



# SpotOption API Documentation

## Contents of the document

1. General Information .....	7
1.1 Prerequisites .....	8
1.1.1 API URL .....	8
1.1.2 API User .....	8
1.2 Record Limits .....	9
1.2.1 Default Record Limit .....	9
1.2.2 Manual Record Limits .....	9
1.3 BATCH .....	10
1.3.1 Examples .....	10
1.4 FILTER .....	11
Simple Search .....	11
Min & Max .....	11
Advanced Searches (OR, AND) .....	11
FILTER OUT (NOT) .....	11
1.5 LIMIT .....	13
1.5.1 Examples: .....	13
1.6 ORDER BY .....	14
1.6.1 Examples: .....	14
2. API Modules .....	15
2.1 Customer .....	16
2.1.1 Fields .....	16
2.1.2 Examples: .....	18
2.1.3 Status messages .....	19
2.2 Lead .....	22
2.2.1 Fields .....	22
2.2.2 Examples .....	23
2.2.3 Status messages .....	24
2.3 Positions .....	25
2.3.1 Add/View Fields .....	25
2.3.2 Rollover Fields .....	26
2.3.3 Cancel Fields .....	26
2.3.4 Examples .....	26
2.3.5 Status messages .....	27
2.4 CreditCardUser .....	30
2.4.1 Fields .....	30
2.4.2 Examples .....	31

2.4.3 Status messages.....	31
2.5 CustomerDeposits .....	32
2.5.1 Fields .....	32
2.5.3 Edit Fields .....	34
2.5.3 Examples .....	34
2.5.4 Status messages.....	35
2.6 Options.....	37
2.6.1 Fields .....	37
2.6.2 Examples .....	38
2.6.3 Status messages.....	39
2.7 Withdrawals.....	40
2.7.1 Fields .....	40
2.7.2 Examples .....	42
2.7.3 Status messages.....	42
2.8 Assets .....	44
2.8.1 Fields .....	44
2.8.2 Examples .....	44
2.8.3 Status messages.....	44
2.9 AssetsHistory .....	45
2.9.1 Fields .....	45
2.9.2 Examples .....	45
2.9.3 Status messages.....	45
2.10 Country.....	46
2.10.1 Examples .....	46
2.10.2 Status messages.....	46
2.11 Campaign.....	47
2.11.1 Fields .....	47
2.11.2 Examples .....	47
2.11.3 Status messages.....	47
2.12 Sub Campaigns .....	49
2.12.1 Fields .....	49
2.12.2 Examples .....	49
2.12.3 Status messages.....	49
2.13 Calls .....	50
2.13.1 Fields .....	50
2.13.2 Examples .....	50
2.13.3 Status messages.....	50

2.14 Rules .....	51
2.14.1 Examples .....	51
2.14.2 Status messages.....	51
2.15 DepositsLog .....	52
2.15.1 Fields .....	52
2.15.2 Examples .....	52
2.15.3 Status messages.....	53
2.16 CampaignStats .....	54
2.16.1 View .....	54
2.16.2 Edit.....	54
2.16.3 Examples .....	54
2.16.4 Status messages.....	55
2.17 SubCampaignStats.....	56
2.17.1 View .....	56
2.17.2 Edit.....	56
2.17.3 Examples .....	56
2.17.4 Status messages.....	57
2.18 ExternalCampaign .....	58
2.18.1 Add .....	58
2.18.2 Edit.....	58
2.18.3 Parameters .....	59
2.18.4 Examples .....	60
2.18.5 Status messages.....	60
2.19 ClickedAssets .....	61
2.19.1 Add .....	61
2.19.2 Examples .....	61
2.19.3 Status messages.....	62
2.20 CustomerData .....	63
2.20.1 View .....	63
2.20.2 Add .....	63
2.20.3 Examples .....	63
2.20.4 Status messages.....	64
2.21 Regulation .....	65
2.21.1 View .....	65
2.21.2 Add .....	65
2.21.3 Examples .....	65
2.21.4 Status messages.....	66

2.22 OneTouch.....	67
2.22.1 View.....	67
2.22.2 Add .....	67
2.22.3 Examples .....	67
2.22.4 Status messages.....	68
2.23 BMOLog .....	69
2.23.1 View .....	69
2.23.2 Examples .....	69
2.23.3 Status messages.....	69
2.24 User.....	70
2.24.1 Fields .....	70
2.24.2 Examples .....	71
2.24.3 Status messages.....	71
2.25 AssetLog .....	73
2.25.1 View .....	73
2.25.2 Examples .....	73
2.25.3 Status messages.....	73
2.26 CurrencyHistory .....	74
2.26.1 .....	74
2.26.2 Status messages.....	74
2.27 LDO .....	75
2.27.1 Examples .....	75
2.27.2 Status messages.....	75
2.28 CustomerLoginLog .....	76
2.28.1 Examples .....	76
2.28.2 Status messages.....	76
2.29 LDOCustomerLimits .....	77
2.29.1 Fields .....	77
2.29.2 Examples .....	77
2.29.3 Status messages.....	78
2.30 Affiliate .....	79
2.30.1 Fields .....	79
2.30.2 Examples .....	79
2.30.3 Status messages.....	79
2.31 CustomSettings .....	80
2.31.1 Fields .....	80
2.31.2 Examples .....	80

2.31.3 Status messages.....	80
2.32 SpotFollow.....	81
2.32.1 Fields .....	81
2.32.2 Examples .....	82
3. Code Examples.....	85
3.1 HTML .....	86
3.2 PHP .....	87
3.3 C#.....	88
4. Appendix.....	89
Common API Errors.....	89

# 1. General Information

SpotOption's API is a RESTful web API, to which you POST data to, the API response is an XML or JSON output.

The XML is the default response, in order to get JSON response this param and value should be added to the request: `jsonResponse='true'` (the true must be string)

In general, the actions you can perform are usually compatible with the CRUD cycle (Search, Create, Read, Update and Delete).

In this document we will survey how to access, and use the API.

The following pages will contain examples of common usages of the API, a complete revision of the API Modules, code examples, and other important information.

NOTE: This document is under constant work, so if you feel that something important is missing in this document, or if you notice something is incorrect with the data in this document, feel free to contact us, at SpotOption.

## 1.1 Prerequisites

To use the API, you require several things:

1. a SpotOption platform
2. The API URL - for example, if you platform's URL is <http://www.your-platform.com>, than your API URL would usually be at <http://www.api.your-platform.com/Api>  
If you are using Spot.2 platform your API URL is: <http://api-spotplatform.your-platform.com/Api>
3. API username , with the required permissions

### 1.1.1 API URL

You should contact SpotOption If you see that the API URL isn't working, a simple check would be clicking on the URL, and seeing this XML Response:

```
- <status>
  - <errors>
    <error>parametersMissing</error>
  </errors>
</status>
```

If you see this, it means the API is working, and that your URL is correct.

### 1.1.2 API User

An API username consists of credentials (username + password), and Permissions (which actions it's allowed to perform).

#### API credentials

To get live API credentials, you will need to create an API User in the CRM of the platform / Have someone with CRM access create it for you.

If you wish to get demo credentials to use the API, in order to test things, Please contact SpotOption.

#### API Permissions & IP White List

When you create a new API user, you will need to give permissions to modules (to perform different actions), and to set the IP white list, from which you will be accessing the API.

By default, if an IP address isn't white listed for that API user – all requests to the API from that machine for that API user, will fail.



## 1.2 Record Limits

### 1.2.1 Default Record Limit

If you see that not all of the data you needed is returned after a view command, perhaps it is because of the response size-limitations,

At the moment it is set to 500 (i.e. you will get 500 items on every request, if we do have 500 items for you request - that is), but this number can change in the future.

If you wish to view the rest of the data, you could view it by posting the "page" param, with the number of page you wish to view (if you don't post the page param, you will get the first 500 results by default).

#### 1.2.1.1 Examples

##### **Getting page 2 of all of your customers**

```
MODULE=Customer&COMMAND=view&page=2
```

##### **Getting page 1 of all of your customers**

###### **Variation a**

```
MODULE=Customer&COMMAND=view&page=1
```

###### **Variation b**

```
MODULE=Customer&COMMAND=view
```

### 1.2.2 Manual Record Limits

It is possible to set up manual record limits (as long as they don't overpass the default record limits), to understand how to use this feature, please refer to section 1.5 of this document.

## 1.3 BATCH

If you want to post several actions at once, you can POST them in one request using BATCH, instead of making several requests – which will save the roundtrips.

The idea is to wrap each action, in an array key, and your final post would be an array of commands.

You will only need to post the credentials just once for this request, And Send the BATCH array...

### 1.3.1 Examples

**Making 3 actions, in one request:**

- a. Viewing Live Options**
- b. Viewing Customer 1**
- c. Viewing Position 100**

```
BATCH[0][MODULE]=Options&BATCH[0][COMMAND]=view&BATCH[1][MODULE]=Customer&BATCH[1][COMMAND]=view&BATCH[1][FILTER][id]=1&BATCH[2][MODULE]=Positions&BATCH[2][COMMAND]=view&BATCH[2][FILTER][id]=100
```

## 1.4 FILTER

When using the command VIEW (the need to search for a specific set of data in one of the modules), we will use the FILTER array, to hold the search criteria.

### Simple Search

If we were to search for a customer (Using the Customer Module, with command VIEW), who's Id is 1, and we will need to use this FILTER array:

```
FILTER[id]=1
```

The result will hold only 1 customer (id = 1), or 0 customer (in cases we don't have a customer with this id)

### Min & Max

If we were to search for positions (Using the Positions Module, with command VIEW), that were placed between 2 dates, we will need to use the min & max FILTER – like this:

```
FILTER[date][min]=2013-06-01&FILTER[date][max]=2013-06-05
```

The result will hold all positions between the 1<sup>st</sup> & the 5<sup>th</sup> of June, 2013 (unless we surpass the allowed result limit, which is 500, in that case – refer to Section 1.2 of this document)

### Advanced Searches (OR, AND)

By default, all the FILTER keys use the AND Command (all FILTER criteria must be met), for example, if we were to perform the VIEW action on the Customer Module with this FILTER:

```
FILTER[id]=1&FILTER[regTime][min]=2013-06-05
```

The response will hold the customer whose Id is 1, only if he registered after June 5<sup>th</sup> 2013, as if both conditions were not met, a response of 'noResults' will be replied.

In some cases we are able to use OR command, inside the FILTER Array, By Creating a sub-array inside the FILTER criteria - in the following way:

```
FILTER[id][]=1&FILTER[id][]=2&FILTER[regTime][min]=2013-06-05
```

This will basically search for customers whose Id is 1 OR 2, AND have registered after June 5<sup>th</sup> 2013, and this allows us to widen the search, without performing unnecessary calls to the API.

### FILTER OUT (NOT)

This filter allows us to get results which are different from the sent items. For example we can exclude demo customers from our query by sending the filter not array.

```
FILTER[NOT][isDemo] = 1
```

This will produce a result without demo users.

Furthermore, this filter can be used to eliminate multiple fields from our results.

```
FILTER[NOT][status][0] = 'open' & FILTER[NOT][status][1] = 'lost'
```

This will produce results without positions in status open or lost.

## 1.5 LIMIT

By default, each response is limited to X amount of records (usually 500), but it is possible to set custom records limitation, for every request – e.g. to receive only 10 records, even if there are more in the result set.

There are 2 parameters to be passed, Inside the LIMIT array:

1. recordsToShow – how many records you wish to receive in the response  
i.e. the size of the result set
2. recordStart (Optional) – from which record do you want the results to start  
e.g. if the complete result set is 500, and you wish to receive 10 results (set by 'recordsToShow'), starting with result number 100 (set by recordStart), it will show you records 100-110...

if you wish the results to start from the first record available, you can either post 0 as the recordStart, or simply don't post it, at all (it is optional).

This feature is very much like the MySQL LIMIT command.

### 1.5.1 Examples:

An example of getting the first 5 customers (by default it would have returned the first 500):

```
MODULE=Customer&COMMAND=view&LIMIT[recordsToShow]=5&LIMIT[recordStart]=0
```

When using VIEW commands, the results are sorted by id, e.g. in the customer Module it's the customer id, in the Positions Module it's position Id, etc.

Another Example:

```
MODULE=Customer&COMMAND=view&FILTER[id]=1&LIMIT[recordsToShow]=1&LIMIT[recordStart]=1
```

In this case, we will receive 0 results, even if we have a customer with id 1, since the recordStart is 1, and the recordsToShow is 1. Basically it searched for the 2<sup>nd</sup> customer found in the result set, but since we have only 1 – it will return 'noResults'.

NOTE A: the requested LIMIT[recordsToShow] cannot surpass the default limit set (you cannot request 510 results, for example), in case it does, the default limitations will be used.

NOTE B: you cannot use the page command (discussed in section 1.2 of this document), when using the custom record limitation.

## 1.6 ORDER BY

By default, each response is ordered by A to Z, but there is a possibility to set it in the opposite order (Z to A).

The parameter that in charge of the order is: sortOrder.

The value can be either **a-z** (ascending as default, no need to send it) or **z-a**, if you want the order to be descending.

### **1.6.1 Examples:**

An example to getting descending sort:

```
MODULE=Positions&COMMAND=view&FILTER[customerId] =3&ORDER[sortOrder]=z-a
```

In this case we get the positions sorted in descending order.

NOTE A: When using VIEW commands, the results are sorted by id, e.g.  
In the Positions Module it's position Id.

NOTE B: if you want the order to be ascending, there is no need to send this parameter, as the ascending order is the default.

## 2. API Modules

The API contains several modules, and on each module you can perform several actions.

This is the largest section of this document, as it contains a complete revision of all modules, of the API.

In this section you will be able to view the fields, commands, and watch examples on how to perform common tasks using the API.

In general, the actions in the modules are usually compatible with the CRUD cycle (Create, Read, Update and Delete).

## 2.1 Customer

### 2.1.1 Fields

Field	Type	Default	Mandatory (add customer)	Editable	Comments
verification	enum('None', 'Partial', 'Full')	None	✗	✓	
FirstName	varchar(45)		✓	✓	
LastName	varchar(45)		✓	✓	
gender	enum('male', 'female')		✗	✓	
email*	varchar(255)		✓	✓	
password*	varchar(32)		✓	✓	
authKey	varchar(32)		✗	✓	
authKeyExpiry	TIMESTAMP		✗	✗	
cellphone	varchar(20)		✗	✓	
Phone	varchar(20)		✓	✓	
fax	varchar(20)		✗	✓	
personalId	varchar(16)		✗	✓	The personalId can be used for passport id as well
specialAccountNumber	varchar(32)		✗	✓	
Country	smallint(4)		✓	✓	
registrationCountry	smallint(4)		✗	✓	Not empty or null
State	varchar(2)		✗	✓	
City	varchar(45)		✗	✓	
street	varchar(45)		✗	✓	
houseNumber	smallint(6)		✗	✓	
aptNumber	smallint(6)		✗	✓	
postCode	varchar(10)		✗	✓	
approvesEmailAds	tinyint(1)		✗	✓	
campaignId	int(11)		✗	✓	
subCampaign	varchar(100)		✗	✓	
regTime	timestamp	CURRENT_TIMESTAMP	✗	✗	
lastTimeActive	timestamp	0000-00-00 00:00:00	✗	✓	
lastUpdate	timestamp	0000-00-00 00:00:00	✗	✗	



<b>birthday</b>	date		✗	✓	
<b>accountBalance</b>	float(10,2)		✗	✗	
<b>currency</b>	enum('USD', 'EUR')		✓	✓	On edit - possible only to customers that didn't deposit yet.
<b>potential</b>	enum('low', 'medium', 'high')		✗	✓	
<b>risk</b>	enum('low', 'medium', 'high')	low	✗	✓	
<b>employeeInChargedId</b>	int(11)		✗	✓	
<b>regStatus</b>	enum('pending', 'activated')		✗	✓	
<b>isDemo</b>	tinyint(1)	0	✗	✓	
<b>saleStatus</b>	enum('new', 'noAnswer', 'checkNumber', 'callAgain', 'notIntrested', 'inTheMoney', 'noCall')		✗	✓	
<b>leadStatus</b>	enum('new', 'noAnswer', 'checkNumber', 'callAgain', 'notIntrested', 'inTheMoney', 'noCall')		✗	✓	
<b>promotionalEmails</b>	Int(1)	0	✗	✓	
<b>tradingEmails</b>	Int(1)	0	✗	✓	
<b>regulateStatus</b>	enum('none', 'pending', 'suspend', 'approve')		If regulated site ✓ if not ✗	✓	
<b>regulateType</b>	Int(11)		If regulated site ✓ if not ✗	✓	Id of the regulation
<b>vipGroup</b>	enum('VIP', 'Gold', 'Silver', 'Regular')		✗	✓	
<b>a_aid</b>	varchar(255)		✗	✓	affiliate id
<b>a_bid</b>	varchar(255)		✗	✓	banner id
<b>a_cid</b>	varchar(255)		✗	✓	Campaign id
<b>isSuspended</b>	tinyint(1)	0	✗	✓	
<b>groupId</b>	tinyint(3)		✗	✓	from customer_groups
<b>siteLanguage</b>	varchar	language s.default	✗	✓	
<b>referLink</b>	text		✗	✓	
<b>contactByCoach</b>	bit(1)	0	✗	✗	
<b>timezone</b>	text		✗	✓	
<b>regIP</b>	varchar		✗	✓	

<b>affiliateId</b>	Int(11)	0	✗	✓	Affiliate id
<b>sendEmail</b>	Tinyint(1)	-	✗	✗	
<b>employmentStatus</b>	varchar		✗	✓	EDIT ONLY
<b>nationality</b>	Int(11)		✗	✓	EDIT ONLY

\* **Required for "validate"**

- When passing a **password** field to the filter, it automatically encrypts it.
- When passing a **subCampaign** field on **add**, a sub campaign will automatically be created, and the add will also return the created sub campaign-id

## 2.1.2 Examples:

### Adding a new customer

```
MODULE=Customer&COMMAND=add&FirstName=Test&LastName=Customer&email=test1@SpotOption.com&password=123456&Country=226&campaignId=7&subCampaign=45&currency=USD&a_aid=4c1784efdccc&b&birthday=1980-07-21
```

### Verifying customer login credentials using the view model

```
MODULE=Customer&COMMAND=view&FILTER[email]=lior@iokun.com&FILTER[password]=123456
```

### Verifying customer login credentials using the validate model (lighter)

```
MODULE=Customer&COMMAND=validate&FILTER[email]=lior@iokun.com&FILTER[password]=123456
```

### Editing a Customer (using his Id -123), and making him a demo customer

```
MODULE=Customer&COMMAND=edit&customerId=123&isDemo=1&subCampaignId=45
```

### Getting customers who first deposited today or yesterday, and are not demo

```
MODULE=Customer&COMMAND=view&FILTER[firstDepositDate][min]=2010-06-01&FILTER[firstDepositDate][max]=2010-07-01&FILTER[isDemo]=0
```

### Getting a specific customer details

```
MODULE=Customer&COMMAND=view&FILTER[id]=123
```

### Filter by account balance

```
MODULE=Customer&COMMAND=view&FILTER[accountBalance]=100
```

```
MODULE=Customer&COMMAND=view&FILTER[accountBalance][min]=100&FILTER[accountBalance][max]=200
```

### 2.1.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	(edit command)	Customer was not found
<b>addFailed</b>	(add command)	General error while trying to add the record
<b>requiredFieldsMissing</b>	(add command)	Some of the required fields were not passed
<b>invalidFirstName</b>	FirstName	First name should be at 2-45 characters
<b>invalidLastName</b>	LastName	Last name should be at 2-45 characters
<b>invalidGender</b>	gender	Gender should be 'male' or 'female'
<b>invalidEmail</b>	email	Email address was not validated
<b>emailAlreadyExists</b>	email	Email address already exists in customers table
<b>passwordTooshort</b>	password	Password should be at least 6 characters
<b>invalidAuthKey</b>	authKey	Auth key token should be exactly 32 characters length (md5)
<b>invalidCellPhoneNo</b>	cellphone	Cellphone number should 20 characters max
<b>invalidPhoneNo</b>	Phone	Phone number should 20 characters max
<b>invalidPersonalId</b>	personalId	Personal id should 20 characters max
<b>invalidFaxNo</b>	fax	Fax number should 16 characters max
<b>invalidCountry</b>	Country	Country should be an integer between 1 and 239. See country list.
<b>invalidRegistrationCountry</b>	Country	Country should be an integer between 1 and 239. See country list.
<b>invalidState</b>	State	State field is less than 2 characters
<b>invalidCity</b>	City	City field is over than 2 characters
<b>invalidStreetAddress</b>	street	Street field should be less than 45 characters
<b>invalidCurrency</b>	currency	Currency used is invalid

<b>invalidHouseNo</b>	houseNumber	Field is not numeric
<b>invalidApartmentNo</b>	aptNumber	Field is not numeric
<b>invalidApprovesEmailAdsStatus</b>	approvesEmailAds	Field is not numeric
<b>invalidPostalCode</b>	postCode	Postal code should be less than 10 characters
<b>invalidCampaignId</b>	campaignId	Campaign does not exist in campaign table
<b>invalidSubCampaign</b>	subCampaign	Subcampaign parameter is not a string
<b>invalidAffiliateId</b>	a_aid	Field is not numeric
<b>invalidBirthday</b>	birthday	Field is not a string
<b>customerTooYoung</b>	birthday	Customer should be at least 18 years old
<b>invalidEmployee</b>	employeeInChargedId	Employee does not exist in users table
<b>invalidDate</b>	lastTimeActive	Field is not a string
<b>invalidPotential</b>	potential	Field value is not one of the possible values
<b>invalidRisk</b>	risk	Field value is not one of the possible values
<b>invalidRegStatus</b>	regStatus	Field value is not one of the possible values
<b>invalidSaleStatus</b>	saleStatus	Field value is not one of the possible values
<b>invalidCustomerGroup</b>	groupId	Field value is not one of the possible values
<b>invalidReferLink</b>	groupId	The refer link is invalid
<b>invalidIsDemo</b>	isDemo	isDemo should be 0 or 1
<b>invalidSpecialAccountNumber</b>	specialAccountNumber	Field should be String, no longer than 32 chars
<b>specialAccountNumberExists</b>	specialAccountNumber	Special Account Number already exists in customers table
<b>customerIsActive</b>	isDemo (edit)	You cannot edit the demo status of customers with any financial history
<b>invalidRegIP</b>		Checks ip is valid and permitted
<b>invalidCampaignIdPermission</b>		campaigns permissions denied
<b>blockedRegistrationCountry</b>	Country	Country value doesn't exist in list
<b>invalidPromotionalEmails</b>	promotionalEmails	Must be variable int
<b>invalidTradingEmails</b>	tradingEmails	Must be variable int
<b>invalidTimeZone</b>	timezone	Does not match values from timezone list
<b>invalidSiteLanguage</b>	siteLanguage	Max length 3
<b>invalidRegulateStatus</b>	regulateStatus	Not in regulation types
<b>invalidRegulateType</b>	regulateType	Must be variable int
<b>InvalidBrokerId</b>	brokerId	Must be variable int
<b>invalidLeadConversionDate</b>	leadConversionDate	Date is not valid or collides
<b>InvalidLeadConversionSource</b>	leadConversionSource	Must be variable string
<b>InvalidRealPassword</b>	realPassword	Must be variable int
<b>InvalidVipGroupStatus</b>	vipGroup	Not in VIP group list

<b>InvalidSpecialAccountNumber</b>	specialAccountNumber	maxLength(32) , must be unique
<b>invalidCurrency</b>	currency	Must be in currency Options
<b>SpecialAccountNumberExists</b>	specialAccountNumber	Special Account Number notFound
<b>invalidServerURL</b>	serverURL	minLength(2)
<b>accountAlreadyExistsOnThisServer</b>	id	Customer was not found
<b>wrongBirthdayFormat</b>	birthday	Please check birthday format
<b>invalidIsBlackList</b>	isBlackList	Must be variable int
<b>invalidBannerId</b>	a_bid	Must be variable int
<b>invalidNationality</b>	nationality	Musr be from country option list
<b>pleaseSelectEmploymentStatus</b>	employmentStatus	Must be from employment Statuses
<b>invalidAutomaticUpdate</b>	automaticUpdate	Must be variable int
<b>invalidCustomerIdOnServer</b>	customerId	Not included in MtId
<b>noFieldsToEdit</b>		Nothing was sent
<b>updateFailed</b>		API failed
<b>customerIsActive</b>		Customer is active
<b>Validate:</b>		
<b>pleaseUpdateApp</b>		Mobile Force Update is enabled. Application requires software update
<b>InvalidCustomerDetails</b>	email, password	Email and password did not pass authentication by token
<b>validateMobile:</b>		
<b>invalidSiteLogin</b>	user, password	Login Failed
<b>pleaseUpdateApp</b>		Mobile Force Update is enabled. Application requires software update
<b>the parameters are not valid</b>		validateMobile failed
<b>SiteBalance:</b>		
<b>invalidIntegrationSite</b>	customerId , balanceType	File for integration is missing (Api.php)
<b>invalidBalanceType</b>	balanceType	balanceType does not match balance type for label
<b>the parameters are not valid</b>		SiteBalance failed
<b>error getting the balance</b>		Unable to get balance
<b>Open External Account:</b>		
<b>pleaseUpdateApp</b>		Mobile Force Update is enabled. Application requires software update
<b>requiredFieldsMissing</b>	lang, Country, isDemo, FirstName, LastName, email, phone_area, phone_prefix, Phone, password, currency, campaignId	Make sure all fields are filled correctly!
<b>errorMsg*</b>		Open External Account result

## 2.2 Lead

### 2.2.1 Fields

Field	Type	Default	Mandatory (add Lead)	Optional (add Lead)	Editable	Comments
verification	enum('None', 'Partial', 'Full')	None				
leadId	Int	-	✗	✗	✗	Mandatory - editable
FirstName	varchar(45)	-	✓	✗	✓	
LastName	varchar(45)	-	✓	✗	✓	
gender	enum('male', 'female')	-	✗	✗	✓	
email	varchar(255)	-	✓	✗	✓	
cellphone	varchar(20)	-	✗	✗	✓	
Phone	varchar(20)	-	✗	✓	✓	
fax	Int(16)	-	✗	✗	✓	
Country	smallint(4)	-	✗	✓	✓	
registrationCountry	smallint(4)	-	✗	✓	✗	
State	varchar(2)	-	✗	✗	✓	
City	varchar(45)	-	✗	✗	✓	
street	varchar(45)	-	✗	✗	✓	
houseNumber	varchar(10)	-	✗	✗	✓	
postCode	varchar(10)	-	✗	✗	✓	
campaignId	int(11)	-	✗	✓	✓	
campaignName	Varchar(150)		✗	✗	✗	
employeeInChargedId	int(11)	-	✗	✓	✓	
subCampaignId	Int(11)		✗	✓	✗	
subCampaign	varchar(100)	-	✗	✓	✗	param
birthday	date	-	✗	✗	✓	
referLink	text	-	✗	✓	✓	
isLead	Tinyint(1)	1	✗	✗	✗	
newCustomer	Tinyint(1)	-	✗	✗	✗	
a_aid	varchar(255)	-	✗	✓	✓	affiliate id
personalId	varchar(20)	-	✗	✗	✓	
aptNumber	varchar(10)	0	✗	✗	✓	
approvesEmailAds	tinyint(1)	-	✗	✗	✓	

<b>lastUpdateFormatted</b>	Timestamp	-	✗	✗	✗	
<b>lastTimeActive</b>	timestamp	0000-00-00 00:00:00	✗	✗	✓	
<b>regTime</b>	timestamp	0000-00-00 00:00:00	✗	✗	✗	
<b>callBackTime</b>	time	00:00:00	✗	✗	✗	
<b>contactByCoach</b>	Bit(1)	-	✗	✗	✗	
<b>saleStatus</b>	enum('new', 'noAnswer', 'checkNumber', 'callAgain', 'notIntrested', 'inTheMoney', 'noCall')	-	✗	✗	✓	
<b>leadStatus</b>	enum('new', 'noAnswer', 'callAgain', 'checkNumber', 'notSupported', 'notValidPhone')	-	✗	✗	✓	
<b>sendEmail</b>	Tinyiny(1)	-	✗	✓	✗	
<b>siteLanguage</b>	Varchar(3)	-	✗	✓	✓	
<b>regIP</b>	varchar		✗	✓	✓	

- When passing a **subCampaign** field on **add**, a sub campaign will automatically be created, and the add will also return the created sub campaign-id

## 2.2.2 Examples

### Adding a new lead

```
MODULE=Lead&COMMAND=add&FirstName=Lior&LastName=Chen&email=lior@iokun.com&campaignId=7&subCampaignId=45
```

### Getting leads who registered today or yesterday

```
MODULE=Lead&COMMAND=view&FILTER[regTime][min]=2010-06-01&FILTER[regTime][max]=2010-07-01&FILTER[isDemo]=0
```

### Editing the employeeInChargeld of Lead Id 1 to Employee Id 100

```
MODULE=Lead&COMMAND=edit&leadId=1&employeeInChargeld=100
```

### 2.2.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>addFailed</b>	(add)	General error while trying to add the record
<b>noSuchLead</b>	(edit)	no lead with give Id was found
<b>requiredFieldsMissing</b>	(add)	Some of the required fields were not passed
<b>invalidFirstName</b>	FirstName	First name should be at 2-45 characters
<b>invalidLastName</b>	LastName	Last name should be at 2-45 characters
<b>invalidEmail</b>	email	Email address was not validated
<b>emailAlreadyExists</b>	email	Email address already exists in customers table
<b>invalidPhoneNo</b>	Phone	Phone number should 20 characters max
<b>invalidCountry</b>	Country	Country should be an integer between 1 and 239. See country list.
<b>invalidCampaignId</b>	campaignId	Campaign does not exist in campaign table
<b>invalidSubCampaign</b>	subCampaign	Subcampaign parameter is not a string
<b>invalidAffiliateId</b>	a_aid	Field is not numeric
<b>invalidReferLink</b>	groupId	The refer link is invalid
<b>invalidRegistrationCountry</b>	registrationCountry	Country should be an integer between 1 and 239. See country list.
<b>blockedRegistrationCountry</b>	Country	Not in Allowed Registration Countries list
<b>invalidSiteLanguage</b>	siteLanguage	maxLength(3)
<b>invalidEmployeeInChargeId</b>	employeeInChargeId	Not in employees Options
<b>invalidRegIP</b>		Checks ip is valid and permitted



## 2.3 Positions

Using this module, you could view, add (create), cancel, and Rollover a position

### 2.3.1 Add/View Fields

Field	Type	Null	Default	Mandatory (Add Position)	Comments
position	enum('call', 'put')	No		✓	
customerId	int(11)	No		✓	
rate	decimal(20,8)	No	0.00000000	✗	
originalRate	decimal(20,8)	Yes	NULL	✗	
optionId	int(11)	No		✓	
amount	float	No		✓	
currency	enum('EUR', 'USD')	No		✗	
status	enum('open', 'won', 'lost', 'tie')	No		✗	
date	timestamp	No		✗	
payout	float(10,2)	No		✗	
ip	Varchar(45)	No		✗	

**Linked modules:** Assets, Options, Customers

**\* For adding Option Builder position the following fields are also required**

Fields	Type	Null	Default	Comments
product	varchar	No		optionsBuilder
ruleId	int(11)	No		The id of the rule
assetId	int(11)	No		The id of the asset
expireTime	unix timestamp	No		expire time that was chosen in unix timestamp format (GMT) for example: 1324548000

**\* For adding Sixty Seconds position the following fields are also required**

Field	Type	Null	Default	Mandatory	Comments
product	varchar	No		✓	sixtySeconds
ruleId	int(11)	No		✗	The id of the rule
assetId	int(11)	No		✓	The id of the asset
expireIn	Int(3)	No	60	✗	30, 60, 90,120...

### 2.3.2 Rollover Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		The Customer Id
optionId	int(11)	No		The id of the new option (the one we want to rollover to)
assetId	int(11)	No		The id of the asset (must be the asset of the current position, and the new option)
positionId	int(11)	No		The Id of the current position

### 2.3.3 Cancel Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		The Customer Id
positionId	int(11)	No		The Id of the position we want to cancel

### 2.3.4 Examples

Filter the positions by product (sixtySeconds, optionsBuilder) using LFILTER:

```
MODULE=Positions&COMMAND=view&LFILTER[options][product]=optionsBuilder
MODULE=Positions&COMMAND=view&LFILTER[options][product]=sixtySeconds
```

Getting a customer's 60 seconds positions, filtered by the option endDate using LFILTER:

```
MODULE=Positions&COMMAND=view&FILTER[customerId]=106&LFILTER[options][endDate][min]=2012-06-15 23:59&LFILTER[options][product]=sixtySeconds
```

Getting today's positions:

```
MODULE=Positions&COMMAND=view&FILTER[date][min]=2010-07-01 0:00&FILTER[date][max]=2010-08-01 00:00
```

Getting a customer's positions

```
MODULE=Positions&COMMAND=view&FILTER[customerId]=108
```

Getting a customer's positions of a currencies assets (filter by the linked Assets module)

```
MODULE=Positions&COMMAND=view&FILTER[customerId]=108&LFILTER[Assets][type]=currencies
```

Adding an Option Builder position

```
MODULE=Positions&COMMAND=add&product=optionsBuilder&customerId=2&position=call&amount=100&ruId=3&expireTime=1324548000
```

## Adding a Sixty Seconds position

```
MODULE=Positions&COMMAND=add&product=sixtySeconds&customerId=2&position=call&amount=100&assetId=3
```

## Adding a regular position

```
MODULE=Positions&COMMAND=add&product=sixtySeconds&customerId=2&position=call&amount=100&assetId=3&expireIn=90
```

```
MODULE=Positions&COMMAND=add&position=call&customerId=14760&optionId=4594274&amount=25&product=regular
```

## Rolling-Over a Position

```
MODULE=Positions&COMMAND=rollover&customerId=10489&optionId=4594274&positionId=1520&assetId=91
```

## Cancelling a Position

```
MODULE=Positions&COMMAND=cancel&customerId=10489&positionId=1520
```

## 2.3.5 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>addFailed</b>	(add)	General error while trying to add the record
<b>invalidCustomerId</b>	customerId	Customer does not exist in customers table
<b>invalidOptionId</b>	optionId	Field is not numeric
<b>invalidPosition</b>	position	Position should be 'call' or 'put'
<b>invalidAmount</b>	amount	Field is not numeric
<b>insufficientFunds</b>	amount	The position amount is higher than the customer's balance
<b>optionSuspended</b>	optionId	The selected option is currently suspended
<b>customerSuspended</b>	customerId	The selected client is currently suspended
<b>invalidAssetRate</b>	optionId	The asset of the option currently has an invalid rate

<b>optionIsClosed</b>	optionId	The selected option is closed
<b>riskSuspensionActivated</b>	-	Risk suspension for the selected customer is in effect
<b>noTradeableNow</b>		Asset is not currently in trading hours
<b>addOptionFailed-productParameterIsNotSet</b>	-	The product parameter is not set (optionsBuilder/sixtySeconds)
<b>addOptionFailed-givenAssetIdsNotLikeInRule</b>	-	The asset id is not like the asset id in rule
<b>addOptionFailed-customerIsSuspended</b>	-	The customer is suspend
<b>addOptionFailed-assetRateInvalid</b>	-	Problem with the asset rate
<b>addOptionFailed-theAssetIsNotTradeable</b>	-	The asset is not tradeable
<b>addOptionFailed-theAssetIsSuspended</b>	-	The asset is suspended
<b>addOptionFailed-invalidAmount-shouldBeBetweenXandY</b>	-	The amount sent is illegal - should be between X = minimum and Y = maximum
<b>addOptionFailed-expireTimeIsInThePast</b>	-	The expire time of the option should be in the future
<b>addOptionFailed-expireTimeShouldBeBetweenXandY</b>	-	The expire time should be between X and Y
<b>addOptionFailed-insufficientFunds</b>	-	Insufficient funds to make position
<b>addOptionFailed-riskSuspensionActivated</b>	-	risk suspension activated
<b>noTime</b>	(rollover)	You Can't rollover this position, because noRolloverTime has passed
<b>cannotRolloverCustomOption</b>	(rollover)	Cannot Rollover An Options Builder Option
<b>numberOfRolloversExceeded</b>	(rollover)	Maximum number of rollovers exceeded
<b>positionWinning</b>	(rollover)	Can't rollover winning positions
<b>optionIsClosed</b>	(rollover)	Can't rollover to or from a closed option
<b>invalidPositionId</b>	positionId	No positions were found by positionId
<b>invalidReceptionEmployeeId</b>	receptionEmployeeId	Must be variable int
<b>invalidMTPositionId</b>	MTPositionId	minLength(1)
<b>updateFailed</b>		Api request failed to run
<b>unavailableProduct</b>	product	If product is optionsBuilder and platform is V2
<b>customerNotAllowedToTrade</b>	regStatus	regStatus Is no trade
<b>invalidCustomer</b>	regStatus	regStatus Is deleted
<b>invalidCustomerRegStatus</b>	regStatus	regStatus Is not activated
<b>invalidCustomerRegulateStatus</b>	regulateStatus, regulateType	check customer's regulation
<b>positionLimited</b>	api_username	check user limitation
<b>BAT-invalidCalculatedAmount</b>	calculatedAmount	Must be variable float
<b>invalidServerURL</b>	serverURL	minLength(2)
<b>invalidCustomerIdOnServer</b>	MTId	Customer does not exist mtId

<b>noPermissionsForThisCustomer</b>	campaignId	Customer do not have permission
<b>noTagPermissionsForThisCustomer</b>	customerId	No permission for taggedCustomers
<b>invalidRuleId</b>	ruleId	For sixtySeconds , Must be variable int
<b>notRegulationApproved</b>	regulateStatus	For non-pending customers and while allow Trading For Pending Customers is allowed
<b>noRuleFound</b>	ruleId	ruleId not found in request
<b>minAmountIs (minAmount)</b>	amountTooLow	SUSPENSION error map
<b>maxAmountIs (maxAmount)</b>	amountTooHigh	SUSPENSION error map
<b>noAssetId</b>	assetId	If no positions for asset id
<b>invalidAssetId</b>	assetId	If no positions for asset id
<b>invalidPositionId</b>	position	Error map

## 2.4 CreditCardUser

Using this module, you could view and edit.

### 2.4.1 Fields

Field	Type	Null	Default	Mandatory (Edit )	Comments
<b>creditCardId</b>	int(10)	No		✓	The Credit Card Id
<b>username</b>	varchar(40)	No		✗	Allcharge username
<b>netpay_cardId</b>	int(10)	No		✗	Netpay card id
<b>cardType</b>	enum('1', '2', '3', '4', '5', '6')	No		✗	
<b>cardNum</b>	varchar(4)	No		✗	4 Last digits
<b>ExpMonth</b>	int(2)	No		✗	2 digits month. E.g. 02
<b>ExpYear</b>	int(4)	No		✗	4 digits year. E.g.2012
<b>FirstName</b>	varchar(20)	No		✗	
<b>LastName</b>	varchar(20)	No		✗	
<b>Address</b>	varchar(40)	No		✗	
<b>City</b>	varchar(30)	No		✗	
<b>State</b>	varchar(30)	No		✗	
<b>CVV2/PIN</b>	varchar(5)	No		✗	
<b>postCode</b>	varchar(15)	No		✗	
<b>Country</b>	varchar(4)	No		✗	
<b>Phone</b>	varchar(25)	No		✗	
<b>currency</b>	varchar(5)	No		✗	
<b>customerId</b>	int(10)	No		✓	
<b>fundId*</b>	tinyint(3)	No		✗	Used for multiple cards for a client. First fund id is 1. Add credit card is -1.
<b>cardBalance</b>	int(7)	No		✗	Total money deposited by this card (excluding withdrawals)
<b>status</b>	enum('active', 'canceled')	No		✗	

## 2.4.2 Examples

### Getting customer's credit cards array

```
MODULE=CreditCardUser&COMMAND=view&FILTER[customerId]=108
```

### Getting customer given credit card details

```
MODULE=CreditCardUser&COMMAND=view&FILTER[customerId]=108&fundId=2
```

### Deleting (Cancelling) a credit card

```
MODULE=CreditCardUser&COMMAND=edit&creditCardId=5&status=canceled
```

## 2.4.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>noFilter</b>	FILTER[]	No filters were found
<b>creditCardIdMustBeInt</b>	creditCardId	Must be variable of type int
<b>invalidStatus</b>	creditCardId	Must be int an exists in creditcard user's status
<b>invalidCustomerId</b>	customerId	Checks customerId is valid
<b>CreditCardNotFound</b>	status	Returns errors if the credit card is not valid or was not found

## 2.5 CustomerDeposits

CustomerDeposit module is used to make new deposits.

New deposits can be made in several payment methods:

1. Wire transfer
2. Credit Card
3. Bonus (for back office)
4. CashU
5. ccTerminal - Just adding a record in the table, not actually charging the user (used to manually add credit card deposits)
6. depositTerminal - Just adding a record in the table, not actually charging the user (used to manually add other deposit types like CashU, Moneybookers, etc.)
7. pendingDepositTerminal – adding a record with a pending status

When selecting credit card, it is possible to either use an existing credit card, by passing it's fundId, or to add a new card by passing fundId=-1.

If you selected to add a new card, you will also need to pass the required fields of the CreditCardUsers module.

### 2.5.1 Fields

Field	Type	Null	Default	Mandatory (add creditCard)	Comments
<b>(method)*</b>	('wire', creditCard', 'bonus', 'cashU', 'ccTerminal', 'depositTerminal', 'pendingDepositTerminal')				Used to determine deposit payment method. The DB field is actually called paymentMethod and has different options.
<b>cardNum</b>	Int			✓	
<b>CVV2/PIN</b>	varchar			✓	
<b>cardType</b>	Int			✓	
<b>ExpMonth</b>	int			✓	example -02
<b>ExpYear</b>	Int			✓	example- 2017
<b>FirstName</b>	varchar(19)			✓	
<b>LastName</b>	varchar(19)			✓	
<b>Address</b>	varchar(45)			✓	
<b>City</b>	varchar(45)			✓	
<b>State</b>	varchar(5)			✗	
<b>postCode</b>	varchar			✓	
<b>Country</b>	int			✓	
<b>Phone</b>	Int(20)			✓	



<b>bankName</b>	varchar(45)	No		✗	(For wire transfers only)
<b>bankNumber</b>	varchar(45)	No		✗	(For wire transfers only)
<b>accountNumber</b>	varchar(45)	No		✗	(For wire transfers only)
<b>branchNumber</b>	varchar(45)	No		✗	(For wire transfers only)
<b>iban</b>	varchar(45)	No		✗	(For wire transfers only)
<b>customerId</b>	int(10)	No		✓	
<b>clearedBy</b>	enum('AllCharge', 'NetPay', 'cashU', 'Moneybookers')	No		✗	Clearing company for credit card deposits- Required fields for ccTerminal
<b>paymentMethod</b>	enum('Wire', 'Credit Card', 'eGold', 'Western Union', 'Bonus', 'PayPal', 'cashU', 'Moneybookers', 'WebMoney', 'MonetaRU', 'Wallet1', 'Alert Pay', 'Qiwil', 'Qiwil', 'Wallets', 'CQR', 'RBK', 'depositTerminal', 'pendingDepositTerminal', 'Neteller, Customized')			✗	Clearing company for credit card deposits- Required fields for depositTerminal, pendingDepositTerminal
<b>clearingUserID</b>	int(5)	No		✗	The ID of the allcharge user used to make the deposit
<b>receptionEmployeeId</b>	int(10)	No		✗	
<b>processEmployeeId</b>	int(10)	No		✗	
<b>amount</b>	float(10,2)	No		✓	
<b>currency</b>	enum('EUR', 'USD')	No		✗	
<b>transactionID</b>	varchar(32)	No		✗	Required for ccTerminal, depositTerminal, pendingDepositTerminal
<b>IPAddress</b>	varchar(16)	No		✗	Can to add and edit on ccTerminal, deposit Terminal, pendingDepositTerminal only
<b>confirmationCode</b>	varchar(32)	No		✗	
<b>requestTime</b>	datetime	No		✗	
<b>confirmTime</b>	timestamp	No	0000-00-00 00:00:00	✗	
<b>type</b>	enum('deposit')	No	deposit	✗	
<b>Status</b>	enum('pending', 'rejected', 'approved', 'canceled', 'fraud')	No		✗	
<b>Leverage</b>	int	No		✗	Optional only in bonus method

<b>customPaymentMethod</b>	varchar(150)	No			
----------------------------	--------------	----	--	--	--

### 2.5.3 Edit Fields

Field	Type	Null	Default	Comments
<b>status</b>	enum('approved', 'canceled')	No		Only available for pending deposits!
<b>receptionEmployeeId</b>	int(10)	No		
<b>processEmployeeId</b>	int(10)	No		
<b>confirmEmployeeId</b>	int(10)	No		
<b>IPAddress</b>	Varchar	No		Only for ccTerminal, depositTerminal, pendingDepositTerminal

### 2.5.3 Examples

#### Making a deposit using an existing card

```
MODULE=CustomerDeposits&COMMAND=add&method=creditCard&customerId=12&fundId=2&amount=200
```

#### Making a deposit using a new credit card

```
MODULE=CustomerDeposits&COMMAND=add&method=creditCard&customerId=12&fundId=-1&cardType=3&cardNum=4580100010011002&ExpMonth=06&ExpYear=2017&CVV2/PIN=1234&FirstName=Ior&LastName=Test&Address=street50&City=Petahtiqua&postCode=4000&Country=120&State=NY&Phone=0333313&currency=USD&amount=300
```

#### Getting a list of a customer's deposits

```
MODULE=CustomerDeposits&COMMAND=view&FILTER[customerId]=13
```

#### Giving a bonus to Customer (using his Id -123)

```
MODULE=CustomerDeposits&COMMAND=add&customerId=123&method=bonus&amount=200
```

#### Adding a Cashu deposit using depositTerminal

```
MODULE=CustomerDeposits&COMMAND=add&method=depositTerminal&customerId=12&amount=300&transactionID=abcdTest123&paymentMethod=cashU
```

#### Adding a Cashu deposit using pendingDepositTerminal

```
MODULE=CustomerDeposits&COMMAND=add&method=pendingDepositTerminal&customerId=12&amount=300&transactionID=abcdTest123&paymentMethod=cashU
```

#### Changing processEmployeeId of a deposit 27954

MODULE=CustomerDeposits&amp;COMMAND=edit&amp;depositId=27954&amp;processEmployeeId=128

## 2.5.4 Status messages

\*Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	(edit command)	Customer was not found
<b>addFailed</b>	(add command)	General error while trying to add the record
<b>requiredFieldsMissing</b>	(add command)	Some of the required fields were not passed
<b>invalidFirstName</b>	FirstName	First name should be at 2-45 characters
<b>invalidLastName</b>	LastName	Last name should be at 2-45 characters
<b>invalidCreditCard</b>	fundId	Credit card number does not exist
<b>creditCardAlreadyExists</b>	cardNum	Credit card already exists for this user
<b>invalidAmount</b>	amount	Field is not numeric
<b>invalidCardNo</b>	cardNum	Field is not numeric
<b>invalidCVV</b>	CVV2/PIN	Field is not numeric
<b>invalidCardType</b>	cardType	Field is not numeric
<b>invalidExpMonth</b>	ExpMonth	Field is not numeric
<b>invalidExpYear</b>	ExpYear	Year should be in the future
<b>invalidCountry</b>	Country	Country should be an integer between 1 and 239. See country list.
<b>invalidCity</b>	City	City field is over than 2 characters
<b>invalidAddress</b>	Address	Address should be less than 45 characters
<b>invalidPhoneNo</b>	Phone	Phone should be less than 20 characters
<b>invalidPostalCode</b>	postCode	Postal code should be less than 10 characters
<b>invalidTransactionId</b>	transactionID	The transaction id is invalid or not specified
<b>transactionExists</b>	transactionID	A credit card deposit with the same transaction id already exists
<b>invalidClearingCompany</b>	clearedBy	The clearedBy field does not contain one of the possible clearing companies
<b>paramsError</b>		Generic error - deposit has failed because the clearing

		company rejected one of the parameters
<b>unableToDeposit</b>		Generic error - deposit was rejected by clearing company
<b>cardDeclined</b>		Generic error - Card was declined by clearing company
<b>invalidIPAddress</b>		Failed to verify ip address
<b>noCustomerId</b>	id	Customer id not found
<b>demoUsersDepositBlocked</b>	isDemo	Block Demo User Deposits is enabled
<b>invalidReceptionEmployeeId</b>	receptionEmployeeId	Not found in employees Options
<b>invalidProcessEmployeeId</b>	processEmployeeId	Not found in employees Options
<b>invalidConfirmEmployeeId</b>	confirmEmployeeId	Not found in employees Options
<b>invalidCustomPaymentMethod</b>	customPaymentMethod	Must be variable string
<b>invalidDescription</b>	description	Must be variable string
<b>invalidPaymentMethod</b>	paymentMethod	Not found in payment Options
<b>leverageShouldBeAboveMinLeverage</b>		Must be within Min Bonus Leverage limits
<b>pleaseEnterValidBankName</b>	bankName	minLength(3) maxLength(45)
<b>pleaseEnterValidBankNumber</b>	bankNumber	minLength(3) maxLength(45)
<b>pleaseEnterValidaccountNumber</b>	accountNumber	minLength(3) maxLength(45)
<b>pleaseEnterValidbranchNumber</b>	branchNumber	minLength(3) maxLength(45)
<b>pleaseEnterValidiban</b>	iban	minLength(3) maxLength(45)
<b>invalidState</b>	State	minLength(1) maxLength(5)
<b>invalidPostCode</b>	postCode	Must be variable string
<b>invalidMethod</b>		Must be one of: ccTerminal, depositTerminal, pendingDepositTerminal, bonus, wire, creditCard
<b>invalidStatus</b>	status	Must be either: Approved, canceled
<b>invalidDepositId</b>	depositId	Deposit id was not found
<b>transactionNotPending</b>		Approved - Not pending
<b>noFieldsToEdit</b>		Api response is empty

## 2.6 Options

### 2.6.1 Fields

Field	Type	Null	Default	Comments
<b>assetId</b>	int(45)	No		
<b>startDate</b>	timestamp	No	CURRENT_TIMESTAMP	
<b>endDate</b>	timestamp	No	0000-00-00 00:00:00	
<b>profit</b>	smallint(6)	No		
<b>loss</b>	smallint(6)	No		
<b>multiplier</b>	smallint(6)	No	100	in %, modifies the value of the asset
<b>noPositionTime</b>	smallint(6)	No		time in minutes in which option ends and no put/call can be done
<b>lastPositionTime</b>	smallint(6)	No		time in minutes, close to position end at which the client can still put/call
<b>lastPositionTime</b>	smallint(6)	No		
<b>endRate</b>	decimal(20,8)	No	0.00000000	
<b>status</b>	enum('open', 'expired', 'canceled')	No	open	
<b>endTradePrice</b>	decimal(20,8)	No		
<b>priceTick</b>	int(2)	No		
<b>color</b>	tinyint(1)	No	0	0 stands for red ; 1 stands for green
<b>ruleId</b>	int(11)	No		
<b>callPositionsNum</b>	smallint(5)	No		
<b>putPositionsNum</b>	smallint(5)	No		
<b>callPositionsAmount</b>	mediumint(6)	No		
<b>putPositionsAmount</b>	mediumint(6)	No		
<b>suspended</b>	tinyint(1)	No	0	if option suspended, no position can be made
<b>pricePer</b>	decimal(4,3)	No		
<b>goalRate</b>	Float	No		
<b>isWeekendOption</b>	tinyint(1)	No	0	

### Default filter

The default filter will return an array of currently trading options (startDate < now, endDate > now), ordered by showInHome, grouped by asset id's.

The asset ids and the option ids will be the suffix of the xml node (e.g: data\_35)

## 2.6.2 Examples

### Getting options by default filter (all current open options)

```
MODULE=Options&COMMAND=view
```

### Getting open options for 10:30: ( min = from, max = to )

```
MODULE=Options&COMMAND=view&FILTER[startDate][max]=2010-07-01  
10:30:00&FILTER[endDate][min]=2010-07-01 10:30:00&FILTER[status]='open'
```

### Getting all Option Builder options

```
MODULE=Options&COMMAND=view&FILTER[product]=optionsBuilder
```

### Getting live Option Builder options

```
MODULE=Options&COMMAND=view&FILTER[product]=optionsBuilder&FILTER[options]=live
```

### Getting all Sixty Seconds options

```
MODULE=Options&COMMAND=view&FILTER[product]=sixtySeconds
```

### Getting live Sixty Seconds options

```
MODULE=Options&COMMAND=view&FILTER[product]=sixtySeconds&FILTER[options]=live
```

### 2.6.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidId</b>	id	Invalid option id
<b>optionIsCurrentlyTrading</b>	optionId	Cannot remove option - option is currently trading
<b>noPermissionSite</b>		isWeatherSite is false
<b>invalidProduct</b>	product	Not found in regular products
<b>invalidAssetId</b>	assetId	Not found in assets
<b>invalidStartDate</b>	startDate	Must be in future
<b>invalidEndDate</b>	endDate	Must be in future
<b>invalidLoss</b>	loss	Must be variable int
<b>invalidProfit</b>	profit	Must be variable int
<b>invalidGoalRate</b>	goalRate	Must be variable int
<b>invalidNoPositionTime</b>	noPositionTime	Must be variable int
<b>invalidNoRolloverTime</b>	noRolloverTime	Must be variable int
<b>invalidLastPositionTime</b>	lastPositionTime	Must be variable int
<b>addOptionFailed-productParameterIsNotSet</b>	product	Product was not sent
<b>addOptionFailed-givenAssetIdsNotLikeInRule</b>	assetId	authentication - check if rule->assetId == assetId
<b>addOptionFailed-customerIsSuspended</b>	customer	Customer is suspended
<b>addOptionFailed-assetRateInvalid</b>		rate <= 0
<b>addOptionFailed-theAssetsIsNotTradeable</b>	assetId	assetId is not in tradable assets
<b>addOptionFailed-theAssetsIsSuspended</b>	assetId	assetId is suspended by rule
<b>addOptionFailed-invalidAmount-shouldBeBetween (finalMinInvestment) and (finalMaxInvestment)</b>	amount	Must be between min and max investment
<b>addOptionFailed-expireTimeIsInThePast</b>	expireTime	check if the given expire time is in the future
<b>addOptionFailed-expireTimeShouldBeBetween (minOptionTime) and (maxOptionTime)</b>	expireTime	Must be between min and max option time
<b>addOptionFailed-theAssetsIsNotTradeable</b>		check if asset is tradeable
<b>addOptionFailed-customerSuspended</b>	additionalParams, originalRate	Customer is suspended
<b>addOptionFailed-invalidPositionRate</b>	additionalParams, originalRate	check if the option rate is set
<b>addOptionFailed-insufficientFunds</b>	sourcePlatform, amount	check if the customer has sufficient funds
<b>addOptionFailed-riskSuspensionActivated</b>	AppRiskLevelSuspensions, assetId	check risk suspension
<b>invalidVIPGroup</b>	VIPGroup	Not in VIP Group Options

## 2.7 Withdrawals

### 2.7.1 Fields

Field	Type	Null	Default	Mandatory (add)	Editable	Comments
method	wire, creditCard ccTerminal, ccPendingTerminal, chargeback,bonus, withdrawalTerminal,fraud		wire		✗	Withdrawal method
(fundId)	int			If creditCard method ✓ if not ✗	✗	If credit card withdrawal, the fund id of the card
confirmation Code	varchar(32)	No		✗	✗	
amount	float(10,2)	No		✓	✓	
requestTime	timestamp	No	CURRENT_TIMESTAMP	✗	✗	Withdrawal request time
confirmTime	timestamp	No	0000-00-00 00:00:00	✗	✗	Withdrawal approve time
status	enum('pending', 'rejected', 'approved', 'canceled')	No		✗	✓	
iban	varchar(40)	No		If wire method ✓ if not ✗	✓	
branchNumber	varchar(45)	No		If wire method ✓ if not ✗	✓	
bankName	varchar(45)	No		If wire method ✓ if not ✗	✓	
bankNumber	varchar(45)	No		If wire method ✓ if not ✗	✓	
accountNumber	varchar(45)	No		If wire method ✓ if not ✗	✓	
currency	enum('EUR', 'USD')	No		✗		
cancelReason	text	No		✗	✓	
receptionEmployeeId	int(11)	No		✗	✓	Employee who received the request. empty if the request was made by the site



<b>processEmployeeId</b>	int(11)	No		x	✓	Employee who processed the request
<b>customerId</b>	int(11)	No		✓	x	Customer who made the request
<b>type</b>	enum('withdrawal')	No	withdrawal		x	
<b>clearingUserID</b>	int(11)	No		x	x	If credit card withdrawal, the credit card user id
<b>transactionID</b>	varchar(32)	No		If ccTerminal or ccPendingTerminal or withdrawalTerminal method ✓ if not x	x	Required for ccTerminal
<b>swiftCode</b>	varchar(12)			If wire method ✓ if not x	✓	Swift Code to be entered by approving employee
<b>paymentMethod</b>	Enum('Wire', 'CreditCard', 'eGold', 'Western Union', 'Bonus', 'PayPal', 'cashU', 'Moneybookers', 'WebMoney', 'Wallet1', 'walpay', 'AlertPay', 'MonetaRU', 'libertyReserve', 'AllCharge', 'UnionPay', 'credorax', 'OKPay', 'chargeBack', 'Neteller', 'SafeCharge', 'moneyTransfer', 'other', 'Qiji', 'Customized')			If withdrawalTerminal method ✓ if not x	✓	
<b>clearedBy</b>	varchar(64)			x	x	Optional to add in 'withdrawal Terminal' method
<b>externalId</b>	varchar(255)			x	✓	
<b>customPaymentMethod</b>	varchar(150)	No		x		

**Automatically set by the system and cannot be edited**

## 2.7.2 Examples

### Getting customer credit card withdrawals:

```
MODULE=Withdrawal&COMMAND=view&FILTER[customerId]=120&FILTER[method]=creditCard
```

### Adding new credit card withdrawal:

```
MODULE=Withdrawal&COMMAND=add&customerId=120&fundId=2&amount=100&method=creditCard
```

### Adding new Charge Back / Fraud withdrawal:

```
MODULE=Withdrawal&COMMAND=add&customerId=120&amount=100&clearedBy=allCharge&method=chargeBack / fraud
```

Notice: In clearedBy field you can send the following values - allCharge, netPay

## 2.7.3 Status messages

\*Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	(edit command)	Customer was not found
<b>addFailed</b>	(add command)	General error while trying to add the record
<b>requiredFieldsMissing</b>	(add command)	Some of the required fields were not passed
<b>noCustomerId</b>	customerId	Customer does not exist in customers table
<b>invalidAmount</b>	amount	Amount should be numeric
<b>withdrawalAmountHigherThanBalance</b>	amount	Selected amount to withdraw is higher than the user balance
<b>invalidCreditCard</b>	fundId	Field is not numeric
<b>invalidIban</b>	iban	iban should be 5 to 45 characters
<b>invalidBranchNumber</b>	branchNumber	Field is not numeric
<b>invalidBankName</b>	bankName	Bank name should be 3 to 45 characters
<b>invalidBankNumber</b>	bankNumber	Field is not numeric
<b>invalidAccountNumber</b>	accountNumber	Account number should be 5 to 45 characters
<b>creditCardNotFound</b>	fundId	Selected fund was not found
<b>invalidWithdrawalId</b>	id	'id' is not valid or withdrawal does not exist
<b>cannotEditApprovedWithdrawal</b>	id	The selected withdrawal cannot be edited because it has already been approved

<b>invalidConfirmationCode</b>	confirmationCode	Code should be 2 to 32 characters
<b>invalidReceptionEmployee</b>	receptionEmployeeId	Employee does not exist
<b>invalidProcessEmployee</b>	processEmployeeId	Employee does not exist
<b>invalidStatus</b>	invalidStatus	Status value is not one of the possible status values
<b>cancelReason</b>	cancelReason	Field is not a string
<b>withdrawalAmountCannotBeNegative</b>	amount	Withdrawal amount cannot be negative
<b>invalidExternalId</b>	externalId	Check not external id
<b>invalidPaymentMethod</b>	paymentMethod	Check exists in paymentMethodOptions
<b>invalidSwiftCode</b>	swiftCode	minLength(8)->maxLength(11)
<b>externalIdTooShort</b>	externalId	Smaller than min length
<b>invalidDoughtFlowUse</b>	doughtflow	Checks equals 1
<b>cannotEditApprovedWithdrawal</b>		Checks status is approved and doughtflow was not sent
<b>cannotEditRejectedWithdrawal</b>		Withdrawal rejected
<b>invalidFundId</b>	fundId	
<b>invalidTransactionId</b>	customerId	Checks user exists and gets their fund id
<b>invalidClearedBy</b>	clearedBy	maxLength(25)
<b>invalidCardnumber</b>	Cardnumber	minLength(9)->maxLength(9)
<b>invalidIBAN</b>	iban	minLength(5)->maxLength(45)

## 2.8 Assets

### 2.8.1 Fields

Field	Type	Null	Default	Comments
<b>name</b>	varchar(45)	No		
<b>isTradeable</b>	tinyint(1)	No		-1 - not tradable in gcp, 0 - not tradable in whitelabel, 1 - tradable in white label
<b>symbol</b>	varchar(20)	No		This is the symbol of the asset which'll be queried from Reuters servers
<b>rate</b>	decimal(20,8)	No	0.00000000	Latest asset value that was synced from GCP
<b>tailDigits</b>	smallint(6)	No		Number of tail digits
<b>lastUpdated</b>	datetime	No		Last sync from GCP time
<b>type</b>	enum('commodities', 'currencies', 'indices', 'stocks')	No	stocks	Asset type
<b>showInHome</b>	tinyint(1)	No		If this is true, this asset will be preferable to be shown in home page
<b>priceTick</b>	int(11)	No		Price tick field from Reuters
<b>color</b>	tinyint(1)	No		0 stands for red ; 1 stands for green
<b>correlation</b>	varchar(5)	No		
<b>trend</b>	enum('call', 'put', 'noTrend')	No		
<b>trendType</b>	tinyint(2)	No		

### 2.8.2 Examples

#### Getting all the trade-able currencies

```
MODULE=Assets&COMMAND=view&FILTER[isTradeable]=1&FILTER[type]=currencies
```

### 2.8.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>noPermissionSite</b>		config.Api.isWeatherSite is disabled
<b>invalidAssetId</b>	assetId	The asset id
<b>invalidSuspendType</b>	suspendType	regularSuspended, builderSuspended, sixtySuspended, forexSuspended, ldoSuspended, dotSuspended
<b>invalidSuspend</b>	suspend	Receives values 0 or 1

## 2.9 AssetsHistory

### 2.9.1 Fields

Field	Type	Null	Default	Comments
<b>date</b>	timestamp	No	CURRENT_TIMESTAMP	
<b>assetId</b>	int(11)	No		
<b>rate</b>	decimal(20,8)	No		
<b>tradePrice</b>	decimal(20,8)	No		TRDPRC_1 field from Reuters
<b>priceTick</b>	int(11)	No		PRCTCK_1 field from Reuters

### 2.9.2 Examples

**Getting rates of asset id 2 that were recorded after 8 PM at October 24th**

```
MODULE=AssetsHistory&COMMAND=view&FILTER[assetId]=2&FILTER[date][min]=2010-10-24 20:00:00
```

If no date filter is passed at all, it will fetch records that have been recorded within the last hour

**Getting rates of asset id 2 that have been recorded within the last hour**

```
MODULE=AssetsHistory&COMMAND=view&FILTER[assetId]=2
```

### 2.9.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>mustUseFilter</b>		

## 2.10 Country

Field	Type	Null	Default	Comments
<b>id</b>	int(10)			Country ID
<b>iso</b>	char(2)	No		ISO 2 letters country code
<b>name</b>	varchar(80)	No		Country name
<b>iso3</b>	char(3)	Yes	<i>NULL</i>	ISO 3 letters country code
<b>numcode</b>	smallint(6)	Yes	<i>NULL</i>	
<b>block</b>	tinyint(1)	No		Block country?
<b>allowRegistration</b>	tinyint(1)	No		Allow registration from this country?
<b>prefix</b>	char(5)	Yes	<i>NULL</i>	The country phone prefix
<b>city*</b>				The city that correspond to the user's IP
<b>region*</b>				The region that correspond to the user's IP

- Currently, this module is only used for "view" command.
- 'allowRegistration' and 'block' are only enabled for some of the white labels.
- Country can also be searched by non-standard field - 'ip'. If you send a request with FILTER[ip], it will attempt to match that ip to a country.
- \* "city" and "region" fields are only returned for a given IP, if found. They are not real fields, so they can't be queried.

### 2.10.1 Examples

#### Getting a country by IP:

```
MODULE=Country&COMMAND=view&FILTER[ip]=74.125.43.99
```

#### Getting a country by ISO code:

```
MODULE=Country&COMMAND=view&FILTER[iso3]=ISR
```

#### Getting all the blocked countries

```
MODULE=Country&COMMAND=view&FILTER[block]=1
```

### 2.10.2 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidCountry</b>	country	From all countries in DB, country is not valid
<b>invalidAllowRegistration</b>	allowRegistration	Receives 0 or 1

## 2.11 Campaign

### 2.11.1 Fields

Field	Type	Default	Mandatory (add Campaign)	Editable	Comments
name	varchar(150)	No	✓	✓	
startDate	date	No	✗	✓	
endDate	date	No	✗	✓	
type	enum('CPL', 'CPA', 'FIX')	No	✓	✓	
visitorCode	text	No	✗	✓	
leadCode	text	No	✗	✓	
customerCode	text	No	✗	✓	
playerCode	text	No	✗	✓	
assetImageEnable	int	No	✓	✓	Select Asset Display
country	smallint(5)	No	✓	✓	id in country table
cookieTimeout	int(10)	No	✓	✓	
budget	int(10)	No	✓	✓	
leadsNum	int(10)	No	✗		
customersNum	mediumint(8)	No	✗		
visitsNum	int(10)	No	✗		
uniqueVisits	int(10)	No	✗		
playersNum	smallint(7)	No	✗		
totalDeposits	int(10)	No	✗		

### 2.11.2 Examples

#### Getting details about Google campaign

```
MODULE=Campaign&COMMAND=view&FILTER[name]=Google
```

#### Add Campaign

```
MODULE=Campaign&COMMAND=add&name=mytest&type=CPL&assetImageEnable=1&country=104&budget=1000&cookieTimeout=45
```

#### Edit Campaign

```
MODULE=Campaign&COMMAND=edit&id=3&name=mytest&type=CPL&assetImageEnable=1&country=104&budget=1000
```

### 2.11.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>updateFailed</b>	(edit command)	Update Failed
<b>pleaseEnterCampaignName</b>	name	Should be string
<b>campaignNameAlreadyExists</b>	name	Campaign name must be unique
<b>pleaseEnterValidStartDateDay</b>	startDate	Wrong date/time format or dates are colliding
<b>pleaseEnterValidEndDateDay</b>	endDate	Wrong date/time format or dates are colliding
<b>pleaseSelectValidCampaignType</b>	type	Need to correspond to campaign types. Example 'CPL','CPA','FIX'
<b>pleaseSelectValidCountry</b>	country	This input must be of variable type int
<b>pleaseSelectAssetDisplay</b>	assetImageEnable	This input must be of variable type int
<b>pleaseEnterBudget</b>	budget	This input must be of variable type string
<b>pleaseEnterValidVisitorCode</b>	visitorCode	This input must be of variable type string
<b>pleaseEnterValidLeadCode</b>	leadCode	This input must be of variable type string
<b>pleaseEnterValidCustomerCode</b>	customerCode	This input must be of variable type string
<b>pleaseEnterValidPlayerCode</b>	playerCode	This input must be of variable type string
<b>pleaseEnterCookieTimeout</b>	cookieTimeout	This input must be of variable type string
<b>invalidCampaignId</b>	id	This input must be of variable type int
<b>invalidCampaignIdNotExists</b>	id	This campaign doesnt exist
<b>noFieldsToEdit</b>		API failed to receive data



## 2.12 Sub Campaigns

### 2.12.1 Fields

Field	Type	Null	Default	Mandatory(add)	Comments
<b>param</b>	varchar(500)	No		✓	unique campaign id given by the affiliate
<b>leadsNum</b>	mediumint(8)	No		✗	
<b>customersNum</b>	mediumint(7)	No		✗	
<b>visitsNum</b>	mediumint(8)	No		✗	
<b>uniqueVisits</b>	mediumint(8)	No		✗	
<b>playersNum</b>	smallint(7)	No		✗	
<b>totalDeposits</b>	int(8)	No		✗	
<b>mainCampaign</b>	smallint(4)	No		✓	

### 2.12.2 Examples

#### Getting details about all sub-campaigns of campaign 4

```
MODULE=SubCampaign&COMMAND=view&FILTER[mainCampaign]=4
```

#### Adding a sub-campaign to Campaign 4

```
MODULE=SubCampaign&COMMAND=add&param=spotOptionTest&mainCampaign=4
```

### 2.12.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidParam</b>	(add command)	param supplied is too short (shorter than 1 char)
<b>paramAlreadyExists</b>	(add command)	param supplied already exists
<b>invalidMainCampaign</b>	(add command)	mainCampaign supplied is not numeric
<b>noSuchMainCampaign</b>	(add command)	mainCampaign supplied doesn't exist

## 2.13 Calls

### 2.13.1 Fields

Field	Type	Null	Default	Mandatory(add)	Comments
id	int(11)	No		✗	
subject	varchar(255)	No		✓	
content	text	No		✓	
date	timestamp	No	CURRENT_TIMESTAMP	✗	
employeeId	int(11)	No		✓	
clientId	int(11)	No		✓	Customer id
custom	varchar			✗	

### 2.13.2 Examples

#### Getting details all the calls related to customer 4

```
MODULE=Call&COMMAND=view&FILTER[clientId]=4
```

#### Adding a call to customer 4

```
MODULE= Call&COMMAND=add&clientId=4&employeeId=10&content=the client is happy&subject=bonus call
```

### 2.13.3 Status messages

Message	Field	Description
noResults	(view command)	No results matching to the filter criteria were found.
invalidEmployeeId	employeeId	A list of employees
contentsEmpty	content	minLength(1)
subjectsEmpty	subject	minLength(1)
invalidCustomerId	clientId	Checking allowed and found customers
addFailed		API result
invalidCustom	custom	Custom field must be string

## 2.14 Rules

Only view is currently available for this module

The rules can be displayed by product (optionsBuilder/sixtySeconds).

### 2.14.1 Examples

**Getting all rules for Options Builder**

```
MODULE=Rules&COMMAND=view&FILTER[product]=optionsBuilder
```

**Getting live rules for Options Builder**

```
MODULE=Rules&COMMAND=view&FILTER[product]=optionsBuilder&filter[rules]=live
```

**Getting live rules for Sixty Seconds**

```
MODULE=Rules&COMMAND=view&FILTER[product]=sixtySeconds&FILTER[rules]=live
```

### 2.14.2 Status messages

Message	Field	Description
noResults	(view command)	No results matching to the filter criteria were found.
mustUseFilter		

## 2.15 DepositsLog

Only view is currently available for this module

### 2.15.1 Fields

Field	Type	Null	Default	Mandatory(add)	Comments
firstName	varchar(45)	No		✗	
lastName	varchar(45)	No		✗	
cardNum	varchar(4)	No		✗	
amount	float(10,2)	No		✓	
currency	enum('USD', 'EUR')	No		✗	
customerId	int(11)	No		✓	
errorText	text	No		✓	
rawError	text	No		✓	
date	timestamp	No		✗	

**NOTE:** The response will return a maximum of 50 results on each request

### 2.15.2 Examples

#### Getting The Deposit Log of Customer 47

```
MODULE=DepositsLog&COMMAND=view&FILTER[customerId]=47
```

#### Getting The Deposit Log where the amount is over 100

```
MODULE=DepositsLog&COMMAND=view&FILTER[amount][min]=100.00
```

#### Getting The Deposit Log between Dates and times

```
MODULE=DepositsLog&COMMAND=view&FILTER[date][min]=2010-08-08 06:00&FILTER[date][max]=2010-08-10 07:00
```

#### Add new row

```
MODULE=DepositsLog&COMMAND=add&cardNum=xxxx&amount=xxx&customerId=xxxx&errorText=Unable to deposit using xxxxx&rawError=Card not supported&date=2014-10-29 15:16:33
```

### 2.15.3 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	customerId	Customer not allowed
<b>invalidIP</b>		Ip is not valid
<b>pleaseEnterValidCardNum</b>	cardNum	Mast be variable int
<b>pleaseEnterAmount</b>	amount	Mast be variable float
<b>invalidCustomerId</b>	customerId	Mast be variable int
<b>pleaseEnterErrorText</b>	errorText	Mast be variable string
<b>pleaseEnterRawError</b>	rawError	Mast be variable string
<b>pleaseEnterDate</b>	date	Must be date format

## 2.16 CampaignStats

### 2.16.1 View

#### Request Fields

Field	Type	Null	Default	Comments
startDate	varchar(45)	No		
endDate	varchar(45)	No		
campaignId	int(11)	No		

NOTE: all three fields are mandatory if you wish to get a result set.

#### Response Fields

Field	Comments
visits	visit count
uniqueVisits	unique visit count
customersNum	Customer count
leadNum	Lead count
playersNum	First depositor count
totalDeposits	Total deposit amount

### 2.16.2 Edit

#### Request Fields

Field	Type	Null	Default	Comments
campaignId	int(11)	No		
visitCount	enum('visitor','uniqueVisitor')	No		

NOTE: the visitCount field states which visit do you want to count, a unique visit or a non-unique one.

### 2.16.3 Examples

#### Getting the Campaign Stats for Campaign 2 between the 21/09/11 and 01/10/11

```
MODULE=CampaignStats&COMMAND=view&startDate=2011-09-21&endDate=2011-10-01&campaignId=2
```

#### Counting a unique visitor of campaign 2

```
MODULE=CampaignStats&COMMAND=edit&visitCount=uniqueVisitor&campaignId=2
```

#### Counting a non-unique visitor of campaign 2

```
MODULE=CampaignStats&COMMAND=edit&visitCount=visitor&campaignId=2
```

## 2.16.4 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidStartDate</b>	(view command)	Field is not a string.
<b>invalidEndDate</b>	(view command)	Field is not a string.
<b>invalidMainCampaign</b>	(view\edit command)	Field is not numeric
<b>noSuchCampaign</b>	(view\edit command)	No campaign was found with the supplied id
<b>invalidVisitorCount</b>	(edit command)	visitorCount can have only the ENUM values

## 2.17 SubCampaignStats

### 2.17.1 View

#### Request Fields

Field	Type	Null	Default	Comments
startDate	varchar(45)	No		
endDate	varchar(45)	No		
campaignId	int(11)	No		
subCampaignId	int(11)	No		

NOTE: all four fields are mandatory if you wish to get a result set.

#### Response Fields

Field	Comments
visits	visit count
uniqueVisits	unique visit count
customersNum	Customer count
leadNum	Lead count
playersNum	First depositor count
totalDeposits	Total deposit amount

NOTE: visits & uniqueVisits visits are not campaign-dependent (it will give you all the visits of this subCampaignId, regardless from which campaign they were)

### 2.17.2 Edit

#### Request Fields

Field	Type	Null	Default	Comments
campaignId	int(11)	No		
param	varchar(45)	No		
visitCount	enum('visitor','uniqueVisitor')	No		

1. The visitCount field states which visit do you want to count, a unique visit or a non-unique one.
2. If a subCampaign does not exist with the supplied Param and campaignId, it will be created.

### 2.17.3 Examples

**Getting the Campaign Stats for Campaign 2 and SubCampaign 153 between the 21/09/11 and 01/10/11**

```
MODULE=SubCampaignStats&COMMAND=view&startDate=2011-09-21&endDate=2011-10-01&campaignId=2&subCampaignId=153
```



### Counting a unique visitor of campaign 2 with a param of 'edgetrackerid\_12345'

```
MODULE=SubCampaignStats&COMMAND=edit&visitCount=uniqueVisitor&campaignId=2&param=edgetrackerid_12345
```

### Counting a non-unique visitor of campaign 2 with a param of 'edgetrackerid\_12345'

```
MODULE=SubCampaignStats&COMMAND=edit&visitCount=visitor&campaignId=2&param=edgetrackerid_12345
```

## 2.17.4 Status messages

Message	Field	Description
<b>noResults</b>	(view command)	No results matching to the filter criteria were found.
<b>invalidStartDate</b>	(view command)	Field is not a string.
<b>invalidEndDate</b>	(view command)	Field is not a string.
<b>invalidMainCampaign</b>	(view\edit command)	Field is not numeric
<b>invalidSubCampaign</b>	(view command)	Field is not numeric
<b>noSuchCampaign</b>	(view command)	No campaign was found with the supplied id
<b>noSuchSubCampaign</b>	(view\edit command)	No sub-campaign was found with the supplied id
<b>invalidParam</b>	(edit command)	Field is not a string
<b>invalidVisitorCount</b>	(edit command)	visitorCount can have only the ENUM values
<b>noSuchSubCampaignIdForThisCampaign</b>	campaignId, subCampaignId	Campaign not found
<b>noFilters</b>	FILTER	No FILTER defined

## 2.18 ExternalCampaign

External campaign is campaign that associated with some user.

The external campaign statistics can be accessed via MAP.

The 'add and edit' operations can be created with the API.

### 2.18.1 Add

Field	Type	Null	Default	Mandatory	Comments
userName	varchar(45)			✓	
password	varchar(45)			✓	
campaignId	int(11)			✓	
commissionFormula	varchar(255)			✗	optional
turnover	tinyInt(1)			✗	optional
PNL	tinyInt(1)			✗	optional
givenBonus	tinyInt(1)			✗	optional
deposits	tinyInt(1)			✗	optional
withdrawals	tinyInt(1)			✗	optional
numOfDepositors	tinyInt(1)			✗	optional
numOfCustomers	tinyInt(1)			✗	optional
displayCustomerName	tinyInt(1)			✗	optional
displayCustomerId	tinyInt(1)			✗	optional
displayParams	tinyInt(1)			✗	optional
FTDBaseline	tinyInt(1)			✗	optional

### 2.18.2 Edit

Field	Type	Null	Default	Mandatory	Comments
userName	varchar(45)			✗	optional
password	varchar(45)			✗	optional
campaignId	int(11)			✓	
commissionFormula	varchar(255)			✗	optional
turnover	tinyint(1)			✗	optional
PNL	tinyint(1)			✗	optional
givenBonus	tinyint(1)			✗	optional
deposits	tinyint(1)			✗	optional

<b>withdrawals</b>	tinyint(1)			<b>x</b>	optional
<b>numOfDepositors</b>	tinyint(1)			<b>x</b>	optional
<b>numOfCustomers</b>	tinyint(1)			<b>x</b>	optional
<b>displayCustomerName</b>	tinyint(1)			<b>x</b>	optional
<b>displayCustomerId</b>	tinyint(1)			<b>x</b>	optional
<b>displayParams</b>	tinyint(1)			<b>x</b>	optional
<b>FTDBaseline</b>	tinyint(1)			<b>x</b>	optional

\*\* tinyint(1) - These fields should be set to 1 or 0. If field value is 1 - this feature will be enabled, 0 - this feature will be disabled.

### **2.18.3 Parameters**

commissionFormula - The commission formula for the campaign. The commission is calculated according to this formula.

The commission formula should be constructed with the following parameters: [Deposits], [Withdrawals], [PNL], [FTD], [Turnover], and [GBonus]

turnover - Display the turnover for customers.

PNL - Display the PNL for customers.

givenBonus - Display the total given bonus.

deposits - Displays the total deposits.

withdrawals - Displays the total withdrawals.

numOfDepositors - Displays number of depositors.

numOfCustomers - Displays number of customers.

displayCustomerName - Displays customer name in the first depositors table.

displayCustomerId - Displays customer ID in the first depositors table.

displayParams - Displays params in the first depositors table

FTDBaseline - This value is the minimum amount of money that should be invested by the customer to be counted as first depositor.

## 2.18.4 Examples

### Add new external campaign

```
MODULE=ExternalCampaign&COMMAND=add&userName=test&campaignId=1&password=123456&FTDBa
seline=[FTD]*0.5
```

### Edit external campaign

```
MODULE=ExternalCampaign&COMMAND=edit&campaignId=1&PNL=1&givenBonus=1
```

## 2.18.5 Status messages

Message	Field	Description
username-min4chars	userName	Username should contain at least 4 chars
username-alreadyExists	userName	Username already exists
password-min6chars	password	Password should contain at least 6 chars
Commission formula - unknown parameter:xxxx	commissionFormula	xxxx is the unknown parameter
Parentheses number does not match	commissionFormula	Check the parentheses in the commission formula
userForThisCampaignAlreadyExists	campaignId	There is already a user defined for given campaign id
requiredFieldsMissing		
userForThisCampaignAlreadyExists		

## 2.19 ClickedAssets

Using this module you can track the clicks of the customers, Only Add is available for this module.

### 2.19.1 Add

#### Request Fields

Field	Type	Null	Default	Mandatory (add)	Comments
<b>customerId</b>	int(11)	No		✓	
<b>direction</b>	ENUM('call','put')	No		✓	
<b>assetId</b>	int(11)	No		✓	
<b>optionId</b>	int(11)	No		✓	
<b>product</b>	ENUM('regular','optionsBuilder','sixtySeconds')	No		✓	

- All these fields are mandatory if you wish to get a result set.
- When tracking clicks of options builder/ sixty seconds options , you should send an option Id of the asset id \* -1

### 2.19.2 Examples

#### Track a click on Regular Binary Option, for customer 1, Option 53845, Asset 91, Call Direction

```
MODULE=ClickedAssets&COMMAND=add&customerId=1&direction=call&assetId=91&optionId=53845&product=regular
```

#### Track a click on Options-Builder Option, for customer 1, Asset 91, Put Direction

```
MODULE=ClickedAssets&COMMAND=add&customerId=1&direction=put&assetId=91&optionId=-91&product=optionsBuilder
```

#### Track a click on Sixty Seconds Option, for customer 1, Asset 91, Call Direction

```
MODULE=ClickedAssets&COMMAND=add&customerId=1&direction=call&assetId=91&optionId=-91&product=sixtySeconds
```

### 2.19.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>customerIdMustBeInteger</b>	customerId	This variable must be int
<b>invalidDirection</b>	direction	Direction (call, put)
<b>assetIdMustBeInteger</b>	assetId	This variable must be int
<b>invalidProduct</b>	product	'regular', 'optionsBuilder', 'sixtySeconds'
<b>optionIdMustBeNumeric</b>	optionId	This variable must be int or array based on [optionId]
<b>invalidCustomerId</b>	customerId	Customer not found
<b>invalidOptionId</b>	optionId	Option not found
<b>invalidAssetId</b>	assetId	Asset not found

## 2.20 CustomerData

Using this module you can add info of the customer, to the system, and you will have additional information, assigned to a specific customer

### 2.20.1 View

#### Request Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		

### 2.20.2 Add

#### Request Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		
keys	array	No		

- these fields are mandatory
- The keys array should be like this keys[YOUR-FIELD-NAME]=YOUR-VALUE, for example keys[favoriteAnimal]=mongoose

### 2.20.3 Examples

#### View customer data by id

```
MODULE=CustomerData&COMMAND=view&customerId=16103
```

#### Add one key of Info to the customer on the system

```
MODULE=CustomerData&COMMAND=add&customerId=16103&keys[favoriteAnimal]=mongoose
```

#### Add More than one key of Info to the customer on the system

```
MODULE=CustomerData&COMMAND=add&customerId=16103&keys[additionalAddress]=Kings Rd&keys[favoriteAnimal]=mongoose
```

### **2.20.4 Status messages**

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	customerId	No Customer found for the supplied Id



## 2.21 Regulation

You can use this module, ONLY if you have the regulation enabled for your platform, using this module you can view some of the regulation information, and add customer's regulation answers

### 2.21.1 View

#### Request Fields

Field	Type	Null	Default	Comments
<b>action</b>	ENUM('getEmploymentStatuses', 'getBusinessNature', 'getFundSource', 'getAllActiveQuestions', 'getAllActiveAnswers')	No		

### 2.21.2 Add

#### Request Fields

Field	Type	Null	Default	Comments
<b>customerId</b>	int(11)	No		
<b>formId</b>	int(11)	No		
<b>action</b>	ENUM('insertDynamicAnswer')	No		
<b>keys</b>	array	No		The key is string, and the value is an int, for example: keys[yearsAsTrader]=3

### 2.21.3 Examples

#### Getting Fund Sources

```
MODULE=Regulation&COMMAND=view&action=getFundSource
```

#### Adding one dynamic answer to the customer on the system

```
MODULE=Regulation&COMMAND=add&action=insertDynamicAnswer&formId=1&customerId=16103&keys[yearsAsTrader]=3
```

#### Adding more than one dynamic answer to the customer on the system

```
MODULE=Regulation&COMMAND=add&action=insertDynamicAnswer&formId=1&customerId=16103&keys[yearsAsTrader]=3&keys[netWorth]=19
```

### **2.21.4 Status messages**

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>invalidAction</b>	action	Answer from surveyManager
<b>invalidCustomerId</b>	customerId	Customer id not found
<b>invalidActions</b>	action	Answer from surveyManager

## 2.22 OneTouch

Using This Module You Could View & add (create) One Touch Positions

### 2.22.1 View

#### Request Fields

Field	Type	Null	Default	Comments
id	int(11)	No		The One Touch Position's Id
amount	float(8,2)	No		
currency	VARCHAR(3)	No		
date	timestamp	No		
rate	decimal(20,8)	No		The rate on which the position was placed
customerId	int(11)	No		
optionId	int(11)	No		
numberUnits	smallint(6)	No		Number of units bought
status	enum('open', 'won', 'lost', 'tie', 'pending')	No		
payout	float(10,2)	No		

### 2.22.2 Add

#### Request Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		
optionId	int(11)	No		
numberUnits	smallint(6)	No		

NOTE: All These Fields Are Mandatory For Add

### 2.22.3 Examples

Use LFILTER to filter one touch positions by their info from onetouch\_options.

Getting all onetouch positions filtered by customerId and assetId

```
MODULE=OneTouch&COMMAND=view&FILTER[customerId]=14&LFILTER[assetId]=66
```

Getting the One Touch Positions for Customer 10489

```
MODULE=OneTouch&COMMAND=view&FILTER[customerId]=10489
```

Buying 5 units of option 6360 for Customer 10489

```
MODULE=OneTouch&COMMAND=add&customerId=10489&optionId=6360&numberUnits=5
```

### Getting The One Touch Open (All current open options)

```
MODULE=OneTouch&COMMAND=viewOneTouchOptions
```

## 2.22.4 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>invalidCustomerId</b>	customerId	No Customer found for the supplied Id
<b>invalidOption</b>	optionId	No Option found for the supplied Id
<b>invalidUnitsNumber</b>	numberUnits	Number of units must be an int
<b>notRegulationApproved</b>		Allow Trading For Pending Customers not available
<b>customerNotAllowedToTrade</b>	regStatus	'noTrade','pending','deleted'
<b>noFilter</b>	FILTER	No FILTER in request
<b>invalidPosition</b>	positionId	Must be variable int
<b>invalidPositionId</b>	position	Must be position
<b>requiredFieldsMissing</b>		Some fields are missing
<b>invalidOneTouchPositionId</b>	FILTER[id]	Position was not found
<b>invalidReceptionEmployee</b>	receptionEmployeeId	Not found in users

## 2.23 BMOLog

Using This Module You Could View Buy Me Out logs

### 2.23.1 View

#### Request Fields

Field	Type	Null	Default	Comments
positionId	int(11)	No		The Position Id - <b>required</b>

### 2.23.2 Examples

Getting the but Me Out logs for position 10489

```
MODULE=BMOLog&COMMAND=view&FILTER[positionId]=10489
```

### 2.23.3 Status messages

Message	Field	Description
noResults	(view)	No results matching to the filter criteria were found.

## 2.24 User

Using This Module You Could Add a CRM user and Validate CRM or CMS username and password

### 2.24.1 Fields

Field	Type	Default	Mandatory (add user)	Editable	Comments
email	varchar(255)		✓	✗	
firstName	varchar(25)		✓	✗	
lastName	varchar(25)		✓	✗	
cellphone	varchar(30)		✓	✗	
username*	varchar(25)		✓	✗	
password*	varchar(255)		✓	✓	
currentPassword	varchar(255)		✗	✓	Only for edit must be at least 6 characters
department	ENUM('manager', 'backOffice', 'Sales', 'retention', 'risk', 'marketing', 'assets', 'general')		✓	✗	
position	ENUM('manager', 'personal', 'helpDesk', 'admin')		✓	✗	
sections	varchar(255)		✓	✗	The section should be <u>one or more</u> from this list with ',' as separator: customers,leads,depositors,management,campaigns,accounting,backoffice,exposure,affiliates,customersData
commision	float(4,2)	0	✗	✗	
enableReports	tinyint(1)	0	✗	✗	
displayEmail	tinyint(1)	0	✗	✗	
displayPhone	tinyint(1)	0	✗	✗	
displayDesks	tinyint(1)	0	✗	✗	
userSelectedLang	tinyint(1)	0	✗	✗	
regulationCompliance	tinyint(1)	0	✗	✗	

\* Required for "validate"

## 2.24.2 Examples

### View all CRM Users

```
MODULE=User&COMMAND=view
```

### View CRM Users by conditions

```
MODULE=User&COMMAND=view&FILTER[username]=mike
```

### Add user to the CRM with permissions to customers, leads and back office

```
MODULE=User&COMMAND=add&email=test@test.com&firstName=testName&lastName=testName&
cellphone=12345678901&username=testUser&password=testPassword&department=Sales&position=personal&
sections=customers,leads,backoffice
```

### Add user to the CRM with permissions to customers and reports

```
MODULE=User&COMMAND=add&email=test@test.com&firstName=testName&lastName=testName&
cellphone=12345678901&username=testUser&password=testPassword&department=Sales&position=personal&
sections=customers&enableReports=1
```

### Check if the username and password are validated

```
MODULE=User&COMMAND=validate&username=testUser&password=testPassword
```

### Change password for CRM/CMS

```
MODULE=User&COMMAND=edit&email=test@spotoption.com&currentPassword=123456&password=newPassword
```

## 2.24.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
<b>noResults</b>	(view)	No results matching to the filter criteria were found.
<b>mustUseFilters</b>	assetId	No assetId filter entered
<b>mustUseFilters</b>	lastUpdated	No lastUpdated filter entered
<b>requiredFieldsMissing</b>		Some fields are missing
<b>emailAlreadyExists</b>	email	Email not found
<b>invalidFirstName</b>	firstName	minLength(1)->maxLength(25)

<b>invalidLastName</b>	lastName	minLength(1)->maxLength(25)
<b>invalidCellPhone</b>	cellphone	maxLength(30)
<b>invalidUsername</b>	username	minLength(4)->maxLength(25)
<b>passwordTooshort</b>	password	minLength(6)
<b>invalidDepartment</b>	department	Not found in departments
<b>invalidPosition</b>	position	Not found in positions
<b>invalidSections</b>	sections	Must be variable string
<b>invalidCommision</b>	commision	Must be variable float
<b>invalidEnableReports</b>	enableReports	Must be variable int
<b>invalidDisplayEmail</b>	displayEmail	Must be variable int
<b>invalidDisplayPhone</b>	displayPhone	Must be variable int
<b>invalidDisplayDesks</b>	displayDesks	Must be variable int
<b>invalidUserSelectedLang</b>	userSelectedLang	minLength(2)->maxLength(2)
<b>invalidRegulationCompliance</b>	regulationCompliance	Must be variable int
<b>invalidEmailNotExist</b>	email	User by email not found
<b>invalidCurrentPassword</b>	password	Password does not match
<b>updateFailed</b>		Api request failed
<b>invalidUsername</b>	username	minLength(4)
<b>invalidPassword</b>	password	minLength(4)



## 2.25 AssetLog

Using This Module You Could View Asset Log for 6 minutes range

### 2.25.1 View

#### Request Fields

Field	Type	Null	Default	Comments
assetId	int(11)	No		The asset Id - <b>required</b>
lastUpdated	timestamp	No		The log date range - <b>required</b>

### 2.25.2 Examples

Getting the asset log for 6 minutes range, 3 minutes before and 3 minutes after the lastUpdate date (from 2013-10-20 20:07:00 and 2013-10-20 20:13:00)

```
MODULE=AssetLog&COMMAND=view&FILTER[assetId]=91&FILTER[lastUpdated]=2013-10-20 20:10:00
```

### 2.25.3 Status messages

Note that some of the fields are not actually required, and their corresponding errors will only appear if they are filled anyway.

Message	Field	Description
noResults	(view)	No results matching to the filter criteria were found.
mustUseFilters	assetId	No assetId filter entered
mustUseFilters	lastUpdated	No lastUpdated filter entered

## 2.26 CurrencyHistory

Only view is currently available for this module

The Currency History is showing the rate (converted to USD) by currency code (EUR, JPY etc.).

### 2.26.1

Getting all history rates for a specific currency

```
MODULE=CurrencyHistory&COMMAND=view&FILTER[code]=JPY
```

### 2.26.2 Status messages

Message	Field	Description
noResults	(view command)	No results matching to the filter criteria were found.

## 2.27 LDO

Only view positions is currently available for this module.

Must be used with filter to get results.

### 2.27.1 Examples

Getting all LDO positions for a specific customer.

```
MODULE=LDO&COMMAND=viewPositions&FILTER[customerId]=123465
```

### 2.27.2 Status messages

Message	Field	Description
noResults	(view command)	No results matching to the filter criteria were found.
mustUseFilter	FILTER	API request must include FILTER
requiredFieldsMissing		Some fields are missing
invalidLDOPositionId	id	Position is not available for LDO
invalidReceptionEmployee	receptionEmployeeId	Not found in users
invalidCustomerId	campaignsPermitted	Campaigns permitted is not included

## 2.28 CustomerLoginLog

Only view Customer Login Log is currently available for this module.

You should use with filter to get results.

### 2.28.1 Examples

Getting all Customer Login Log for a specific customer.

```
MODULE=CustomerLoginLog&COMMAND=view&FILTER[customerId]=123465
```

### 2.28.2 Status messages

Message	Field	Description
noResults	(view command)	No results matching to the filter criteria were found.

## 2.29 LDOCustomerLimits

Using This Module you can add, view and edit LDO Customer Limits.

### 2.29.1 Fields

Field	Type	Default	Mandatory (add user)	Editable
customerId	int(11)		✓ (also on edit)	✗
assetId	int(11)	0	✓ (also on edit) (only when type= maxTotalContractsPer Asset)	✗
type	enum('investmentAmount','lossAmount','maxTotalC ontracts','maxTotalContractsPerAsset','maxTrades')		✓ (also on edit)	✗
value	varchar(255)		✓	✓
status	enum('enable','deleted')		✓	✓

### 2.29.2 Examples

#### View LDO Customer Limits for specific user

```
MODULE=LDOCustomerLimits&COMMAND=view&customerId=123
```

#### Edit LDO Customer Limits for a customer

```
MODULE=LDOCustomerLimits&COMMAND=edit&customerId=123&type=lossAmount&value=222
```

#### Edit LDO Customer Limits for a customer

```
MODULE=LDOCustomerLimits&COMMAND=edit&customerId=123&type=maxTotalContractsPerAsset  
&assetId=91&value=222
```

#### Add LDO Customer Limits for a customer

```
MODULE=LDOCustomerLimits&COMMAND=add&customerId=123&type=maxTotalContractsPerAsset  
&assetId=91&value=222&status=deleted
```

### **2.29.3 Status messages**

Message	Field	Description
<b>updateFailed - noResults</b>	(edit command)	No results matching to the filter criteria were found.
<b>limitAlreadyExist</b>	(add command)	The rule already exist with these params
<b>requiredFieldsMissing</b>		Some required fields are missing
<b>invalidCustomerId</b>	customerId	Customer not found
<b>invalidType</b>	type	Ldo customer limits does not contain this type
<b>invalidValue</b>	value	Must be variable int
<b>invalidStatus</b>	status	ldo_customer_limits does not contain this status
<b>invalidAssetId</b>	assetId	assetId was not found in all assets

## 2.30 Affiliate

Using This Module you can add, view and edit affiliates.

### 2.30.1 Fields

Field	Type	Default	Mandatory (add user)	Editable
id	int(11)		✓ (also on edit)	✗
refId	int(11)	0	✗	✓
firstName	Varchar(255)		✓	✓
lastName	varchar(255)		✓	✓

### 2.30.2 Examples

Getting a specific affiliate details

```
MODULE=Affiliate&COMMAND=view&FILTER[id]=123
```

Add new Affiliate

```
MODULE=Affiliate&COMMAND=add&firstName=test&lastName=test
```

Editing the first name of existing Affiliate

```
MODULE=Affiliate&COMMAND=edit&affiliateId=2&firstName=yuval
```

### 2.30.3 Status messages

Message	Field	Description
invalidFirstName	(add/edit command)	First name is not valid
invalidLastName	(add/edit command)	Last name is not valid
invalidAffiliateId	(edit command)	Affiliate id not exists
requiredFieldsMissing	firstName, lastName	Fields were not passed in properly
invalidRefId	refId	Not mandatory, but has to be int
affiliateIdMissing	affiliateId	For edit COMMAND this field is required
noFieldsToEdit	-	Nothing was found to edit
editFailed	-	Edit failed. Please check errors

## 2.31 CustomSettings

Using This Module you can add, view and edit Custom Settings for crm.

### 2.31.1 Fields

Field	Type	Default	Mandatory (add user)	Editable	Comments
id	int(11)		✓ (also on edit)	✗	customTypeid for edit and add
typeName	varchar(255)		✗	✗	
customSettingsValueid	int(11)		✗	✓	The value id
customValueName	varchar(255)		✓	✓	The value name
customTypeid	int(11)		-	✗	Same as 'id' on view

### 2.31.2 Examples

#### Getting a specific Custom Setting details

```
MODULE=CustomSettings&COMMAND=view&FILTER[id]=2
```

#### Add new Custom Settings value

```
MODULE=CustomSettings&COMMAND=add&customTypeid=2&customValueName='Customer on holiday'
```

#### Editing Custom Settings value

```
MODULE= CustomSettings&COMMAND=edit&id=1&customTypeid=2&customValueName=' Customer on holiday'
```

### 2.31.3 Status messages

Message	Field	Description
invalidTypeid	(add/edit command)	First name is not valid
invalidValueName	(add/edit command)	customValueName is not a valid string
requiredFieldsMissing	customTypeid,customValueName	Fields were not passed in properly
invalidTypeid	customTypeid	Custom settings type id not found or doesn't exist
updateFailed	(edit command)	Edit failed
invalidId	(edit command)	Id not found or doesn't exists in custom settings values



## 2.32 SpotFollow

### 2.32.1 Fields

**viewTopTraders:**

Field	Type	Comments
id	int(11)	
assetId	varchar(255)	
customerId	int(11)	
winAmount	varchar(255)	
totalRank	int(11)	
ransactionNum	float	
successRate	decimal(20, 8)	

**viewFollowSessions:**

Field	Type	Comments
followMeTopTradersId	int(11)	
customerId	int(11)	
investmentPerTrade	int(11)	
totalInvestmentLimitation	decimal(20,11)	
noLimit	tinyint(1)	
following	tinyint(1)	
endingFollowDate	datetime	
duration	'1d', '1w', '2w', '1m', '3m', '6m', '1y'	

**add:**

Field	Type	Default	Mandatory (add user)	Comments
topTraderId	int(11)		✓	
customerId	int(11)		✓	
duration	int(11)		✓	'1d', '1w', '2w', '1m', '3m', '6m', '1y'
tradeLimit	int(11)		✓	The value name

<b>globalLimit</b>	int(11)		✓	Same as 'id' on view
<b>noLimit</b>	tinyint(1)		✗	Pass 1 for no-limit follow

**unfollow:**

Field	Type	Mandatory
<b>topTraderId</b>	int(11)	✓
<b>customerId</b>	int(11)	✓

## 2.32.2 Examples

**viewTopTraders:**

```
MODULE=SpotFollow&COMMAND=viewTopTraders&FILTER[assetId]=0
```

**ViewFollowSessions:**

```
MODULE=SpotFollow&COMMAND=viewFollowSessions&FILTER[customerId]=73234&FILTER[following]=1
```

**add:**

```
MODULE=SpotFollow&COMMAND=add&customerId=14&topTraderId=80&duration=3m&tradeLimit=60&globalLimit=1000
```

**unfollow**

```
MODULE=SpotFollow&COMMAND=unfollow&customerId=14&topTraderId=80
```





## 3. Code Examples

In This section we will demonstrate how to make simple calls to our API, using several examples using HTML, PHP, and other programming languages.

## 3.1 HTML

Getting the customer details, by Id

```
<form method="post" action="http://www.api.your-platform.com/Api">

  <input type="hidden" name="api_username" value="your_api_username"/>
  <input type="hidden" name="api_password" value="your_api_password"/>
  <select name="MODULE">
    <option value="Customer">Customer</option>
  </select>

  <br/>
  <input type="text" name="COMMAND" value="view"/><br/>
  <input type="text" name="FILTER[id]" value="1"/><br/>
  <input type="submit" value="Submit"/>
</form>
```

## 3.2 PHP

Getting the customer details, by Id

```
$apiData = array(
'api_username' => 'your_api_user',
'api_password' => 'your_api_pass',
'MODULE'      => 'Customer',
'COMMAND'     => 'view',
'FILTER[id]'  => '1'
);
$URL = 'http://www.api.you-platform.com/Api';

$ch = curl_init($URL);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($apiData));
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);

$result = curl_exec($ch); // run the whole process

echo $result;
curl_close($ch);
```

## 3.3 C#

Getting the customer details, by Id

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Net;
using System.IO;
using System.Xml;
using System.Xml.Linq;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            string urlDemo = "http://www.api.your-platform.com/Api";

            HttpWebRequest request = (HttpWebRequest)WebRequest.Create(urlDemo);
            // Set the Method property of the request to POST.
            request.Method = "POST";
            // Create POST data and convert it to a byte array.
            String postData = "api_username=__YOUR_API_USERNAME__&api_password=__YOUR_API_PASSWORD__";
            postData += "&MODULE=Customer&COMMAND=view&FILTER[id]=1";

            byte[] byteArray = Encoding.UTF8.GetBytes(postData);
            // Set the ContentType property of the WebRequest.
            request.ContentType = "application/x-www-form-urlencoded";
            // Set the ContentLength property of the WebRequest.
            request.ContentLength = byteArray.Length;
            request.Timeout = 60000;
            // Get the request stream.
            Stream dataStream = request.GetRequestStream();
            // Write the data to the request stream.
            dataStream.Write(byteArray, 0, byteArray.Length);
            // Close the Stream object.

            // Get the response.
            HttpWebResponse response = (HttpWebResponse)request.GetResponse();
            // Display the status.
            Console.WriteLine(((HttpWebResponse)response).StatusDescription);
            // Get the stream containing content returned by the server.
            dataStream = response.GetResponseStream();
            // Open the stream using a StreamReader for easy access.
            StreamReader reader = new StreamReader(dataStream);
            // Read the content.
            string responseFromServer = reader.ReadToEnd();
            // Display the content.
            Console.WriteLine(responseFromServer);

            Console.WriteLine("\nClick On Enter To Close Window");
            Console.ReadLine();

            // Clean up the streams.
            reader.Close();
            dataStream.Close();
            response.Close();
        }
    }
}
```



## 4. Appendix

### Common API Errors

Response	Possible Cause(s)
<b>parametersMissing</b>	<ol style="list-style-type: none"><li>1. You haven't Posted anything to the API</li><li>2. A redirect isn't passing your posted params.</li></ol>
<b>CouldNotValidateIP</b>	The IP you post from isn't in the white-listed IP's for this API User
<b>noPermissions</b>	The API user you are using isn't allowed to perform the required action (combination of module & command)
<b>wrongCredentials</b>	Your API Username \ Password are incorrect