AirMenu

API

Version 1.0.0

05-12-2018

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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={“username”:”user@mail.com”,”password”:”w8D6sc5h”}

Encoded:

www.airmenu.com%2FAirMenuAPI%3FACTION%3DAuthenticate%26VERSION%3D1.0

.0%26KEY%3D2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d%26DATA%3D%7B%E2%80%9Cusername%E2%80%9D%3A%E2%80%9Duser%40mail.com%E2%80%9D%2C%E2%80%9Dpassword%E2%80%9D%3A%E2%80%9Dw8D6sc5h%E2%80%9D%7D

2 - Actions

2.1 - CreateUser

Allows to create users.

2.1.1 - JSON Parameters

- username (string)
- password (string)

All parameters are required. 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

ACTION=CreateUser&VERSION=1.0.0&KEY=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"username":"user@mail.com", "password":"drowssap"}
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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"username":"user@mail.com", "password":"drowssap"}
RESULT={"sessionId":"1vusjr9z0ev2o","enterpriseIds":["1260841159174"],”enterpriseNames”:[“DemoEnterprise”],"action":"Authenticate","version":"1.0.0","status":"SUCCESS","statusCode":1}
```

2.3 - SaveEnterprise

Creates or saves enterprise and return its id.

2.3.1 - JSON Parameters

- sessionId (string)
- enterpriseId (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)

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

apiUser activates a function that allow 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 this parameters are defined, a parent division with name “Room” is created with 24 child divisions.

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

- restaurant
- hotel
- delivery
- golf
- other

2.3.2 - JSON Response

- enterpriseId (string)

2.3.3 - Example

```
ACTION=SaveEnterprise&VERSION=1.0.0&KEY=2aac0c9cf0d05c27df6175a
39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"name":"Rest
aurant", "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={"enterpriseId":"1360841419361","action":"SaveEnterprise","version":"1
.0.0","status":"SUCCESS","statusCode":1}
```

2.4 - SaveMenu

Saves a tree of products.

2.4.1 - JSON Parameters

- sessionId
- enterpriseId (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)
- description (string)
- menuRelation (string)
- price (double)
- plu (string)
- childs (array[menuInstance])
- min (integer)
- max (integer)
- priceCalculation (string)
- tax (double)

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

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

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.

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

2.4.2 - JSON Response

No data is returned.

2.4.3 - Example

```
ACTION=SaveMenu&VERSION=1.0.0&KEY=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"sessionId":"1vusjr9z0ev2o", "enterpriseId":"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,"menuRelation":"complexItem","priceCalculation":"max","childs":[{"plu":"c2","title":"CHOOSE_A","description":"","min":0,"max":1,"price":2,"menuRelation":"complexItem","priceCalculation":"fixed","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":"complexItem","priceCalculation":"sum","childs":[{"plu":"c6","title":"OPTION1","description":"","price":1,"menuRelation":"item"},{"plu":"c7","title":"OPTION5","description":"","price":3,"menuRelation":"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)
- enterpriseId (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***.

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)
- employee (string)
- employeePwd (string)
- extraInfo (object)

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.

2.5.3 - Example

```
ACTION=GetOrders&VERSION=1.0.0&KEY=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={“enterpriseId”:”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","menuRelation":"complementItem","childs":[]},{"title":"Chips","plu":"5","menuRelation":"complementItem","childs":[]}]}]}]}]}}
```

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)
- enterpriseId (string)

All 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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={“sessionId”:”ashdub3445s28b98xXsF”,”enterpriseId”:”123456789”}
```

```
RESULT={"host":"http://ec2-54-235-235-174.compute-1.amazonaws.com/","port":"8001","token":"123456789#13634379146804953","action":"GetNotificationAccess","version":"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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={“sessionId”:”ashdub3445s28b98xXsF”,”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)
- enterpriseId (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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"sessionId":"e45jshs3ksj","enterpriseId":"123456789","nodesToDelete":[{"plu":1,"title":"Appetizers","menuRelation":"family"},{"plu":21,"title":"PORTUGUESE STYLE CLAM STEW","description":"pork belly, fried potatoes","menuRelation":"item"}]}
```

```
RESULT={"action":"DeleteMenuNode","version":"1.0.0","status":"SUCCESS","statusCode":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

- enterpriseId (string)

2.11.2 - JSON Response

- divisions (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.

2.11.3 - Example

ACTION=GetEnterpriseDivisionIds&VERSION=1.0.0&DATA={“sessionId”:”e45jshs3ksj”,”enterpriseId”:”123456789”}

RESULT={"action":"GetEnterpriseDivisionIds","version":"1.0.0","status":"SUCCESS","statusCode":1,"divisions":{"Map1":"1234","Map2":"4321","RoomA":"9876","RoomB":"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

- enterpriseId (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":"ItemF","description":"","price":5.2,"thumbnail":"","plu":"","itemId":"6925","tax":21.0},{"childs":[],"menuRelation":"item","title":"ItemE","description":"","price":5.0,"thumbnail":"","plu":"","itemId":"6318","tax":23.0}],"menuRelation":"complexItem","title":"ComplexC","description":"","price":0.0,"thumbnail":"","min":1,"max":1,"priceCalculation":"sum","plu":"","itemId":"9697","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":"ComItemA","price":0.3,"plu":"","itemId":"4812","tax":0.0},{"childs":[],"menuRelation":"complementItem","title":"ComItemB","price":0.2,"plu":"","itemId":"1129","tax":0.0}],"menuRelation":"complement","title":"CompA","min":1,"max":1,"plu":"","itemId":"9783"}],"menuRelation":"item","title":"ItemB","description":"","price":1.0,"thumbnail":"","plu":"","itemId":"1566","tax":23.0}],"menuRelation":"family","title":"FamilyB","description":"","thumbnail":"","plu":"","itemId":"9232"}],"menuRelation":"family","title":"FamilyA","description":"","thumbnail":"","plu":""

,"itemId":"2332"}],"divisionId":"4321","menuStartTime":518400000,"menuEndTime":604800000,"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

- enterpriseId (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 **enterpriseId**, **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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"sessionId":"1917lh4p0ewv6","enterpriseId":"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

- enterpriseId (string)
- code (string)
- userId (string)
- cultureCode (string)

enterpriseId, **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=2aac0c9cf0d05c27df6175a39068d5e55e3bfbb5d51c7b41da434499951be78d&DATA={"sessionId":"1917lh4p0ewv6","enterpriseId":"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

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

Both **enterpriseId** 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) and indicates in which time period this lot is available. **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=2aac0c9cf0d05c27df6175a3906
8d5e55e3bfbb5d51c7b41da434499951be78d&DATA={“sessionId”:”fffuufjdeennn
”,”enterpriseId”:”1360841419361”,”lots”:[{“type”:”consumable”,”name”:”autom”,”st
art”:1554022419000,”expiration”:1554022519000,”plu”:”AM115”,”qty”:15}]}
```

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 stablish 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 stablished, 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 succeed, 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 have 30 seconds of time out to be associated with an enterprise.

3.2 - Notifications

Notification 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**).