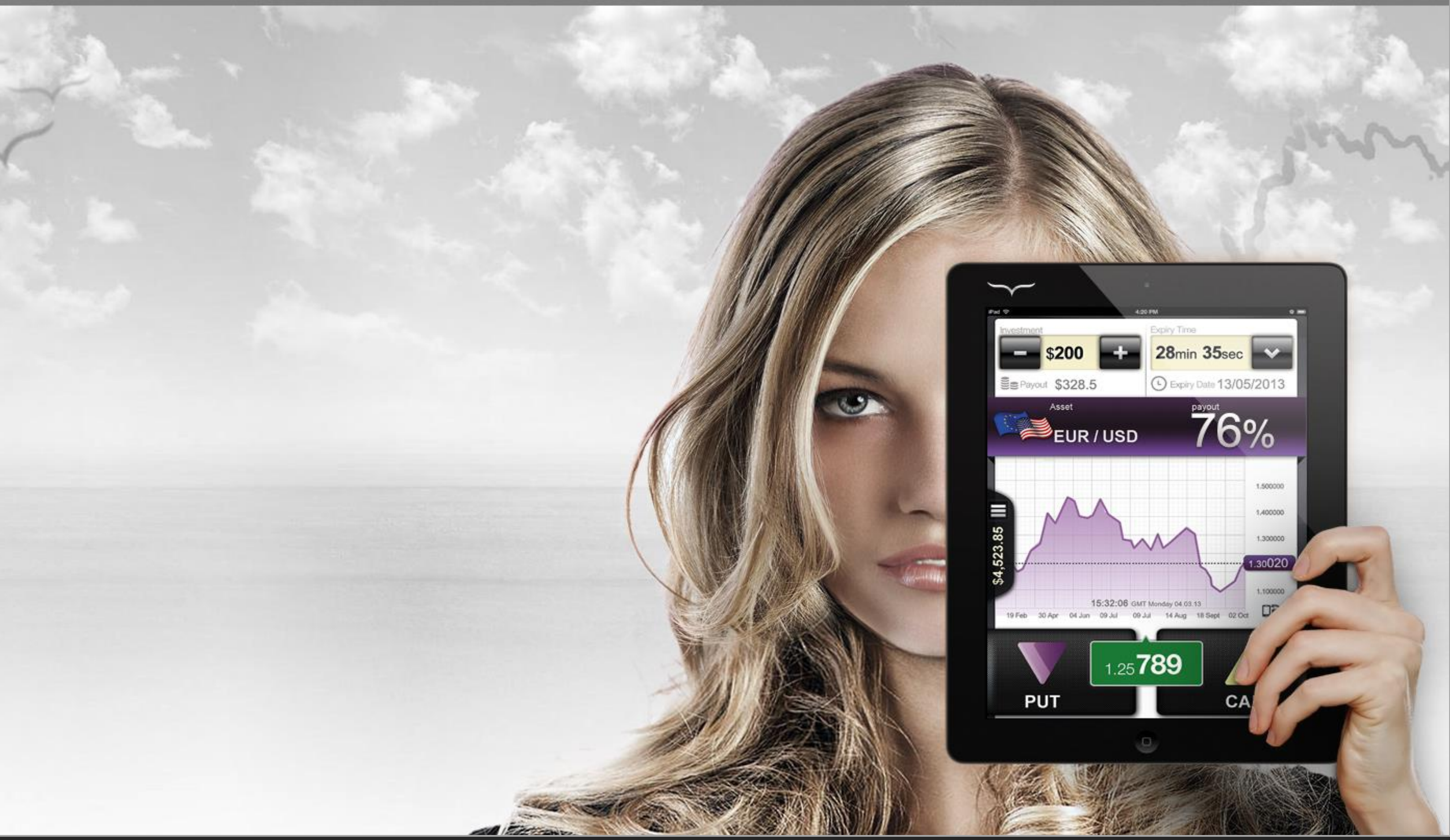
SPOTOPTION
Providing Platforms that Profit



SpotOption

API Documentation

July 2014

Contents of the document

1. General Information

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

In general, the actions you can preform are usually compatible with the SCRUD 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.

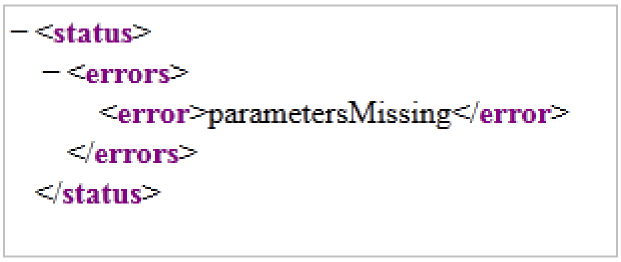
1.1 Prerequisites

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://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), whose Id is 1, 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 1st & the 5th of June, 2013 (unless we surpass the allowed result limit, which ssis 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 5th 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 5th 2013, And this allows us to widen the search, without performing unnecessary calls to the API.

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 simple don't post it, at all (it is optional)ss

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' etc'

Another Exmple:

```
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 recrodsToShow is 1. basically it searched for the 2nd 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)		✗	✓	
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	✗	✗	
birthday	date		✗	✓	

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

*** Required for "validate"**

- When passing a **password** field to the filter, it automatically encrypts it.
- When passing a **subCampaign** field on **add**, a subcampaign will automatically be created, and the add will also return the created subcampaign-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&subCampaignId=45&currency=USD&a_aid=4c1784efdcccb&bi
rthday=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
```

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
```

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
noResults	(view command)	No results matching to the filter criteria were found.
invalidCustomerId	(edit command)	Customer was not found
addFailed	(add command)	General error while trying to add the record
requiredFieldsMissing	(add command)	Some of the required fields were not passed
invalidFirstName	FirstName	First name should be at 2-45

		characters
invalidLastName	LastName	Last name should be at 2-45 characters
invalidGender	gender	Gender should be 'male' or 'female'
invalidEmail	email	Email address was not validated
emailAlreadyExists	email	Email address already exists in customers table
passwordTooshort	password	Password should be at least 6 characters
invalidAuthKey	authKey	Auth key token should be exactly 32 characters length (md5)
invalidCellPhoneNo	cellphone	Cellphone number should 20 characters max
invalidPhoneNo	Phone	Phone number should 20 characters max
invalidPersonalId	personalId	Personal id should 20 characters max
invalidFaxNo	fax	Fax number should 16 characters max
invalidCountry	Country	Country should be an integer between 1 and 239. See country list.
invalidRegistrationCountry	Country	Country should be an integer between 1 and 239. See country list.
invalidState	State	State field is less than 2 characters
invalidCity	City	City field is over than 2 characters
invalidStreetAddress	street	Street field should be less than 45 characters
invalidCurrency	currency	Currency used is invalid
invalidHouseNo	houseNumber	Field is not numeric
invalidApartmentNo	aptNumber	Field is not numeric
invalidApprovesEmailAdsStatus	approvesEmailAds	Field is not numeric
invalidPostalCode	postCode	Postal code should be less than 10 characters
invalidCampaignId	campaignId	Campaign does not exist in campaign table
invalidSubCampaign	subCampaign	Subcampaign parameter is not a string
invalidAffiliateId	a_aid	Field is not numeric
invalidBirthday	birthday	Field is not a string
customerTooYoung	birthday	Customer should be at least 18 years old
invalidEmployee	employeeInChargeId	Employee does not exist in users table
invalidDate	lastTimeActive	Field is not a string
invalidPotential	potential	Field value is not one of the possible values
invalidRisk	risk	Field value is not one of the possible values
invaludRegStatus	regStatus	Field value is not one of the possible values
invalidSaleStatus	saleStatus	Field value is not one of the possible values
invalidCustomerGroup	groupId	Field value is not one of the possible values
invalidReferLink	groupId	The refer link is invalid
invalidIsDemo	isDemo	isDemo should be 0 or 1

invalidSpecialAccountNumber	specialAccountNumber	Field should be String, no longer than 32 chars
specialAccountNumberExists	specialAccountNumber	Special Account Number already exists in customers table
customerIsActive	isDemo (edit)	You cannot edit the demo status of customers with any financial history

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)	-	✗	✓	✓	
employeeInChargeId	int(11)	-	✗	✓	✓	
subCampaign	varchar(100)	-	✗	✓	✗	
birthday	date	-	✗	✗	✓	
referLink	text	-	✗	✓	✗	
a_aid	varchar(255)	-	✗	✓	✓	affiliate id
personalId	varchar(20)	-	✗	✗	✓	
aptNumber	varchar(10)	0	✗	✗	✓	
approvesEmailAds	tinyint(1)	-	✗	✗	✓	
lastTimeActive	timestamp	0000-00-00 00:00:00	✗	✗	✓	

saleStatus	enum('new', 'noAnswer', 'checkNumber', 'callAgain', 'notIntrested', 'inTheMoney', 'noCall')	-	✗	✗	✓	
leadStatus	enum('new', 'noAnswer', 'callAgain', 'checkNumber', 'notSupported', 'notValidPhone')	-	✗	✗	✓	

- When passing a **subCampaign** field on **add**, a subcampaign will automatically be created, and the add will also return the created subcampaign-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 employeeInChargeId of Lead Id 1 to Employee Id 100

```
MODULE=Lead&COMMAND=edit&leadId=1&employeeInChargeId=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
noResults	(view)	No results matching to the filter criteria were found.
addFailed	(add)	General error while trying to add the record
nuSuchLead	(edit)	no lead with give Id was found
requiredFieldsMissing	(add)	Some of the required fields were not passed
invalidFirstName	FirstName	First name should be at 2-45 characters
invalidLastName	LastName	Last name should be at 2-45 characters
invalidEmail	email	Email address was not validated
emailAlreadyExists	email	Email address already exists in customers table
invalidPhoneNo	Phone	Phone number should 20 characters max
invalidCountry	Country	Country should be an integer between 1 and 239. See country list.
invalidCampaignId	campaignId	Campaign does not exist in campaign table

invalidSubCampaign	subCampaign	Subcampaign parameter is not a string
invalidAffiliateId	a_aid	Field is not numeric
invalidReferLink	groupId	The refer link is invalid
invalidRegistrationCountry	Country	Country should be an integer between 1 and 239. See country list.

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		✗	

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 followinf fields are also required**

Field	Type	Null	Default	Comments
product	varchar	No		sixtySeconds
ruleId	int(11)	No		The id of the rule
assetId	int(11)	No		The id of the asset

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 (regular,sixtySeconds,optionsBuilder) using LFILTER :

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

Getting a customers 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&ruleI
d=3&expireTime=1324548000
```

Adding a Sixty Seconds position

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

Adding a regular position

```
MODULE=Positions&COMMAND=add&postion=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
```

Canceling 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
noResults	(view)	No results matching to the filter criteria were found.
addFailed	(add)	General error while trying to add the record
invalidCustomerId	customerId	Customer does not exist in customers table
invalidOptionId	optionId	Field is not numeric
invalidPosition	position	Position should be 'call' or 'put'
invalidAmount	amount	Field is not numeric
insufficientFunds	amount	The position amount is higher than the customer's balance
optionSuspended	optionId	The selected option is currently suspended
customerSuspended	customerId	The selected client is currently suspended
invalidAssetRate	optionId	The asset of the option currently has an invalid rate
optionIsClosed	optionId	The selected option is closed
riskSuspentionActivated	-	Risk suspension for the selected customer is in effect
noTradeableNow		Asset is not currently in trading hours
addOptionFailed-productParameterIsNotSet	-	The product parameter is not set (optionsBuilder/sixtySeconds)
addOptionFailed-givenAssetIdIsNotLikeInRule	-	The asset id is not like the asset id in rule
addOptionFailed-customerIsSuspended	-	The customer is suspend
addOptionFailed-assetRateInvalid	-	Problem with the asset rate

addOptionFailed-theAssetIsNotTradeable	-	The asset is not tradeable
addOptionFailed-theAssetIsSuspended	-	The asset is suspended
addOptionFailed-invalidAmount-shouldBeBetweenXandY	-	The amount sent is illegal - should be between X = minimum and Y = maximum
addOptionFailed-expireTimeIsInThePast	-	The expire time of the option should be in the future
addOptionFailed-expireTimeShouldBeBetweenXandY	-	The expire time should be between X and Y
addOptionFailed-insufficientFunds	-	Insufficient funds to make position
addOptionFailed-riskSuspentionActivated	-	risk suspention activated
noTime	(rollover)	You Can't rollover this position, because noRolloverTime has passed
cannotRolloverCustomOption	(rollover)	Cannot Rollover An Options Builder Option
numberOfRolloversExceeded	(rollover)	Maximum number of rollovers exxeded
positionWinning	(rollover)	Can't rollover winning positions
optionIsClosed	(rollover)	Can't rollover to or from a closed option

2.4 CreditCardUser

Using this module, you could view and edit.

2.4.1 Fields

Field	Type	Null	Default	Mandatory (Edit)	Comments
creditCardId	int(10)	No		✓	The Credit Card Id
username	varchar(40)	No		✘	Allcharge username
netpay_cardId	int(10)	No		✘	Netpay card id
cardType	enum('1', '2', '3', '4', '5', '6')	No		✘	
cardNum	varchar(4)	No		✘	4 Last digits
ExpMonth	int(2)	No		✘	2 digits month. E.g. 02
ExpYear	int(4)	No		✘	4 digits year. E.g.2012
FirstName	varchar(20)	No		✘	
LastName	varchar(20)	No		✘	
Address	varchar(40)	No		✘	
City	varchar(30)	No		✘	
State	varchar(30)	No		✘	
CVV2/PIN	varchar(5)	No		✘	
postCode	varchar(15)	No		✘	
Country	varchar(4)	No		✘	
Phone	varchar(25)	No		✘	
currency	varchar(5)	No		✘	
customerId	int(10)	No		✓	
fundId*	tinyint(3)	No		✘	Used for multiple cards for a client. First fund id is 1. Add credit card is -1.
cardBalance	int(7)	No		✘	Total money deposted by this card (excluding withdrawals)
status	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 (Canceling) 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
noResults	(view command)	No results matching to the filter criteria were 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 backoffice)
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
(method)*	('wire', creditCard', 'bonus', 'cashU', 'ccTerminal', 'depositTerminal' 'pendingDepositTerminal')				Used to determine deposit payment method. The DB field is actually called paymentMethod and has different options.
cardNum	Int			✓	
CVV2/PIN	varchar			✓	
cardType	Int			✓	
ExpMonth	int			✓	example -02
ExpYear	Int			✓	example- 2017
FirstName	varchar(19)			✓	
LastName	varchar(19)			✓	
Address	varchar(45)			✓	
City	varchar(45)			✓	
State	varchar(5)			✓	
postCode	varchar			✓	
Country	int			✓	
Phone	Int(20)			✓	

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

2.5.3 Edit Fields

Field	Type	Null	Default	Comments
status	enum('approved', 'canceled')	No		Only available for pending deposits!
receptionEmployeeId	int(10)	No		
processEmployeeId	int(10)	No		
confirmEmployeeId	int(10)	No		
IPAddess	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=lior&LastName=Test&Address=street50&City=Petahtiqwa&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&transa
ctionID=abcdTest123&paymentMethod=cashU
```

Adding a Cashu deposit using pendingDepositTerminal

```
MODULE=CustomerDeposits&COMMAND=add&method=pendingDepositTerminal&customerId=12&amount=30
0&transactionID=abcdTest123&paymentMethod=cashU
```

Changing processEmployeeId of a deposit 27954

```
MODULE=CustomerDeposits&COMMAND=edit&depositId=27954&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
noResults	(view command)	No results matching to the filter criteria were found.
invalidCustomerId	(edit command)	Customer was not found
addFailed	(add command)	General error while trying to add the record
requiredFieldsMissing	(add command)	Some of the required fields were not passed
invalidFirstName	FirstName	First name should be at 2-45 characters
invalidLastName	LastName	Last name should be at 2-45 characters
invalidCreditCard	fundId	Credit card number does not exist
creditCardAlreadyExists	cardNum	Credit card already exists for this user
invalidAmount	amount	Field is not numeric
invalidCardNo	cardNum	Field is not numeric
invalidCVV	CVV2/PIN	Field is not numeric
invalidCardType	cardType	Field is not numeric
invalidExpMonth	ExpMonth	Field is not numeric
invalidExpYear	ExpYear	Year should be in the future
invalidCountry	Country	Country should be an integer between 1 and 239. See country list.
invalidCity	City	City field is over than 2 characters
invalidAddress	Address	Address should be less than 45 characters
invalidPhoneNo	Phone	Phone should be less than 20 characters
invalidPostalCode	postCode	Postal code should be less than 10 characters
invalidTransactionId	transactionID	The transaction id is invalid or not specified
transactionExists	transactionID	A credit card deposit with the same transaction id already exists
invalidClearingCompany	clearedBy	The clearedBy field does not contain one of the possible clearing companies
paramsError		Generic error - deposit has failed because the clearing company rejected one of the parameters
unableToDeposit		Generic error - deposit was rejected by clearing company
cardDeclined		Generic error - Card was declined by clearing company

2.6 Options

2.6.1 Fields

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

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

Add options – optionsBuilder /sixtySeconds:

```
MODULE=Options&COMMAND=add&product=optionsBuilder&ruleId&1465&assetId=304
```

Getting open options for 10:30:

```
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 options by default filter (all current open options)

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

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
noResults	(view command)	No results matching to the filter criteria were found.
invalidId	id	Invalid option id
optionIsCurrentlyTrading	optionId	Cannot remove option - option is currently trading

2.7 Withdrawals

2.7.1 Fields

Field	Type	Null	Default	Mandatory (add)	Editable	Comments
method	wire, creditCard ccTerminal, ccPendingTerminal, chargeback,bonus, withdrawalTerminal		wire		✗	Withdrawal method
(fundId)	int			If creditCard method ✓ if not ✗	✗	If credit card withdrawal, the fund id of the card
confirmationCode	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

processEmployeeId	int(11)	No		✘	✓	Employee who processed the request
customerId	int(11)	No		✓	✘	Customer who made the request
type	enum('withdrawal')	No	withdrawal		✘	
clearingUserID	int(11)	No		✘	✘	If credit card withdrawal, the credit card user id
transactionID	varchar(32)	No		If ccTerminal or ccPendingTerminal or withdrawalTerminal method ✓ if not ✘	✘	Required for ccTerminal
swiftCode	varchar(12)			If wire method ✓ if not ✘	✓	Swift Code to be entered by approving employee
paymentMethod	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','Qiwi')			If withdrawalTerminal method ✓ if not ✘	✓	
clearedBy	varchar(64)			✘	✘	Optional to add in 'withdrawal Terminal' method
externalId	varchar(255)			✘	✓	

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
```

Adding new Charge Back withdrawal:

```
MODULE=Withdrawal&COMMAND=add&customerId=120&amount=100&clearedBy=allCharge&method=charge
Back
```

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

		has already been approved
invalidConfirmationCode	confirmationCode	Code should be 2 to 32 characters
receptionEmployeeId	invalidReceptionEmployee	Employee does not exist
processEmployeeId	invalidProcessEmployee	Employee does not exist
invalidStatus	invalidStatus	Status value is not one of the possible status values
cancelReason	cancelReason	Field is not a string
withdrawalAmountCannotBeNegative	amount	Withdrawal amount cannot be negative

2.8 Assets

2.8.1 Fields

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

2.9 AssetsHistory

2.9.1 Fields

Field	Type	Null	Default	Comments
date	timestamp	No	CURRENT_TIMESTAMP	
assetId	int(11)	No		
rate	decimal(20,8)	No		
tradePrice	decimal(20,8)	No		TRDPRC_1 field from Reuters
priceTick	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
noResults	(view command)	No results matching to the filter criteria were found.

2.10 Country

Field	Type	Null	Default	Comments
id	int(10)			Country ID
iso	char(2)	No		ISO 2 letters country code
name	varchar(80)	No		Country name
iso3	char(3)	Yes	*NULL*	ISO 3 letters country code
numcode	smallint(6)	Yes	*NULL*	
block	tinyint(1)	No		Block country?
allowRegistration	tinyint(1)	No		Allow registration from this country?
prefix	char(5)	Yes	*NULL*	The country phone prefix
city*				The city that correspond to the user's IP
region*				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
noResults	(view command)	No results matching to the filter criteria were found.

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
noResults	(view command)	No results matching to the filter criteria were found.
updateFailed	(edit command)	Update Failed

2.12 Sub Campaigns

2.12.1 Fields

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

Update: You Can also create new subcampains using SubCampaign:add

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
noResults	(view command)	No results matching to the filter criteria were found.
invalidParam	(add command)	param supplied is too short (shorter than 1 char)
paramAlreadyExists	(add command)	param supplied already exists
invalidMainCampaign	(add command)	mainCampaign supplied is not numeric
noSuchMainCampaign	(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

You can also add new calls by calling to Call:add

2.13.2 Examples

Getting details all the calls related to customer 4

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

2.13.3 Status messages

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

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.

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
```

2.15.3 Status messages

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

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 depossit 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
noResults	(view command)	No results matching to the filter criteria were found.
invalidStartDate	(view command)	Field is not a string.
invalidEndDate	(view command)	Field is not a string.
invalidMainCampaign	(view\edit command)	Field is not numeric
noSuchCampaign	(view\edit command)	No campaign was found with the supplied id
invalidVisitorCount	(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 depossit 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=edgetrackeri
d_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_1234
5
```

2.17.4 Status messages

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

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

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

** 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 commisssion formula should be constructed with the following parameters: [Deposits] , [Withdrawals] , [PNL] , [FTD] , [Turnover] , [GBonus]

turnover - Display the turnover for customers.

PNL - Display the PNL for customers.

givenBonus - Display the total given bonus.

deposits - Display the total deposits.

withdrawals - DIsplay the total withdrawals.

numOfDepositors - Display number of depositors.

numOfCustomers - Display number of customers.

displayCustomerName - Display customer name in the first depositors table.

displayCustomerId - Display customer ID in the first depositors table.

displayParams - Display 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&FTDBaseline=[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

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
customerId	int(11)	No		✓	
direction	ENUM('call','put')	No		✓	
assetId	int(11)	No		✓	
optionId	int(11)	No		✓	
product	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
```

2.20 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 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

2.20.2 Examples

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
```

2.21 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
action	ENUM('getEmploymentStatuses', 'getBusinessNature', 'getFundSource', 'getAllActiveQuestions', 'getAllActiveAnswers')	No		

2.21.2 Add

Request Fields

Field	Type	Null	Default	Comments
customerId	int(11)	No		
formId	int(11)	No		
action	ENUM('insertDynamicAnswer')	No		
keys	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
```

2.22 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

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
```

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
noResults	(view)	No results matching to the filter criteria were found.
invalidCustomerId	customerId	No Customer found for the supplied Id
invalidOption	optionId	No Option found for the supplied Id
invalidUnitsNumber	numberUnits	Number of units must be an int

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 - **required**

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 one or more 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 backoffice

```
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
```

2.25 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 - **required**
lastUpdated	timestamp	No		The log date range - **required**

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.

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= maxTotalContactsPerAsset)	✗
type	enum('investmentAmount','lossAmount','maxTotalContracts','maxTotalContractsPerAsset','maxContractsPerTrade')		✓ (also on edit)	✗
value	varchar(255)		✓	✓
status	enum('enabled','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
updateFailed - noResults	(edit command)	No results matching to the filter criteria were found.
limitAlreadyExist	(add command)	The rule already exist with these params

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";

            // ReadCountries();
            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)
parametersMissing	1. You haven't Posted anything to the API 2. A redirect isn't passing your posted params.
CouldNotValidateIP	The IP you post from isn't in the white-listed IP's for this API User
noPermissions	The API user you are using isn't allowed to perform the required action (combination of module & command)
wrongCredentials	Your API Username \ Password are incorect