Campo	Tipo	Descripción
countopcional	number	Número de líneas desde el final Valor por defecto: 50

- Obtener los últimos 50 registros [#parameter-examples-Utills-GetSecondaryLogs-1_0_0-0]
- Obtener los últimos 100 registros [#parameter-examples-Utills-GetSecondaryLogs-1_0_0-1]

/api/logs/secondary

/api/logs/secondary?count=100

- Respuesta: Últimos registros del log [#success-examples-Utills-GetSecondaryLogs-1_0_0-0]

```
2020-08-26T16:07:33.017Z [INFO] [Warehouse][food-city]: Server is listening at port 4022
2020-08-26T16:08:01.324Z [DEBUG] [Warehouse][everything-you-need]: [Request] - ::ffff:127.0.0.1
2020-08-26T16:08:01.324Z [DEBUG] [Warehouse][fresh-food]: [Request] - ::ffff:127.0.0.1 PO
2020-08-26T16:08:01.324Z [DEBUG] [Warehouse][big-world]: [SOAP Client] received <?xml ver
2020-08-26T16:08:01.361Z [DEBUG] [Warehouse][food-city]: [Request] - ::ffff:127.0.0.1 POS
2020-08-26T16:39:17.168Z [DEBUG] [Warehouse][big-world]: [SOAP Client] received <?xml ver
2020-08-26T16:39:17.168Z [DEBUG] [Warehouse][fresh-food]: [Request] - ::ffff:127.0.0.1 PO
2020-08-26T16:39:17.170Z [DEBUG] [Warehouse][everything-you-need]: [Request] - ::ffff:127
2020-08-26T16:39:17.173Z [DEBUG] [Warehouse][food-city]: [Request] - ::ffff:127.0.0.1 POS
2020-08-26T16:39:21.398Z [DEBUG] [Courier][food-service]: [SOAP Client] received <?xml ve
2020-08-26T16:39:21.453Z [DEBUG] [Courier][fast-delivery]: [Request] - ::ffff:127.0.0.1 P
2020-08-26T16:39:21.481Z [DEBUG] [Courier][order-and-go]: [Request] - ::ffff:127.0.0.1 PO
2020-08-26T16:39:21.489Z [DEBUG] [Courier][speedy]: [Request] - ::ffff:127.0.0.1 POST:/sp
```

Utils - Logs del servidor principal

Esta es una herramienta para obtener las últimas líneas en el registro del servidor principal. El parámetro count indica el número de líneas que se leerán y se devolverán a Response

GET

/api/logs/main

Parámetro

Campo	Tipo	Descripción
countopcional	number	Número de líneas desde el final Valor por defecto: 50

- Obtener los últimos 50 registros [#parameter-examples-Utills-GetMainLogs-1_0_0-0]
- Obtener los últimos 100 registros [#parameter-examples-Utills-GetMainLogs-1_0_0-1]

/api/logs/main

/api/logs/main?count=100

- Respuesta: Últimos registros del log [#success-examples-Utills-GetMainLogs-1_0_0-0]

```
2020-08-26T16:38:31.489Z [INFO] [Main]: Server is listening at port 4000
2020-08-26T16:38:31.500Z [INFO] [Main]: [SOAP Client] Soap is listening at /api/wsd1
2020-08-26T16:38:31.532Z [DEBUG] [Main]: [SOAP Client] TRAIN client is initialized
2020-08-26T16:38:31.564Z [DEBUG] [Main]: [SOAP Client] WORLD client is initialized
2020-08-26T16:38:31.609Z [INFO] [Main]: [PostgreSQL] PostgreSQL is initialized
2020-08-26T16:38:35.057Z [DEBUG] [Main]: [Request] - ::1 GET:/api/logs/main - HTTP/1.1 -
2020-08-26T16:39:17.106Z [DEBUG] [Main]: [Request] - ::1 POST:/api/v1/warehouses/amount?d
2020-08-26T16:39:21.358Z [DEBUG] [Main]: [Request] - ::1 POST:/api/v1/couriers/check - HT
2020-08-26T16:39:36.767Z [DEBUG] [Main]: [Request] - ::1 GET:/api/v1/orders - HTTP/1.1 -
2020-08-26T16:39:47.908Z [DEBUG] [Main]: [Request] - ::1 GET:/api/logs/main - HTTP/1.1 -
2020-08-26T16:07:32.123Z [ERROR] [Main]: Unexpected error:
2020-08-26T16:07:32.127Z [ERROR] [Main]: {"message":"request to http://localhost:4022
2020-08-26T16:07:32.131Z [ERROR] [Main]: Unexpected error:
2020-08-26T16:07:32.162Z [ERROR] [Main]: {"stack":"RangeError [ERR_HTTP_INVALID_STATU
2020-08-26T16:08:01.311Z [DEBUG] [Main]: [Request] - ::1 POST:/api/v1/warehouses/amount -
```

Utills - Recuperar información de la tabla de base de datos

GET

```
/api/db/resources/{table_name}.csv
```

Parámetro

Campo	Tipo	Descripción
table_name	string	El nombre de la tabla en la base de datos. Se pasa en la URL.

- Recibir la tabla product_model [#parameter-examples-Utills-GetDBResource-1_0_0-0]
- Recibir la tabla kit_model [#parameter-examples-Utills-GetDBResource-1_0_0-1]

```
/api/db/resources/product_model.csv
```

```
/api/db/resources/kit_model.csv
```

- Respuesta: Archivo con el contenido de la tabla product_model [#success-examples-Utills-GetDBResource-1_0_0-0]

```
id,name,price,weight,units,categoryId,
1,Orange Juice - Cold-Pressed, No Added Sugar, Preservative Free,149,473,ml,1
2,Refresco Mountain Dew,89,1,1,1
3,Refresco Pepsi,109,1,1,1
4,Refresco Sprite,79,900,ml,1
5,Jugo Fruit Power: Lichi,349,900,ml,1
```