

AirMenu

API

API Version: 1.0.0

Document Version: 1.0.28

02-09-2022

1 - Requests

1.1 - Address

1.2 - Parameters

1.2.1 - ACTION

1.2.2 - VERSION

1.2.3 - DATA

1.2.4 - KEY

1.3 - Response

1.4 - Example

2 - Actions

2.1 - CreateUser

2.1.1 - JSON Parameters

2.1.2 - JSON Response

2.1.3 - Example 1

2.1.4 - Example 2

2.2 - Authenticate

2.2.1 - JSON Parameters

2.2.2 - JSON Response

2.2.3 - Example

2.2.a - GetEnterprisePosIds

2.2.1 - JSON Parameters

2.2.2 - JSON Response

2.2.3 - Example

2.3 - SaveEnterprise

2.3.1 - JSON Parameters

2.3.2 - JSON Response

2.3.3 - Example

2.3.b - GetEnterprise

2.3.b.1 - JSON Parameters

2.3.b.2 - JSON Response

2.4 - SaveMenu

2.4.1 - JSON Parameters

2.4.3 - JSON Schedule parameters

2.4.2 - JSON Response

2.4.3 - Example

2.5 - GetOrders

2.5.1 - JSON Parameters

- [2.5.2 - JSON Response](#)
- [2.5.3 - Example](#)
- [2.5.a - GetOrderIds](#)
 - [2.5.a.1 - JSON Parameters](#)
 - [2.5.a.2 - JSON response](#)
 - [2.5.a.3 - JSON response example](#)
- [2.6 - GetNotificationAccess](#)
 - [2.6.1 - JSON Parameters](#)
 - [2.6.2 - JSON Response](#)
 - [2.6.3 - Example](#)
- [2.7 - SendMessage](#)
 - [2.7.1 - JSON Parameters](#)
 - [2.7.2 - JSON Response](#)
 - [2.7.3 - Example](#)
- [2.8 - DeleteMenuNode](#)
 - [2.8.1 - JSON Parameters](#)
 - [2.8.2 - JSON Response](#)
 - [2.8.3 - Example](#)
- [2.9 - LockUser](#)
 - [2.9.1 - JSON Parameters](#)
 - [2.9.2 - JSON Response](#)
 - [2.9.3 - Example](#)
- [2.10 - UnlockUser](#)
 - [2.10.1 - JSON Parameters](#)
 - [2.10.2 - JSON Response](#)
 - [2.10.3 - Example](#)
- [2.11 - GetEnterpriseDivisionIds](#)
 - [2.11.1 - JSON Parameters](#)
 - [2.11.2 - JSON Response](#)
 - [2.11.3 - Example](#)
- [2.12 - GetMenu](#)
 - [2.12.1 - JSON Parameters](#)
 - [2.12.2 - JSON Response](#)
 - [2.12.3 - Example](#)
- [2.13 - CreatePromo](#)
 - [2.13.1 - JSON Parameters](#)
 - [2.13.2 - JSON Response](#)
 - [2.13.3 - Example](#)
- [2.14 - RedeemCode](#)

- [2.14.1 - JSON Parameters](#)
 - [2.14.2 - JSON Response](#)
 - [2.14.3 - Example](#)
- [2.15 - UpdateStock](#)
 - [2.15.1 - JSON Parameters](#)
 - [2.15.2 - JSON Response](#)
 - [2.15.3 - Example](#)
- [2.16 - UploadImage](#)
 - [2.16.1 - JSON Parameters](#)
 - [2.16.2 - JSON Response](#)
 - [2.16.3 - Example](#)
- [2.17 - GetLocationDivisionEnterprises](#)
 - [2.17.1 - JSON Parameters](#)
 - [2.17.2 - JSON Response](#)
 - [2.17.3 - Example](#)
- [2.18 - SendOrder](#)
 - [2.18.1 - JSON Parameters](#)
 - [2.18.2 - JSON Response](#)
 - [2.18.3 - Example](#)
- [2.19 - ConfigurePayment](#)
 - [2.19.1 - JSON Parameters](#)
 - [2.19.2 - JSON Response](#)
 - [2.19.3 - Example](#)
- [2.20 - SaveEnterpriseForm](#)
 - [2.20.1 - JSON Parameters](#)
 - [2.20.2 - JSON Response](#)
 - [2.20.3 - Example](#)
- [2.21 - ActivateFlag](#)
 - [2.21.1 - JSON Parameters](#)
 - [2.21.2 - JSON Response](#)
- [2.22 - GetActiveFlags](#)
 - [2.22.1 - JSON Parameters](#)
 - [2.22.2 - JSON Response](#)
- [2.23 - GetEnterprises](#)
 - [2.23.1 - JSON Parameters](#)
 - [2.23.2 - JSON Response](#)

[3 - Notification Service](#)

- [3.1 - Establishing connection](#)

[3.1.1 - Server and connection data](#)

[3.1.2 - Client association](#)

[3.2 - Notifications](#)

[3.2.1 ApiAction GetOrders](#)

[4 - Help with delivery integrations.](#)

[4.1 - Overview to create an enterprise using the API](#)

[4.1.1 - Create User.](#)

[4.1.2 - Authenticate User](#)

[4.1.3 - Create Enterprise](#)

[4.2 - Overview to receive orders](#)

[4.3 - Overview to update the menu](#)

[4.4 - Flags / Order Status](#)

[4.4.1 - Receive order status update](#)

[4.4.2 - Update order status](#)

[4.4.3 - Special Fields \(Extra Info\)](#)

[4.4.4 - Examples:](#)

1 - Requests

1.1 - Address

www.airmenu.com/AirMenuAPI

1.2 - Parameters

Parameters are defined in a ***x-www-form-urlencoded*** query string. API Requests could be sent as GET or POST requests.

1.2.1 - ACTION

Defines which action should be executed. ***DATA*** parameter must have the right contents for each action.

The value of this parameter must be one of the actions described in section 2.

1.2.2 - VERSION

Defines the API version to be used for this request. Current version is 1.0.0 and is used as default.

1.2.3 - DATA

Defines a JSON object with parameters and values to be consumed by the specified action. Value should be url encoded to avoid errors due to special characters reserved for ***x-www-form*** data.

1.2.4 - KEY

String with 64 chars which identifies the company using AirMenu API. To obtain an API Key email us at info@airmenu.com with your company name.

1.3 - Response

Response is a ***x-www-form-urlencoded*** with a **RESPONSE** parameter containing a JSON object.

Response JSON parameters may vary with the performed API action and always include the following parameters:

- action (string)
- version (string)
- status (string)
- statusCode (integer)

If the API action executes successfully, **statusCode** is returned as 1 (SUCCESS) otherwise returns -1 (ERROR).

When an error occurs, an error code, description, and message are included to help track the problem.

1.4 - Example

Normal:

[www.airmenu.com/AirMenuAPI?ACTION=Authenticate&VERSION=1.0.0&KEY=ab2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={\"username\":\"user@mail.com\",\"password\":\"w8D6sc5h\"}](http://www.airmenu.com/AirMenuAPI?ACTION=Authenticate&VERSION=1.0.0&KEY=ab2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={\)

Encoded:

<http://www.airmenu.com/AirMenuAPI?ACTION=Authenticate&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA=%7B%22username%22%3A%22user%40mail.com%22%2C+%22password%22%3A%22w8D6sc5h%22%7D>

2 - Actions

2.1 - CreateUser

Allows creating users.

2.1.1 - JSON Parameters

- username (string)
- password (string)
 - "", will send an email asking the user.
 - "!!NOPASSWORD!!", will use the user next password when login.
- addressForm (optional JSON)
 - address (String)
 - number (string)
 - complement (string)
 - city (string)
 - postalCode (string)
 - Phone (string)
 - countryCode (2 letters string)
- enterpriseld (string)
- firstName (optional string)
- lastName (optional string)

Username must be a valid email and shouldn't be in use.

2.1.2 - JSON Response

No data is returned.

2.1.3 - Example 1

```
ACTION=CreateUser&VERSION=1.0.0&KEY=abOEFKf6I5qJjrW3ArYZxtjfGx49
QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'username':'user@mail.c
om', 'password':'drowssap'}
RESULT={'action':'CreateUser','version':'1.0.0','status':'SUCCESS','statusCode':1
}
```

2.1.4 - Example 2

```
ACTION=CreateUser&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49
QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={"sessionId":"XxBoec8lbE
1yU-2woreLtQ","username":"nicolau124@gmail.com","password":"!!NOPASSWO
RD!!","firstName":"Nicolau","lastName":"Lucas","enterpriseId":"1538840998186","
addressForm":{"phone":"123456789","address":"Rua Primeiro de Dezembro
243","postalCode":"4450-227","city":"Matosinhos","countryCode":"PT"}}
```

```
RESULT={'action':'CreateUser','version':'1.0.0','status':'SUCCESS','statusCode':1
}
```

2.2 - Authenticate

Authenticates user and returns session id.

2.2.1 - JSON Parameters

- username (string)
- password (string)

All parameters are required. Username must be a valid email.
Session expires after 30 minutes.

2.2.2 - JSON Response

- sessionId (string)
- enterpriseIds (array[string])
- enterpriseNames (array[string])

2.2.3 - Example

```
ACTION=Authenticate&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49
QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'username':'user@mail.c
om','password':'drowssap'}
RESULT={'sessionId':'1vusjr9z0ev2o','enterpriseIds':['1260841159174'],'enterpris
eNames':['DemoEnterprise'],'action':'Authenticate','version':'1.0.0','status':'SUCCE
SS','statusCode':1}
```

2.2.a - GetEnterprisePosIds

Return POS ids for a specific enterprise.

2.2.1 - JSON Parameters

- enterpriseld (string)

All parameters are required.

2.2.2 - JSON Response

- posIds (Array [string])

2.2.3 - Example

```
https://www.airmenu.com/AirMenuAPI?ACTION=GetEnterprisePosIds&VERSION=1.0.0&KEY={{Key}}&DATA={"sessionId":  
"{{SessionID}}", "enterpriseld": "{{Enterpriseld}}"}}
```

```
RESULT={"posIds":["UPDATEFLAGS"], "action": "GetEnterprisePosIds", "version": "  
1.0.0", "status": "SUCCESS", "statusCode": 1}
```

2.3 - SaveEnterprise

Creates or saves enterprise and returns its id.

2.3.1 - JSON Parameters

- sessionId (string)
- enterpriseld (string)
- name (string)
- thumbnail (string)
- address (string)

- postalCode (string)
- city (string)
- email (string)
- phone (string)
- fiscalNumber (string)
- timeZone (string)
- orderPanelName (string)
- group (string)
- latitude (string)
- longitude (string)
- locale (string)
- basicMode (boolean)
- apiUser (boolean)
- parentDivisionName (string)
- divisionCount (int)
- enterpriseType (string)
- personalizedUrl (string)

All parameters are required except **group**, **orderPanelName**, **basicMode**, **parentDivisionName**, **divisionCount** and **enterpriseType**. When **basicMode** is included and set as true, all fields are redundant except **name**, **email**, **locale** and **timeZone**.

If **enterpriseld** is omitted, a new enterprise is created and its id is returned.

apiUser activates a function that allows the enterprise to export orders using the API function: **GetOrders**.

parentDivisionName and **divisionCount** are used to configure how automatic division creation works. If none of these parameters are defined, a parent division with name 'Room' is created with 24 child divisions.

personalizedUrl allows specifying a user-readable name to combine with the AirMenu URL to open the enterprise. Must be a unique alphanumeric word with a length between 3 and 20 chars. Example: <https://www.airmenu.com/anEnterprise>

Enterprise type is defined with **enterpriseType** and the supported types are:

- restaurant
- hotel
- delivery
- golf
- other

2.3.2 - JSON Response

- enterpriseld (string)

2.3.3 - Example

```
ACTION=SaveEnterprise&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx
49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'name':'Restaurant',
'address':'1st Street 22', 'postalCode':'22222', 'city':'Boston MA',
'latitude':'42.370468', 'longitude':'-71.033806', 'phone':'(617) 222-2222',
'locale':'en_US', 'timeZone':'America/Adak', 'basicMode':true,
'sessionId':'1vusjr9z0ev2o', 'email':'user@mail.com', 'fiscalNumber':'22222222',
'group':'default'}
```

```
RESULT={'enterpriseld':'1360841419361','action':'SaveEnterprise','version':'1.0.0'
,'status':'SUCCESS','statusCode':1}
```

2.3.b - GetEnterprise

Returns enterprise data

2.3.b.1 - JSON Parameters

- sessionId
- enterpriseld

2.3.b.2 - JSON Response

- sessionId (string)
- enterpriseld (string)
- name (string)
- thumbnail (string)
- address (string)
- postalCode (string)

- city (string)
- email (string)
- phone (string)
- fiscalNumber (string)
- timeZone (string)
- orderPanelName (string)
- group (string)
- latitude (string)
- longitude (string)
- locale (string)
- basicMode (boolean)
- apiUser (boolean)
- parentDivisionName (string)
- divisionCount (int)
- enterpriseType (string)
- personalizedUrl (string)

2.4 - **SaveMenu**

Saves a tree of products.

2.4.1 - JSON Parameters

- sessionId
- enterpriseld (string)
- menuInstance (object)
- roomName (string)(obsolete)
- deleteAllFirst (boolean)

All parameters are required except **roomName** and **deleteAllFirst**.

deleteAllFirst allows to delete current menu structure before saving menu.

menuInstance is a JSON object and defines a tree of products organized as family, item, modifier family and modifier. **menuInstance** has the following parameters:

- title (string)
- available (boolean)

- description (string)
- menuRelation (string)
- price (double)
- plu (string)
- childs (array[menuItem])
- min (integer)
- max (integer)
- priceCalculation (string)
- tax (double)
- avoidUpdate (boolean)
- automaticType (string)
- booking (object)

title and **menuRelation** are always required, **plu** is only required for items, modifiers and combos. Remaining parameters are redundant.

available is true by default and allows to disable a menu element.

menuRelation must take one of the following values: **family**, **item**, **complement**, **complementItem**, **enterprise** and **complexItem**; meaning respectively: family of products, product, modifier family, modifier, enterprise and combo product;

min and **max** are used in **complement** and **complexItem** instances.

priceCalculation is used to change how a price is calculated when traversing the tree of a combo product, being only used inside instances of **complexItem**. This parameter can have one of the following values:

- fixed - The element's own price
- sum - Sum of the price of its child elements
- fixedAndSum - The element's own price added to the sum of its childs
- max - Maximum price of its child elements
- min - Minimum price of its child elements
- fixedMulSum - Sum of the price of its child elements multiplied by its own price

Menu values that can be changed after insertion are: **title**, **price**, **min**, **max** and **priceCalculation**;

family titles must be changed only if they are identified with a **plu**, otherwise it may lead to unwanted results.

avoidUpdate allows to ignore updates to specific menu elements if they already exist. If the menu element does not exist or is not connected with its parent element, this parameter will be ignored and the element will be created and connected normally.

automaticType allows configure an item to be added to the order list

automatically. This parameter is not required and is only used in menu elements with relationType of item and complexItem. Possible values are:

- none - Disable automatic behaviour.
- group - Groups automatic items when navigating the menu and prompt them when finishing the order.
- immediate - Prompt automatic items when navigating the menu.

booking is an object containing the configuration to allow certain menu branches to ask a specific booking time. This parameter is only used in **menuInstances** with a **menuRelation** with value **family**. A **booking** object must define the following parameters:

- bookingType (string) - Mandatory with values of **disabled**, **stockOnly**, **separateOrder** and **replaceBooking**.
- onlyAllowBooking (boolean) - Allows to continue without booking or force the end user to pick one time slot.
- bookingMinimumTimeLimit (integer) - Minimum time margin from the instant of picking a time, in minutes, for the first time slot option.
- bookingMaximumTimeLimit (integer) - Maximum time margin from the instant of picking a time, in minutes, for the last time slot option.
- bookingMinuteInterval (integer) - Time slot size, in minutes.
- bookingIntervals (array[object]) - A list of objects defining time periods in a week to allow booking. This configuration is used in combination with **bookingMinimumTimeLimit** and **bookingMaximumTimeLimit**. If not defined, the whole week will be allowed. Each of this objects must contain the following parameters:
 - day (byte) - A bitwise parameter defining the day or group of days for this time period. Each bit represents a week day and could be combined to represent multiple days. In decimal: 0 - All week, 1 - Sunday, 2 - Monday, 4 - Tuesday, 8 - Wednesday, 16 - Thursday, 32 - Friday, 64 - Saturday.
 - start (integer) - Day time, in minutes, in which this time period starts, counting from 0 at 00h00 to 1439 at 23h59.
 - duration (integer) - The time span, in minutes, that this period lasts.

A **booking** configuration is not required to be sent every time the **menuInstance** is updated, being only necessary when disabling (setting **bookingType** to disabled) or when updating booking configuration.

roomName is obsolete, divisions are now created in **SaveEnterpriseAction** API action (see section 2.3).

In the case of Enterprise nodes, **menuRelation** should be **enterprise**, **referencedEnterpriseId** should have enterprise Id and **title** the enterprise name.

2.4.3 - JSON Schedule parameters

AirMenu allows to specify in which periods of time and for what division/time a given menu element is restricted. The object that defines these restrictions is called **schedule**. To configure schedules, two additional parameters can be used inside a **menuInstance** object:

- **scheduleList** (array[object])
- **scheduleNames** (array[string])
- **schedulePrices** (object)

scheduleList is only used in the top level object of **menuInstance**. This parameter is a list of objects containing the definition of the schedules to be created or updated and is not mandatory. So, it can be defined only when creating new schedules or when updating existing ones. This list only creates the schedules so they can be referenced by its title across the whole menu. A schedule object must contain the following properties:

- **title** (string) - Name of the schedule. This value is used as an id to reference the schedules that restrict a given menu element.
- **division** (string) - Id of the division that represents the location that restricts the schedule. Division ids can be found using the API call [2.11 - GetEnterpriseDivisionIds](#). Divisions from a direct parent enterprise in the menu hierarchy can also be used as restriction.
- **start** (integer) - Minute of the day this time starts. Example 60 will be 01:00
- **duration** (integer) - Number of minutes. Example, start:60, duration:30, time will be available between 01:00 and 10:30.
- **day** (integer) - You should add multiple values to get multiple days. Example 3 will be Sunday and Monday.
 - 1 - Sunday
 - 2 - Monday
 - 4 - Tuesday
 - 8 - Wednesday
 - 16 - Thursday
 - 32 - Friday
 - 64 - Saturday

scheduleNames is a list containing the title of the schedules that restrict a given

menu element. The title of the schedule must be equal to the title of the desired schedule configured in **scheduleList**. This parameter is not mandatory and can be used only when changing the schedules that restrict the menu element.

schedulePrices should be used to have different prices on a given menu element.

```
{
  "sessionId": "se_s7778999ffggggqytOlqP44887822",
  "enterpriseId": "11488927656836",
  "menuInstance": {
    "title": "Menu",
    "menuRelation": "family",
    "scheduleList": [
      {
        "title": "ScheduleA",
        "division": "1697616"
      },
      {
        "title": "ScheduleB",
        "division": "3888910"
      }
    ],
    "childs": [
      {
        "plu": 1,
        "title": "Pasta",
        "menuRelation": "family",
        "childs": [
          {
            "plu": 4,
            "title": "MEAT SAUCE",
            "description": "",
            "price": 8.95,
            "menuRelation": "item",
            "scheduleNames": [
              "ScheduleA",
              "ScheduleB"
            ]
          }
        ]
      }
    ]
  }
}
```

```

    ]
  }
]
}
}

```

2.4.2 - JSON Response

No data is returned.

2.4.3 - Example

```

ACTION=SaveMenu&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArYZxtjfGx49Q
sAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'1vusjr9z0ev2o',
'enterpriseld':'1360841419361', 'menuInstance':{'title':'Menu',
'menuRelation':'family', 'childs':{'plu':1, 'title':'Pasta', 'menuRelation':'family',
'childs':{'plu':'c1', 'title':'COMBO', 'description':'', 'min':1, 'max':2, 'price':0, 'menuRelati
on':'complexItem', 'priceCalculation':'max', 'childs':{'plu':'c2', 'title':'CHOOSE_A', 'des
cription':'', 'min':0, 'max':1, 'price':2, 'menuRelation':'complexItem', 'priceCalculation':'f
ixed', 'childs':{'plu':'c4', 'title':'OPTION1', 'description':'', 'price':4, 'menuRelation':'item'
}, {'plu':'c5', 'title':'OPTION2', 'description':'', 'price':3, 'menuRelation':'item'}}}, {'plu':'c3'
, 'title':'CHOOSE_B', 'description':'', 'min':0, 'max':1, 'price':0, 'menuRelation':'complex
Item', 'priceCalculation':'sum', 'childs':{'plu':'c6', 'title':'OPTION1', 'description':'', 'pric
e':1, 'menuRelation':'item'}, {'plu':'c7', 'title':'OPTION5', 'description':'', 'price':3, 'menu
Relation':'item'}}}}, {'plu':2, 'title':'ZITI, LINGUINI', 'description':'', 'price':7.95,
'menuRelation':'item'}, {'plu':3, 'title':'MARINARA', 'description':'', 'price':8.95,
'menuRelation':'item'}, {'plu':4, 'title':'MEAT SAUCE', 'description':'', 'price':8.95,
'menuRelation':'item'}, {'plu':5, 'title':'FISH N CHIPS', 'description':'', 'price':8.95,
'menuRelation':'item'}}}, {'plu':6, 'title':'Oysters & Clams', 'menuRelation':'family',
'childs':{'plu':7, 'title':'STEAMED OR RAW STEAMERS', 'description':'',
'price':19.95, 'menuRelation':'item'}, {'plu':8, 'title':'CHERRY STONE',
'description':'1/2 Dozen', 'price':11.95, 'menuRelation':'item'}, {'plu':9,
'title':'CHERRY STONE PLATTER (BAKERS DOZEN)', 'description':'',
'price':21.95, 'menuRelation':'item'}}}}

```

```

RESULT={'action':'SaveMenu', 'version':'1.0.0', 'status':'SUCCESS', 'statusCode':1}

```

2.5 - GetOrders

Get order items marked be exported.

2.5.1 - JSON Parameters

- sessionId (string)
- enterpriseld (string)
- exportMode (string)
- employeeExportMode (string)

All parameters are required except exportMode and employeeExportMode. exportMode and employeeExportMode should take the value of manual, automatic or disabled. When automatic is defined, all orders are automatically closed and exported. disabled disable any exportation when it is defined in exportMode or employeeExportMode.

Alternatively, getOrders can be used to get a single order:

GetOrders

```
{
  "sessionId": "6ZxPqkCgcecyB355W",
  "enterpriseld": "1538840998888"
  "orderId": "9940215"
}
```

2.5.2 - JSON Response

Returns a JSON map of arrays; orders. Each key of this map is represented by divisionName, and each value is a JSON array of objects containing the tree structure of an ordered item; orderItemInstances. Each tree object contains the following parameters:

- plu (string)
- title (string)
- count (integer)
- price (double)
- menuRelation (string)
- username (string)

- childs (array[orderItemInstance])
- orderId (string)
- orderCounter (integer)
- orderDate(integer)
- firstName(string)
- lastName(string)
- paymentMethod(String)
- employee (string)
- employeePwd (string)
- extraInfo (object)
- activeFlags(array)

Some **orderItemInstances** contained in a tree might not have **price**, **count** and **plu** if the value of this parameters does not exist or if its menuRelation doesn't justify it.

extraInfo, when available, contains extra information about the user and the order.

menuRelation defines the type of menu element in the tree structure. It could have one of the following values: **family**, **item**, **complement**, **complementItem** and **complexItem**; meaning respectively: family of products, product, modifier family, modifier and combined product.

If the order is a waiter call or bill request, the tree structure contains only a root element with **menuRelation** having one of the following values: **waiter** and **bill**; meaning respectively: call waiter and request bill.

activeFlags contains a list of all the active flags of the order. (operatorEmail, datetime, orderId, key and operator)

2.5.3 - Example

ACTION=GetOrders&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'enterprisId':'1260841159174','sessionId':'1vusjr9z0ev2o'}

RESULT={'action':'GetOrders','version':'1.0.0','status':'SUCCESS','statusCode':1,'orders':{'Table1':{'title':'Menu1','menuRelation':'family','childs':{'title':'Eggs','price':1.5,'count':1,'plu':'1','menuRelation':'item','childs':[]},{'title':'Soup','price':2,'count':2,'plu':'2','menuRelation':'item','childs':[]}}},{'Table2':{'title':'Menu2','menuRelation':'family','childs':{'title':'Potatoes','plu':'3','price':3.5,'count':3,'menuRelation':'item','childs':{'title':'Cut','menuRelation':'complement','childs':{'title':'French','plu':'4','menuR

elation':'complementItem','childs':[],{'title':'Chips','plu':'5','menuRelation':'complementItem','childs':[]}}}}}}}}}

2.5.a - GetOrderIds

Get all order Ids between 2 dates in milliseconds using Unix Time. You can later use these orderIds to get the orders using GetOrders API.

2.5.a.1 - JSON Parameters

- sessionId (string)
- enterpriseld (string)
- startDate (long)
- endDate (long)
- bookingStartDate (long)
- bookingEndDate (long)

Use (startDate AND endDate) OR (bookingStartdate AND bookingEndDate)

2.5.a.2 - JSON response

- orderLimitReached (boolean)
- orderIds (array)

2.5.a.3 - JSON response example

```
RESULT={"orderIds":["1260486","1260485","1259747"],"orderLimitReached":false,"action":"Get OrderIds","version":"1.0.0","status":"SUCCESS","statusCode":1}
```

2.6 - GetNotificationAccess

Requests Notification Service access. All returned parameters are required to setup a TCP Socket to receive notifications of server events. Notification Service is described in **section 3**.

2.6.1 - JSON Parameters

- sessionId (string)

- enterpriseld (string)
- POSId (String, Optional) (GetEnterprisePosIds)
- asExecutor (boolean, Optional)

First 2 parameters are required.

2.6.2 - JSON Response

- host (string)
- port (string)
- token (string)

token is used to associate the socket connection to the enterprise and have a validity from 2 to 4 minutes.

2.6.3 - Example

```
ACTION=GetNotificationAccess&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArY
ZxtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'ash
dub3445s28b98xXsF','enterpriseld':'123456789'}
```

```
RESULT={'host':'http://ec2-54-235-235-174.compute-1.amazonaws.com/','port':'8
001','token':'123456789#13634379146804953','action':'GetNotificationAccess','ve
rsion':'1.0.0','status':'SUCCESS','statusCode':1}
```

2.7 - SendMessage

Allows API users to send a message to a smartphone client.

2.7.1 - JSON Parameters

- sessionId (string)
- username (string)

- message (string)
- employee (string)

All parameters are required except **employee**. If **employee** is defined, a message will be sent to the smartphone used by the specified employee otherwise it will be sent to the user smartphone.

2.7.2 - JSON Response

No data is returned.

2.7.3 - Example

```
ACTION=SendMessage&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx4
9QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'ashdub3445
s28b98xXsF','username':'user@mail.com','message':'Hello World!'}
```

```
RESULT={'action':'SendMessage','version':'1.0.0','status':'SUCCESS','statusCode':1}
```

2.8 - **DeleteMenuNode**

Allows to delete a given menu node and its child elements.

2.8.1 - JSON Parameters

- sessionId (string)
- enterprisId (string)
- nodesToDelete (array[object])

All parameters are required. Parameter **nodesToDelete** must contain a list of objects containing **plu** to identify items and modifiers or **title** to identify families and modifier families without **plu**.

2.8.2 - JSON Response

No data is returned.

2.8.3 - Example

```
ACTION=DeleteMenuNode&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49QsAP9bksWj
4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'e45jshs3ksj','enterpriseId':'
123456789','nodesToDelete':[{'plu':1,'title':'Appetizers','menuRelation':'family'},{'pl
u':21,'title':'PORTUGUESE STYLE CLAM STEW','description':'pork belly, fried
potatoes','menuRelation':'item'}]}
```

```
RESULT={'action':'DeleteMenuNode','version':'1.0.0','status':'SUCCESS','statusC
ode':1}
```

2.9 - LockUser

Adds a user to the list of trusted users in a given enterprise (similar to checkin in an hotel). With locked users, an enterprise can restrict or allow access to some portions of the menu (restrictions can be created in the options dialog of family nodes in the menu editor on Backoffice and users are added to a special group called Clients).

2.9.1 - JSON Parameters

- sessionId (string)
- enterpriseId (string)
- userEmail (string)
- userForm (object)

All fields but **userForm** are required.

userForm is a key value map containing the values of the enterprise form fields (enterprise forms can be created in AirMenu Backoffice under Enterprise->Edit->Utilities->Forms editor) to be saved with the user. The key of the map should be the value of the key that defines a field. After locking a user, the user will appear on the User tab of the Frontoffice with a locker icon.

userEmail is the email of the user to be locked. If the user does not exist, an account will be created and the user will receive a confirmation email.

2.9.2 - JSON Response

No data is returned.

2.9.3 - Example

```
ACTION=LockUser&VERSION=1.0.0&DATA={'sessionId':'e45jshs3ksj','enterpriseId':'123456789','userEmail':'userEmail@mail.com','userForm':{'Name':'User Name','RoomNumber':'Room Number'}}
```

```
RESULT=RESULT={'action':'LockUser','version':'1.0.0','status':'SUCCESS','statusCode':1}
```

2.10 - UnlockUser

Removes a user from the list of trusted users in a given enterprise (similar to checkout in an hotel).

2.10.1 - JSON Parameters

- sessionId (string)
- enterpriseId (string)
- userEmail (string)

All fields are required. After unlocking a user, the user on the User tab of the Frontoffice will appear with an unlocked locker.

userEmail is the email of the user to be unlocked.

2.10.2 - JSON Response

No data is returned.

2.10.3 - Example

```
ACTION=UnlockUser&VERSION=1.0.0&DATA={'sessionId':'e45jshs3ksj','enterpriseId':'123456789','userEmail':'userEmail@mail.com'}
```

```
RESULT={'action':'UnlockUser','version':'1.0.0','status':'SUCCESS','statusCode':1
}
```

2.11 - GetEnterpriseDivisionIds

Returns division names and ids from a given enterprise. In AirMenu, divisions are a representation of locations from which orders can be placed. Division ids are used to get data from other API calls, such as [2.12 - GetMenu](#).

2.11.1 - JSON Parameters

- enterpriseld (string)

2.11.2 - JSON Response

- divisions (object)
- divisionTree (array[object])

divisions is a key value map where the key is the name of the division and the value is the id of the division. **divisionTree** contains a tree representation of all divisions. Each object inside **divisionTree** contains a parameter **name** (string), **id** (string) and **childDivisions** (array[object]). **childDivisions** has the same structure as **divisionTree**.

2.11.3 - Example

```
ACTION=GetEnterpriseDivisionIds&VERSION=1.0.0&DATA={'sessionId':'e45jshs
3ksj','enterpriseld':'123456789'}
```

```
RESULT={'action':'GetEnterpriseDivisionIds','version':'1.0.0','status':'SUCCESS',
'stateCode':1,'divisions':{'Map1':'1234','Map2':'4321','RoomA':'9876','RoomB':'678
9'},'divisionTree':[{'name':'Map1','childDivisions':[{'name':'3','id':'4236'}],
'id':'1234'},{'name':'Map2','childDivisions':[{'name':'2','id':'4237'}],
'id':'4321'},{'name':'RoomA','childDivisions':[{'name':'1','id':'4235'}],
'id':'9876'},{'name':'RoomB','childDivisions':[{'name':'4','id':'4268'}],
'id':'6789'}]}
```

2.12 - GetMenu

Returns a menu for a given division and time of the week. In AirMenu, menu organization and prices can vary depending on the time of the day, week day and division in which an order will be placed. For that reason, a division and time should be given so that a representation of a menu could be returned.

2.12.1 - JSON Parameters

- `enterpriseld` (string)
- `divisionId` (string)
- `weekTime` (long)

divisionId is the id of the division that represents a location. Available divisions can be returned using [2.11 - GetEnterpriseDivisionIds](#).

weekTime is the number of milliseconds since the beginning of a week. A week starts at 00:00:00 of Sunday and ends at 23:59:59 of Saturday being the week times 0ms and 604800000ms respectively.

2.12.2 - JSON Response

- menu (array[object])
- menuStartTime(long)
- menuEndTime(long)
- divisionId (string)

menu is a list of objects that represents a menu tree. This objects can be fetched recursively and their structure are very similar to the structure of an order returned in [2.5 - GetOrders](#) or a menu submitted using [2.4 - SaveMenu](#). Each element of this structure includes the property **id** representing a node id. The same itemId can exist for different node ids.

menuStartTime and **menuEndTime** are times since the beginning of the week in milliseconds indicating the beginning and the end of the returned menu availability for a division **divisionId**.

2.12.3 - Example

ACTION=GetMenu&VERSION=1.0.0&DATA={'sessionId':'e45jshs3ksj','enterpriseId':'123456789','divisionId':'1234','weekTime':800000}

```
RESULT={'menu':{'childs':{'childs':[],'menuRelation':'enterprise','title':'EnterpriseA','description':'','thumbnail':'','plu':'','referencedEnterpriseId':'2367124366542','itemId':'1234'},{'childs':[],'menuRelation':'item','title':'ItemA','description':'','price':2.0,'thumbnail':'','plu':'','itemId':'3245','tax':0.0},{'childs':{'childs':{'childs':{'childs':[],'menuRelation':'item','title':'ItemC','description':'','price':3.2,'thumbnail':'','plu':'','itemId':'7621','tax':23.0},{'childs':[],'menuRelation':'item','title':'ItemD','description':'','price':4.6,'thumbnail':'','plu':'','itemId':'2816','tax':21.0}],'menuRelation':'complexItem','title':'ComplexB','description':'','price':0.0,'thumbnail':'','min':1,'max':1,'priceCalculation':'sum','plu':'','itemId':'1626','tax':0.0},{'childs':{'childs':[],'menuRelation':'item','title':'ItemE','description':'','price':5.0,'thumbnail':'','plu':'','itemId':'9876','tax':12.0}}]}
```

```

mF','description':'','price':5.2,'thumbnail':'','plu':'','itemId':'6925','tax':21.0},{'childs':[]
,'menuRelation':'item','title':'ItemE','description':'','price':5.0,'thumbnail':'','plu':'','ite
mId':'6318','tax':23.0}], 'menuRelation':'complexItem','title':'ComplexC','description'
:','price':0.0,'thumbnail':'','min':1,'max':1,'priceCalculation':'sum','plu':'','itemId':'969
7','tax':0.0}], 'menuRelation':'complexItem','title':'ComplexA','description':'','price':2.
0,'thumbnail':'','min':2,'max':2,'priceCalculation':'fixedAndSum','plu':'','itemId':'3965'
,'tax':0.0},{ 'childs':[{'childs':[{'childs':[], 'menuRelation':'complementItem','title':'Coml
temA','price':0.3,'plu':'','itemId':'4812','tax':0.0},{ 'childs':[], 'menuRelation':'complem
entItem','title':'ComItemB','price':0.2,'plu':'','itemId':'1129','tax':0.0}], 'menuRelation':
'complement','title':'CompA','min':1,'max':1,'plu':'','itemId':'9783'}]], 'menuRelation':'it
em','title':'ItemB','description':'','price':1.0,'thumbnail':'','plu':'','itemId':'1566','tax':23.
0}], 'menuRelation':'family','title':'FamilyB','description':'','thumbnail':'','plu':'','ite
mId':'2332'}]], 'menuRelation':'family','title':'FamilyA','description':'','thumbnail':'','plu':'','ite
mId':'2332'}]], 'divisionId':'4321','menuStartTime':518400000,'menuEndTime':60480
0000,'action':'GetMenu','version':'1.0.0','status':'SUCCESS','statusCode':1}

```

2.13 - CreatePromo

Creates a promotion that can be triggered with a given code. When triggered, a voucher offering an item is created and associated to an user. The api call RedeemCode ([2.14 - RedeemCode](#)) can be used to trigger a promotion.

2.13.1 - JSON Parameters

- enterpriseld (string)
- title (object)
- description (object)
- validityStart (long)
- validityEnd (long)
- key (string)
- code (string)
- limit (integer)
- userLimit (integer)
- offeredItemCount (integer)
- offerStart (long)
- offerEnd (long)
- offeredItemBranch (array[string])
- deleteExistingPromos (boolean)
- replaceExistingPromo (boolean)

Required parameters are enterpriseld, title, description, code, offeredItemBranch,

validityStart, **validityEnd**, **offerStart** and **offerEnd**. **title** and **description** are key value maps in which the key is the locale code and value is the text. Promo and offer validity should have valid unix timestamps in milliseconds and are defined with **validityStart**, **validityEnd**, **offerStart** and **offerEnd**. **offeredItemBranch** is a list of node ids describing a path to an item. Node ids can be obtained with the api [2.12 - GetMenu](#). **deleteExistingPromos** removes all previously saved promos. **replaceExistingPromo** removes a previously existing promo with the same key.

2.13.2 - JSON Response

No data is returned.

2.13.3 - Example

```
ACTION=CreatePromo&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArYZxtjfGx4
9QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'1917lh4p0e
wv6','enterpriseld':'1360841419361','title':{'en':'From API Promo','pt':'Promo da
API'},'description':{'en':'Created inside the
API'},'validityStart':0,'validityEnd':1515201334188,'key':'fromApi','code':'fromAPI',
'offeredItemCount':1,'offerStart':0,'offerEnd':1515201334188,'offeredItemBranch':['
22507','22545'],'deleteExistingPromos':false, 'replaceExistingPromo':true}
```

2.14 - RedeemCode

Triggers a promotion, unlocking a voucher with an offered item. The voucher is associated to a given user.

2.14.1 - JSON Parameters

- enterpriseld (string)
- code (string)
- userId (string)
- cultureCode (string)

enterpriseld, **code** and **userId** are required parameters. **cultureCode** is a locale code to change the language in which notification emails are generated.

2.14.2 - JSON Response

- createdVouchers (array[string])

An array of unlocked vouchers is returned.

2.14.3 - Example

```
ACTION=RedeemCode&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx4
9QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'1917lh4p0e
wv6','enterpriseld':'1360841419361','code':'fromAPI','userId':'rui@airmenu.com'}
```

```
RESULT={'createdVouchers':['From API
Promo'],'action':'RedeemCode','version':'1.0.0','status':'SUCCESS','statusCode':1
}
```

2.15 - UpdateStock

Allows to create and update the stock of products/services for a given enterprise. The stock is defined in lots with the information of units, available times and product code.

2.15.1 - JSON Parameters

- enterpriseld (string)
- lots (array[object])

Both **enterpriseld** and **lots** are required parameters. **lots** is an array of objects describing a lot who represents a batch of units for a given product or service.

A lot object contains the following properties: **plu**, **type**, **name**, **start**, **expiration** and **qty**. **plu** is the product/service code for this lot and is always required. **type** should take the value **consumable** or **reusable** and is required when the lot is being created. **name** is a display name or code to distinguish the lot when listing. **start** and **expiration** are unix timestamps (long in ms) that indicate in which time period this lot is available and are required when creating a new lot. **qty** defines the number of product/service units in this lot.

When a lot already exists, only **plu** along with the properties to update are required. When a given lot is updated, previous consumed or reserved units are reset.

2.15.2 - JSON Response

No data is returned.

2.15.3 - Example

```
ACTION=UpdateStock&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49
QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'fffuufjdeennn',
'enterpriseld':'1360841419361','lots':[{'type':'consumable','name':'autom','start':15
54022419000,'expiration':1554022519000,'plu':'AM115','qty':15}]}
```


2.16 - UploadImage

Specifies and assigns an image to a menu item or enterprise. The image is submitted as base 64 encoded string of a png or jpg. This is an asynchronous action, the upload will be performed some time after the response has been returned.

2.16.1 - JSON Parameters

- **enterpriseld** (string)
- **plu** (string)
- **imageData** (string)
- **form** (object)
- **divisionId** (string)

enterpriseld and **imageData** are required. To assign an image to a menu item, a **plu** of that item must be provided. If no **plu** is specified, the image is assigned as a thumbnail of the enterprise.

imageData is a base 64 string of a png or jpg image. **imageData** is only allowed to start with 'data:image/png;base64,' or 'data:image/jpg;base64,' prefixes and should not exceed 500000 chars in length.

form is an object containing the necessary description of a form to identify it and apply the image. If this parameter is specified, the **plu**, if present, will be ignored. The necessary parameters to identify a form are: **location** and **key**. If the form where the image is being applied is for a specific location, a **divisionId** must also be provided. More information about the form object can be found in the section [2.20 - SaveEnterpriseForm](#).

divisionId is optional and is used to identify the form's location where the image is being applied. **divisionIds** can be found using the API call [2.11 - GetEnterpriseDivisionIds](#).

2.16.2 - JSON Response

No data is returned

2.16.3 - Example

```
ACTION=UploadImage&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArYZxtjfGx4
9QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'ffuufjdeenn
n','enterpriseld':'1360841419361','plu':'fruit','imageData':'
9j/4AAQSkZJRgABAQAAQAB (...) RRRWUld1N6H/2Q==}'
```

2.17 - GetLocationDivisionEnterprises

Finds all map divisions and enterprises for a given location or address.

2.17.1 - JSON Parameters

- **enterpriseld** (string)
- **address** (string)
- **postalCode** (string)
- **country** (string)
- **latitude** (double)
- **longitude** (double)

enterpriseld is always required. If **latitude** and **longitude** are defined, the remaining parameters are ignored, otherwise, **address**, **postalCode** and **country** will be used to find a location.

An **address** is the complete address of a location (street name, city, state and must include itself the postal code and country). Parameters **postalCode** and **country** should always be defined together. An **address**, if defined, has priority over **postalCode** and **country**, if AirMenu fails to find a valid location from a complete address, **postalCode** and **country** will be used.

2.17.2 - JSON Response

- **foundDivisions** (array[string])
- **foundEnterprises** (array[object])

foundDivisions lists the name of the divisions that contains the location.

foundEnterprises is an array of objects containing the **id** and **name** of the enterprises available for **foundDivisions**.

2.17.3 - Example

```
ACTION=GetLocationDivisionEnterprises&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'ffuufjdeennn','enterpriseld':'1360841419361','address':'Avenida da República, 40, 4440-450, Matosinhos, Portugal','postalCode':'4440-445','country':'Portugal'}
```

```
RESULT={'foundDivisions':['Porto','Matosinhos'],'foundEnterprises':[{'id':'3464675755529','name':'Churrasco1'},{'id':'14249976444','name':'Pizzaria2'}],'version':'1.0.
```

```
0,'status':'SUCCESS','statusCode':1}
```

2.18 - SendOrder

This action sends an order to a given enterprise specifying a list of plus of the items to order. Only simple items without modifiers and single enterprise orders are allowed.

2.18.1 - JSON Parameters

- **enterpriseId** (string)
- **divisionId** (string)
- **pluList** (array[object])
- **bookingTime** (long)
- **username** (string) Optional. Only possible if the **sessionId** is the **sessionId** of an operator. (To be an operator in the AirMenu configurator, this user must be in the Operator's group)

enterpriseId is the id of the enterprise that will receive an order. **divisionId** is the id of the location where the order is being made and can be found using the API call [2.11 - GetEnterpriseDivisionIds](#). **pluList** is an array of objects, each one representing an item of the order. To book an order, the parameter **bookingTime** should be defined with an UNIX timestamp (UTC) representing the date and time of the order.

Each object inside **pluList** must contain a parameter **plu** (string), indicating the item to order, **count** (integer), indicating the quantity to order, and **forms** (object), an optional object with key-value pairs of values to fill form fields associated with the item.

Only plus of simple items or complex items with a price calculation of "Sum + price form field" are allowed. If a "Sum + price form field" complex item is being ordered, use the **forms** parameter to fill the corresponding price form field associated with the item with the desired price. **Username** is the name of the person who will be placing the order. If this parameter is not present the username of the session will be used.

2.18.2 - JSON Response

- **orderNumber** (string)
- **trackingUrl** (string)
- **orderId** (string)

If the order is placed successfully, an order number is returned (**orderNumber**). If the enterprise allows order location tracking (has an executor POS/Keyboard), an URL (**trackingUrl**) is returned to show the location tracking for that order.

2.18.3 - Example

```
ACTION=SendOrder&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArYZxtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'12vyw190p7z72','enterpriseId':'1415201384168','divisionId':'627889','pluList':[{'plu':'coffee','count':2},{ 'plu':'lemonJuice','count':3,'forms':{'desc':'An order description','address','1st Street, The Shire'}}], "bookingTime":1615526325000}
```

```
RESULT={'orderNumber':'133','trackingUrl':'https://www.airmenu.com/?tracking=x4uz1NBC7ft2vPsihefw','action':'SendOrder','version':'1.0.0','status':'SUCCESS','statusCode':1}
```

2.19 - ConfigurePayment

Set up a payment method for a given enterprise. A payment method must be configured with its keys and settings that can be obtained from the payment method's development dashboard or shared by an integration agent.

2.19.1 - JSON Parameters

- **enterpriseId** (string)
- **divisionId** (string)
- **paymentMethod** (string)
- **action** (string)
- **minPrice** (float)
- **maxPrice** (float)
- **allowChildPaymentOverride** (boolean)
- **overrideParentPayment** (boolean)
- **parameters** (object)

enterpriseId is the id of the enterprise where the payment will be configured.

divisionId is the id of the location where the payment will be used and is not required. If not defined, the payment is configured for all locations. **divisionIds** can be found using the API call [2.11 - GetEnterpriseDivisionIds](#).

paymentMethod is the type of payment to configure. Available payment methods are: [EasyPayMB](#), [MBWay](#), [PayPal](#), [PayPalCreditCard](#), [Stripe](#) and [StripeSources](#).

action specifies if the payment is being configured or disabled. Available options are [configure](#) and [disable](#).

minPrice and **maxPrice** sets the minimum and maximum price allowed for the payment method being configured.

allowChildPaymentOverride can be specified even if no payment method is being configured. This parameter allows to indicate that this enterprise's payment methods should override if allowed, the payment methods from the parent enterprise in the menu's hierarchy.

allowChildPaymentOverride can be specified even if no payment method is being configured. This parameter gives permission for any child enterprise in the menu's hierarchy to override payment methods.

parameters is an object containing the necessary keys and options for the payment method being configured. For each of the allowed payment methods, the required parameters are as following:

- EasyPayMB
 - accountId (string)
 - apiKey (string)
 - maxPayTime (integer) - the maximum amount of time, in minutes, to wait for a payment before cancelling the order.
 - shouldSendEmail (boolean) - to indicate that an email should be sent with instructions to complete the payment.
 - sandboxEnvironment (boolean) - must be true for test account ids and api keys.
- MBWay
 - merchantPosId (string)
- PayPal
 - username (string)
 - password (string)
 - signature (string)
 - sandbox (boolean) - must be true for test api keys.
- PayPalCreditCard
 - username (string)
 - password (string)
 - signature (string)
 - sandbox (boolean) - must be true for test api keys.
- Stripe
 - publishableKey (string)
 - secretKey (string)
- StripeSources
 - publishableKey (string)
 - secretKey (string)
 - use3dSecure (string) - allowed values: no, ifPossible and always

2.19.2 - JSON Response

No data is returned

2.19.3 - Example

```
ACTION=ConfigurePayment&VERSION=1.0.0&KEY=abOEFKf6l5qJjrW3ArYZxtj
fGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'eqQp3k
nx67poow','enterpriseId':'1415201384168','paymentMethod':'PayPal','parameter
s':{'username':'business_api1.thestore.com','password':'15535678722','signature
':'WFGddR5GHH88s9WWt.VBhWW2Po99SSddjUUioK','sandbox':'true'}}
```

2.20 - SaveEnterpriseForm

Create, modify or delete forms to be prompted to the end user.

2.20.1 - JSON Parameters

- enterpriseId (string)
- divisionId (string)
- action (string)
- form (object)

enterpriseId is the id of the enterprise where the form will be configured.

divisionId is the id of the location where the form will be used and is not required. If not defined, the form is configured for all locations. **divisionIds** can be found using the API call [2.11 - GetEnterpriseDivisionIds](#).

action specifies if the form is being configured or deleted. Available options are configure and delete. If not specified, the form will be configured.

form is an object containing metadata that specify the structure and behaviour of the form being configured and its fields. A form object has the following parameters:

- cultureTitles (object)
- location (string)
- key (string)
- formFields (array[object])

cultureTitles is a key-value map containing the internationalized title where the key is a ISO 639-1 language code and the value is a title. Example: {"en":"Title", "pt":"Título", "":"Default title"}. A default or single title can be specified by setting an empty language code.

location is the place inside AirMenu where the form will be prompted to the end user and also specifies the purpose of the form. Available values are: PlaceOrder, CancelOrder, Payment and OpenMenu.

key is a unique identifier of the form and is only required for forms with location Payment.

formFields is an array of objects containing the metadata that specifies the title, type of data, validation and default values of the form fields. Each object represents a single form field and has the following parameters:

- cultureTitles (object)
- fieldType (string)
- key (string)
- defaultValue (string)
- required (boolean)
- minLength (int)
- maxLength (int)
- minValue (double)
- maxValue (double)
- availableValues (array[object])

cultureTitles is a key-value map containing the internationalized title of the field. This parameter must be configured in a similar to the **cultureTitles** parameter of the form object.

fieldType specifies the type of data this form field handles and influences its validation. Possible values are: text, textAddress, integer, decimal, decimalPayment, label, orderTotalValueLabel, orderExchangeValueLabel, password, ddMMYY, mmYY, mmYYYY, YYYYmm, ddMMYYYY, YYYYMMdd, MMddYYYY, MMddYY, HHmm, check, select, image, addressForm, priceField and html.

key is a required unique value that identifies the field.

defaultValue is an optional value used to prefill the field.

required indicates if the field is mandatory and must be always filled. The default value is true.

minLength is optional and specifies the minimum length of the field's value. The default value is zero.

maxLength is optional and specifies the maximum length of the field's value. The default value is 100.

minValue is optional and specifies the minimum value for numeric field types. Not defined by default.

maxValue is optional and specifies the maximum value for numeric field types. Not defined by default.

availableValues defines the available options for fields of type 'select'. The parameter is defined as an array of objects where each object represents an option. Each of these objects must contain two parameters: **cultureTitles** and **defaultValue**. **cultureTitles** are similar to the **cultureTitles** parameter of the form and form field and specifies the display value of the option. **defaultValue** specifies the value of the option.

2.20.2 - JSON Response

No data is returned

2.20.3 - Example

```
ACTION=SaveEnterpriseForm&VERSION=1.0.0&KEY=abOEFKf6l5qJrW3ArYZ
xtjfGx49QsAP9bksWj4cX1zQ5rwQSlr4oVhXB8WnUv2&DATA={'sessionId':'eqQp
3knx67poow','enterpriseId':'1415112384168','action':'configure','form':{'cultureTitle
s':{'pt':'Pagamento','en':'Payment'},'location':'Payment','key':'paymentA','formField
s':[{'cultureTitles':{'en':'Description'},'key':'desc','defaultValue':'A
description','fieldType':'text'},{'cultureTitles':{'en':'Qty'},'key':'qty','defaultValue':'1','fi
eldType':'integer','minLength':3,'maxLength':56,'minValue':-10,'maxValue':55,'requ
ired':false},{'cultureTitles':{'en':'Total'},'key':'total','defaultValue':'','fieldType':'orderT
otalValueLabel','minLength':3,'maxLength':56,'minValue':-10,'maxValue':55,'requir
ed':false}]}}
```

2.21 - ActivateFlag

Activate order Flag

2.21.1 - JSON Parameters

- sessionId
- enterpriseId
- orderId
- operatorEmail
- executorEmail
- flagKey

2.21.2 - JSON Response

No data is returned

2.22 - GetActiveFlags

Get all the order Flags

2.22.1 - JSON Parameters

- sessionId
- enterpriseld
- orderId

2.22.2 - JSON Response

Active Flags

2.23 - GetEnterprises

Get user Enterprises

2.23.1 - JSON Parameters

- sessionId

2.23.2 - JSON Response

Return Enterprise list

3 - Notification Service

The Notification service allows API users receive notifications of a given server event. API users must connect to the server through a TCP socket and wait for messages to be pushed from the server.

Notification messages may vary and they notify API users the need to call a specific API function.

3.1 - Establishing connection

3.1.1 - Server and connection data

To establish a connection to the service, API users must call API function **GetNotificationAccess** as described in **section 2.6** to get server details and an association **token**. A TCP socket must be created using the given **host** and **port**.

3.1.2 - Client association

As soon as a socket connection is established, the socket must be associated with an enterprise to start receiving notifications. To associate a socket with an enterprise, the association **token** retrieved from **GetNotificationAccess** must be sent. If the association succeeds, server will push **SUCCESS** and the socket will start to receive notifications for an enterprise, otherwise, the socket will be closed (association will only fail if a wrong association token is submitted or if the token has expired).

Association **token** has a validity of about 2 to 4 minutes after being returned from the API function and, after creation, a socket connection has 30 seconds of time out to be associated with an enterprise.

3.2 - Notifications

Notifications are text messages ending with **NUL** char (\0).
Next are described all possible notification messages:

3.2.1 ApiAction_GetOrders

Alerts API users that new orders are waiting to be exported. To get orders marked to export, the API function **GetOrders** must be called (**section 2.5**).

4 - Help with delivery integrations.

This section tries to explain the singularities of integration thinking about receiving orders from UberEats, BoltFood, and Glovo and updating their menus.

Every API call is an HTTP GET or POST request to <https://www.airmenu.com/AirMenuAPI>

A request must always include the following four parameters:

- ACTION - The action that the API call must perform. Example: Authenticate, GetNotificationAccess, GetOrders
- VERSION - Version of the API. The current and only version is 1.0.0.
- KEY - A key generated and provided by us for BigTrak. Used to identify which Software integration is using our API for support and usage control.
- DATA - A JSON with all the data required for an API action to be executed.

4.1 - Overview to create an enterprise using the API

The AirMenu API can be used to directly create users and enterprises.

4.1.1 - Create User.

Check the following [link](#) to create a user. You will only need to create a user that will be used forever. This user will be the owner of the enterprise.

4.1.2 - Authenticate User

You will need to do authentication to get the sessionId that will be used in all future calls. Use the following [link](#) to do the authentication.

4.1.3 - Create Enterprise

Check the following [link](#) to create an enterprise.

4.2 - Overview to receive orders

Here is a step-by-step guide to the necessary API calls to receive orders:

To simplify and for test purposes, we will assume that a user and a business were already created and configured.

1. Authenticate the owner of the AirMenu business.
 - Call the API with ACTION=[Authenticate](#)
 - Provide a DATA JSON parameter containing a username and password.
 - Example: DATA={'username':'[user@mail.com](#)','password':'w8D6sc5h'}
 - If successful, the response contains a parameter 'sessionId' that must be provided for every other API call to identify the user.
 - The response also includes 'enterpriseNames' and 'enterpriselds' containing respectively the name of the businesses the user created and their corresponding ids. Those lists can be used to obtain the id of the business to refer to in the following API actions.
2. Get access to the notification service
 - Call the API with ACTION=GetNotificationAccess
 - Provide a DATA JSON parameter with a session id and a company id.
 - Example:
{'sessionId':'ashdub3445s28b98xXsF','enterpriseld':'123456789'}
 - The response contains the necessary details to create a TCP socket for our notification service.
3. Do a TCP socket connection to our notification service
 - Connect to the host (without http://) and port provided by the previous API call.
 - As soon as the connection is established, send the token, also provided by the previous API call, through the socket to vinctuate the connection.
 - The server should respond with SUCCESS.
 - This socket should be kept alive waiting for messages "GetOrders" notifying when orders are waiting to be exported by the API.
 - A 'ping' message can be sent from time to time to check the status of the socket.
4. Get the orders waiting to be exported
 - When a notification "GetOrders" is received, the API ACTION=[GetOrders](#) should be called.
 - Provide a DATA JSON parameter with the session id and company id.
 - As a response, a structure containing the ordered items and its menu branches is returned.

4.3 - Overview to update the menu

Here is a step-by-step guide to the necessary API calls to update the menu.

To simplify and for test purposes, we will assume that a user and a business were already created and configured.

1. Authenticate the owner of the AirMenu business.
 - Call the API with ACTION=Authenticate
 - Provide a DATA JSON parameter containing a username and password.
 - Example:
DATA={'username':'[user@mail.com](#)','password':'w8D6sc5h'}
 - If successful, the response contains a parameter 'sessionId' that must be provided for every other API call to identify the user.
 - The response also includes 'enterpriseNames' and 'enterpriselds' containing respectively the name of the businesses the user created and their corresponding ids. Those lists can be used to obtain the id of the business to refer to in the following API actions.
2. Save the menu using the SaveMenu API call.
 - Call the API with ACTION=SaveMenu
 - Provide a DATA JSON parameter containing an enterpriseld(String), menuInstance(JSON),
 - Check the [SaveMenu](#) function to get more info.
3. The AirMenu menu format is very generic and you will be able to create all levels of families within families. To improve compatibility with all delivery platforms, we recommend that you only use one level of families.
4. After saving the menu AirMenu will automatically insert in the menu some special items, that will be used to insert informations like: Order Number, Delivery or Pickup, Observations, Tax Number.

4.4 - Flags / Order Status

You should use 4 flags in your delivery connection.

ACCEPT

The order has been accepted by the store.

DENY

The order has been rejected by the store/courier.

READY

The order is ready to be picked up by a courier or the customer.

PICKED

The order was picked up by a courier or the customer.

4.4.1 - Receive order status update

- Although these flags can be updated by the API, they can also be updated by the AirMenu, so the software must be prepared to receive changes in the state of the flags.
- AirMenu will send a notification with "GetOrders", always that a flag is changed / order is changed.
- After that, you should call the API ACTION=[GetOrders](#), and look at the **activeFlags**(array) inside the order to get the flag status.
- **activeFlags** contains a list of all the active flags of the order. (operatorEmail, datetime, orderId, key and operator)

Example:

```
"activeFlags":[{"operatorEmail":"user1@gmail.com","datetime":1662463898286,"orderId":"192229925","key":"ACCEPT","operator":"User One"},{"operatorEmail":"user2@gmail.com","datetime":1662463903296,"orderId":"192993325","executor":"User two","executorEmail":"executor@gmail.com","key":"PICKED","operator":"Courier"}]
```

4.4.2 - Update order status

You should update the following flags :

- ACCEPT - After the customer accept the order.
 - DENY - If the customer reject the order
 - READY - When the order is ready to be picked.
 - PICKED - After the customer picks up the order.
-
- Call the API ACTION = "ActivateFlag" to activate a flag.
 - Parameters are:
 - sessionId
 - enterpriseld
 - orderId
 - operatorEmail
 - executorEmail
 - flagKey

4.4.3 - Special Fields (Extra Info)

- AM_TAX_PAYER_NUMBER
- AM_PROVIDER_ORDER_ID
- AM_ORDER_TYPE
- AM_PROVIDER_NAME
- AM_ORDER_TYPE
- AM_NOTE (It is possible to have order and item notes)
- AM_PROMO_VALUE
- AM_PROMO_DESC

4.4.4 - Examples:

Example 1:

```
{
  "Matosinhos": {
    "childs": [
      {
        "childs": [
          {
            "childs": [
              {
                "childs": [
                  {
                    "available": true,
                    "menuRelation": "complementItem",
                    "title": "Picante",
                    "price": 0,
                    "auxPrice": 0,
                    "id": "4699678",
                    "plu": "PLUA103"
                  },
                  {
                    "available": true,
                    "menuRelation": "complement",
                    "title": "Picante/Sem Picante",
                    "plu": ""
                  },
                  {
                    "count": 1,
                    "extraInfo": {
                      "AM_NOTE": "SEM OVO, por favor."
                    },
                    "available": true,
                    "menuRelation": "item",
                    "title": "Francesinha à Ship",
                    "price": 12,
                    "auxPrice": 0,
                    "id": "4699640",
                    "plu": "PLU1010",
                    "tax": 13
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Francesinhas Like a Ship",
                    "plu": ""
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Like a Ship Francesinhas",
                    "plu": "Francesinhas Like a Ship"
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Root",
                    "plu": ""
                  }
                ],
                "referencedEnterpriseName": "Like a Ship (Menu Geral Francesinhas)",
                "referencedEnterprisePhone": "934894045",
                "referencedEnterpriseAddress": "Rua Nova Estádio Nº306",
                "referencedEnterpriseCity": "Matosinhos",
                "referencedEnterpriseCountry": "Portugal",
                "referencedEnterpriseLatitude": 38.7027632099636,
                "referencedEnterpriseLongitude": -9.398674025231925,
                "available": true,
                "menuRelation": "enterprise",
                "title": "Like a Ship (Menu Geral Francesinhas)",
                "plu": "",
                "referencedEnterpriseId": "1656930854748",
                "username": "bff285e33e-c8cb-4680-8038-8c345333ce0f@anonymous.com",
                "orderCounter": 200,
                "orderId": 1962675,
                "extraInfo": {
                  "AM_TAX_PAYER_NUMBER": "",
                  "AM_PROVIDER_ORDER_ID": "B30L8",
                  "AM_PROVIDER_NAME": "BoltFood",
                  "AM_ORDER_TYPE": "delivery",
                  "AM_NOTE": ""
                },
                "orderDate": 1664305771089,
                "firstName": "Maria",
                "lastName": "S.",
                "activeFlags": {
                  "operatorEmail": "likeashipburgers@gmail.com",
                  "datetime": 1664305818588,
                  "orderId": "1962675",
                  "key": "ACCEPT",
                  "operator": "Like a Ship Burgers"
                },
                "userId": "388504",
                "paymentMethod": "Money",
                "available": true,
                "trackingUrl": "https://www.airmenu.com/?r=1656864470290&tracking=j9Zr55AiwWbpN/D/dvwrAQ%3D%3D",
                "menuRelation": "family",
                "title": "Menu",
                "plu": ""
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Example 2:

```
{
  "Matosinhos": {
    "childs": [
      {
        "childs": [
          {
            "childs": [
              {
                "childs": [
                  {
                    "count": 1,
                    "available": true,
                    "menuRelation": "item",
                    "title": "Coca-Cola",
                    "price": 2,
                    "auxPrice": 0,
                    "id": "4699657",
                    "plu": "PLU1034",
                    "tax": 23
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Bebidas",
                    "plu": ""
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Like a Ship Francesinhas",
                    "plu": "Francesinhas Like a Ship"
                  },
                  {
                    "available": true,
                    "menuRelation": "family",
                    "title": "Root",
                    "plu": ""
                  }
                ],
                "referencedEnterpriseName": "Like a Ship (Menu Geral Francesinhas)",
                "referencedEnterprisePhone": "934894045",
                "referencedEnterpriseAddress": "Rua Nova Estádio N°306",
                "referencedEnterpriseCity": "Matosinhos",
                "referencedEnterpriseCountry": "Portugal",
                "referencedEnterpriseLatitude": 38.7027632099636,
                "referencedEnterpriseLongitude": -9.398674025231925,
                "available": true,
                "menuRelation": "enterprise",
                "title": "Like a Ship (Menu Geral Francesinhas)",
                "plu": ""
              },
              {
                "referencedEnterpriseId": "1656930854748"
              }
            ],
            "username": "bfecd62906-3bfd-43d4-99d4-d716320d01c8@anonymous.com",
            "orderCounter": 155,
            "orderId": 1945118,
            "extraInfo": {
              "AM_TAX_PAYER_NUMBER": "",
              "AM_PROVIDER_ORDER_ID": "TV7FL",
              "AM_PROVIDER_NAME": "BoltFood",
              "AM_ORDER_TYPE": "delivery",
              "AM_NOTE": "Precisa de talheres por favor! Obrigada!"
            },
            "orderDate": 1663346745003,
            "firstName": "Bruna",
            "lastName": "M.",
            "activeFlags": {
              "operatorEmail": "likeashipburgers@gmail.com",
              "datetime": 1663346772509,
              "orderId": "1945118",
              "key": "ACCEPT",
              "operator": "Like a Ship Burgers"
            },
            "userId": "387521",
            "paymentMethod": "Money",
            "available": true,
            "trackingUrl": "https://www.airmenu.com/?r=1656864470290&tracking=LVSpCj3DWf2K/l/zXEjtjA%3D%3D",
            "menuRelation": "family",
            "title": "Menu",
            "plu": "",
            "childs": [
              {
                "childs": [
                  {
                    "childs": [
                      {
                        "available": true,
                        "menuRelation": "item",
                        "title": "Discount",
                        "price": 0,
                        "auxPrice": 0,
                        "id": "4750081",
                        "plu": "discount"
                      },
                      {
                        "count": 1,
                        "extraInfo": {
                          "AM_PROMO_VALUE": "-2.8",
                          "AM_PROMO_DESC": "Bolt discount 20.0%"
                        },
                        "available": true,
                        "menuRelation": "complexItem",
                        "title": "Promoção",
                        "price": -2.8,
                        "auxPrice": -2.8,
                        "id": "4750080",
                        "plu": "AM_PROMO"
                      }
                    ],
                    "available": true,
                    "menuRelation": "family",
                    "title": "Delivery Platform Link Items",
                    "plu": ""
                  }
                ],
                "username": "bfecd62906-3bfd-43d4-99d4-d716320d01c8@anonymous.com",
                "orderCounter": 155,
                "orderId": 1945118,
                "extraInfo": {
                  "AM_TAX_PAYER_NUMBER": "",
                  "AM_PROVIDER_ORDER_ID": "TV7FL",
                  "AM_PROVIDER_NAME": "BoltFood",
                  "AM_ORDER_TYPE": "delivery",
                  "AM_NOTE": "Precisa de talheres por favor! Obrigada!"
                },
                "orderDate": 1663346745003,
                "firstName": "Bruna",
                "lastName": "M.",
                "activeFlags": {
                  "operatorEmail": "likeashipburgers@gmail.com",
                  "datetime": 1663346772509,
                  "orderId": "1945118",
                  "key": "ACCEPT",
                  "operator": "Like a Ship Burgers"
                },
                "userId": "387521",
                "paymentMethod": "Money",
                "available": true,
                "trackingUrl": "https://www.airmenu.com/?r=1656864470290&tracking=LVSpCj3DWf2K/l/zXEjtjA%3D%3D",
                "menuRelation": "family",
                "title": "Menu",
                "plu": "",
                "childs": [
                  {
                    "childs": [
                      {
                        "childs": [
                          {
                            "childs": [
                              {
                                "childs": [
                                  {
                                    "childs": [
                                      {
                                        "available": true,
                                        "menuRelation": "complementItem",
                                        "title": "Picante",
                                        "price": 0,
                                        "auxPrice": 0,
                                        "id": "4699678",
                                        "plu": "PLUA103"
                                      },
                                      {
                                        "available": true,
                                        "menuRelation": "complement",
                                        "title": "Picante/Sem Picante",
                                        "plu": ""
                                      },
                                      {
                                        "count": 1,
                                        "available": true,
                                        "menuRelation": "item",
                                        "title": "Francesinha à Ship",
                                        "price": 12,
                                        "auxPrice": 0,
                                        "id": "4699640",
                                        "plu": "PLU1010",
                                        "tax": 13
                                      }
                                    ],
                                    "available": true,
                                    "menuRe
```

lation": "family", "title": "Francesinhas Like a Ship", "plu": ""}, {"available": true, "menuRelation": "family", "title": "Like a Ship Francesinhas", "plu": "Francesinhas Like a Ship"}], {"available": true, "menuRelation": "family", "title": "Root", "plu": ""}, {"referencedEnterpriseName": "Like a Ship (Menu Geral Francesinhas)", "referencedEnterprisePhone": "934894045", "referencedEnterpriseAddress": "Rua Nova Estádio N°306", "referencedEnterpriseCity": "Matosinhos", "referencedEnterpriseCountry": "Portugal", "referencedEnterpriseLatitude": 38.7027632099636, "referencedEnterpriseLongitude": -9.398674025231925, "available": true, "menuRelation": "enterprise", "title": "Like a Ship (Menu Geral Francesinhas)", "plu": "", "referencedEnterpriseId": "1656930854748"}], {"username": "bfecd62906-3bfd-43d4-99d4-d716320d01c8@anonymous.com", "orderCounter": 155, "orderId": 1945118, "extraInfo": [{"AM_TAX_PAYER_NUMBER": "", "AM_PROVIDER_ORDER_ID": "TV7FL", "AM_PROVIDER_NAME": "BoltFood", "AM_ORDER_TYPE": "delivery", "AM_NOTE": "Precisa de talheres por favor! Obrigada!"}], "orderDate": 1663346745003, "firstName": "Bruna", "lastName": "M.", "activeFlags": [{"operatorEmail": "likeashipburgers@gmail.com", "datetime": 1663346772509, "orderId": "1945118", "key": "ACCEPT", "operator": "Like a Ship Burgers"}], "userId": "387521", "paymentMethod": "Money", "available": true, "trackingUrl": "[https://www.airmenu.com/?r=1656864470290&tracking=LVSpCj3DWf2Kl/zXEjtjA%3D%3D](\"https://www.airmenu.com/?r=1656864470290&tracking=LVSpCj3DWf2Kl/zXEjtjA%3D%3D\")", "menuRelation": "family", "title": "Menu", "plu": ""}]}

4.5 - Authentications

After the integration, the customer can directly access the link:

<https://airmenu.com/Configure.html?linkConfiguration={enterpriseId}&sessionId={sessionId}>

You must generate a sessionId and provide this link to your customer so that it can authenticate across different platforms.

You should replace {enterpriseId} and {sessionId} with correct values.