



## SpitFire Enterprise API Information

**Access directly to the Database-** We have API's available to insert, delete, and update leads directly from database (single record at a time). An API for export is also available.

**Licensing-** In order to enable API integration please contact SpitFire Sales team and obtain the Username/ Password. This information would be utilized while creating connection to the API service.

**Configuration-** The default URL on which the Service runs is <http://localhost:8090/SpitFireAPI> however it can be modified by editing the base address parameter of SpitFireAPI.exe.config file located in the directory of the API service application.

**Connection Methods-** The SpitFire API allows methods for the client application to connect

1. Using HTTP GET (attaching values to a URL query string)
2. Using HTTP POST (Request and Responses can be XML or JSON based)
3. Using WCF basicHTTPBinding (SOAP 1.1 based clients)
4. Using WCF wsHttpBinding (SOAP 1.2 based clients offering more security and reliability)

### Data Insert API

**Creating live lead using HTTP GET-** The base URL for clients using HTTP GET method would be: <http://{serviceipaddress}:8090/SpitFireAPI/LiveLeadInsertUsingCampaignName> or <http://{serviceipaddress}:8090/SpitFireAPI/LiveLeadInsertUsingListName> (or <http://{serviceipaddress}:8090/SpitFireAPI/LiveleadInsertintoDNC>) depending on whether the campaign name or list name (or voicephone to insert into DNC list) is used as input parameter. This would change if the default URL of the service is modified

**Following is the list of parameters which would be appended to the URL:**

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. Campaign: name of the campaign in which you want to insert the lead
4. ListName: name of the list in which you want to insert the lead
5. VoicePhone: Voicephone that has to be inserted into the DNC List

\*\*\* The rest of the parameters is customer related information \*\*\*

6. name
7. firstname



8. lastname
9. company
10. address
11. city
12. state
13. zipcode
14. country
15. voicephone
16. secVoicephone
17. homephone
18. cellphone
19. faxphone
20. email
21. website
22. xmlData: this is the set of custom fields and their values with the format specified below
23. notes

\*Note: All the customer fields should not contain special characters such as ' or \

#### **Format of xmlData:**

```
<Fields>
  <Field Name="SSN" Value="0123456789 " />
  <Field Name="DOB" Value="01-01-1985 " />
  <Field Name="Married" Value="No" />
</Fields>
```

#### **Below is the Response XML which the client would receive after the post**

```
<PostResultxmlns=http://schemas.datacontract.org/2004/07/API.Business
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ID>0</ID>
  <PostTime>2011-11-29T13:44:23.65625-06:00</PostTime>
  <Result>Error in VerifyVoicePhone: Number not numeric</Result>
</PostResult>
```

ID: This is the unique identifier for the record inserted, 0 if insert failed

PostTime: Timestamp when the post occurred

Result: Success if the post was successful else an error message describing the failure



### Duplicate Check while inserting leads:

Clients may choose to insert duplicate records or have them removed by adjusting a setting in the SpitFireAPI configuration file:

```
<add key="RemoveDuplicates" value="True"/>
```

If set to true, below is the xml response when LiveLeadInsert method is invoked to insert a lead:

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <CampaignCollection />
  <ClientItemCollection />
  <CustomFieldCollection />
  <DialingListCollection />
  <ID>0</ID>
  <PostTime>2015-01-28T17:47:18.71875-06:00</PostTime>
  <Result>Record 9722673279 Already exists.</Result>
</PostResult>
```

### Examples:

```
http://192.168.2.76:8090/SpitFireAPI/LiveLeadInsert?username=API@SpitFire.com&password=
*****&campaign=testCampaign&name=&firstname=Regina&lastname=Thomas&company
=opc&address=3939Beltline&city=Addison&state=TX&zipcode=75001&country=US&voicephon
e=9722673279&secVoicephone=&homephone=&cellphone=&faxphone=&email=&website=&x
mlData=<Fields><Field Name="SSN" Value="0123456789" /><Field Name="DOB" Value="01-01-
1985"/><Field Name="Married" Value="No" /></Fields>&notes=This is test
```

```
http://192.168.2.76:8090/SpitFireAPI/LiveLeadInsertUsingListName?username=API@SpitFire.c
om&password=*****&ListName=testlist&name=&firstname=Cory&lastname=Katz&compa
ny=opc&address=3939Beltline&city=Addison&state=TX&zipcode=75001&country=US&voiceph
one=9722673279&secVoicephone=&homephone=&cellphone=&faxphone=&email=&website=
&xmlData=<Fields><Field Name="SSN" Value="0123456789" /><Field Name="DOB" Value="01-
01-1979"/><Field Name="Married" Value="No" /></Fields>&notes=This is test
```

```
http://192.168.2.76:8090/SpitFireAPI/LiveLeadInsertintoDNC?username=API@SpitFire.com&pa
ssword=*****&voicephone=1234567890
```

**Creating live lead using HTTP POST-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/LiveLeadPost`, the help page specifying the



request and response type (XML or JSON) body is found on <http://{service ip address}:8090/SpitFireAPI/help/operations/LiveLeadPost>

**Creating lead using SOAP based clients-** The WSDL specification for creating leads in conjunction with SOAP based messaging could be found on <http://{service ip address}:8090/SpitFireAPI?wsdl>

### **Methods available for clients are:**

**LiveLeadInsert**(string username, string password, string campaign, string name, string firstname, string lastname, string company, string address, string city, string state, string zipcode, string country string voicephone, string secVoicephone, string homephone, string cellphone, string faxphone, string email, string website, string xmlData, string notes);

**LiveLeadInsertUsingListName**(string username, string password, string ListName, string name, string firstname, string lastname, string company, string address, string city, string state, string zipcode, string country, string voicephone, string secVoicephone, string homephone, string cellphone, string faxphone, string email, string website, string xmlData, string notes);

**LeadInsertintoDNC**(string username, string password, string voicephone);

### **Data Delete API**

**Deleting live lead using HTTP GET-** The base URL for clients using HTTP GET method would be: <http://{serviceipaddress}:8090/SpitFireAPI/LiveLeadDeleteUsingCampaignName> or <http://{service ip address}:8090/SpitFireAPI/LiveLeadDeleteUsingListName> (or <http://{service ip address}:8090/SpitFireAPI/LiveleadDeletefromDNC>), depending on whether the campaign name or list name (or voicephone to delete from DNC List) is used as input parameter. This would change if the default URL of the service is modified

### **Following is the list of parameters which would be appended to the URL:**

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. Campaign: name of the campaign from which the lead should be deleted
4. ListName: name of the list from which the lead should be deleted
5. ClientID: The unique identifier of the record to be deleted
6. VoicePhone: Voicephone that has to be deleted from the DNC List



Below is the Response XML which the client would receive after the post

```
<PostResultxmlns=http://schemas.datacontract.org/2004/07/API.Business
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ID>0</ID>
  <PostTime>2012-09-19T15:11:43.3196511-05:00</PostTime>
  <Result>Success</Result>
</PostResult>
```

ID: 0 if record is successfully deleted

PostTime: Timestamp when the delete occurred

Result: Success if the delete was successful else an error message describing the failure

#### Example:

[http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeleteUsingCampaignName?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&campaign=testCampaign&ClientID=31](http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeleteUsingCampaignName?username=API@SpitFire.com&password=*****&campaign=testCampaign&ClientID=31)

[http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeleteUsingListName?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&ListName=testlist&ClientID=32](http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeleteUsingListName?username=API@SpitFire.com&password=*****&ListName=testlist&ClientID=32)

[http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeletefromDNC?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&voicephone=1234567890](http://192.168.2.76:8090/SpitFireAPI/LiveLeadDeletefromDNC?username=API@SpitFire.com&password=*****&voicephone=1234567890)

**Delete lead using SOAP based clients-** The WSDL specification for deleting leads in conjunction with SOAP based messaging could be found on <http://{service ip address}:8090/SpitFireAPI?wsdl>

#### Methods available for clients are:

**LiveLeadDeleteUsingCampaignName**(string username, string password, string campaign, int ClientID);

**LiveLeadDeleteUsingListName**(string username, string password, string ListName, int ClientID);

**LeadDeletefromDNC**(string username, string password, string voicephone);

#### Data Update API

**Updating live lead using HTTP GET-** The base URL for clients using HTTP GET method would be: <http://{serviceipaddress}:8090/SpitFireAPI/LiveLeadUpdateUsingCampaignName> or <http://{service ip address}:8090/SpitFireAPI/LiveLeadUpdateUsingListName> depending on



whether the campaign name or list name is used as input parameter. This would change if the default URL of the service is modified

**Following is the list of parameters which would be appended to the URL:**

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. Campaign: name of the campaign in which the lead should be updated
4. ListName: name of the list in which the lead should be updated
5. ClientID: The unique identifier of the record to be updated

\*\*\* The remainder of the parameters is customer related information \*\*\*

6. name
7. firstname
8. lastname
9. company
10. address
11. city
12. state
13. zipcode
14. country
15. voicephone
16. secVoicephone
17. homephone
18. cellphone
19. faxphone
20. email
21. website
22. xmlData: this is the set of custom fields and their values with the format specified below
23. notes

Note: All the customer fields should not contain special characters such as ' or \

**Format of xmlData:**

```
<Fields>
<Field Name="SSN" Value="0123456789 " />
<Field Name="DOB" Value="01-01-1985 " />
<Field Name="Married" Value="No" />
```



</Fields>

**Below is the Response XML which the client would receive after the post**

```
<PostResult xmlns=http://schemas.datacontract.org/2004/07/API.Business
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ID>0</ID>
  <PostTime>2012-09-19T15:11:43.3196511-05:00</PostTime>
  <Result>Success</Result>
</PostResult>
```

ID: 0 if record is successfully updated

PostTime: Timestamp when the post occurred

Result: Success if the update was successful else an error message describing the failure

**Example:**

```
http://192.168.2.76:8090/SpitFireAPI/LiveLeadUpdateUsingCampaignName?username=API@SpitFire.com&password=*****&campaign=testCampaign&ClientID=33&name=&firstname=Jane&lastname=Doe&company=opc&address=3939Beltline&city=Addison&state=TX&zipcode=75001&country=US&voicephone=9722673279&secVoicephone=&homephone=&cellphone=&faxphone=&email=&website=&xmlData=<Fields><Field Name="SSN" Value="0123456789" /><Field Name="DOB" Value="09-11-1986"/><Field Name="Married" Value="No" /></Fields>&notes=This is test
```

```
http://192.168.2.76:8090/SpitFireAPI/LiveLeadUpdateUsingListName?username=API@SpitFire.com&password=*****&ListName=testlist&ClientID=33&name=&firstname=Kim&lastname=Jones&company=opc&address=3939Beltline&city=Addison&state=TX&zipcode=75001&country=US&voicephone=8008595924&secVoicephone=&homephone=&cellphone=&faxphone=&email=&website=&xmlData=<Fields><Field Name="SSN" Value="0123456789" /><Field Name="DOB" Value="01-01-1984"/><Field Name="Married" Value="No" /></Fields>&notes=This is test
```

**Update lead using SOAP based clients-** The WSDL specification for updating leads in conjunction with SOAP based messaging could be found on <http://{service ip address}:8090/SpitFireAPI?wsdl>

Methods available for clients are:

**LiveLeadUpdateUsingCampaignName**(string username, string password, string campaign, int ClientID, string name, string firstname, string lastname, string company, string address, string city, string state, string zipcode, string country, string voicephone, string secVoicephone,



string homephone, string cellphone, string faxphone, string email, string website, string xmlData, string notes);

**LiveLeadUpdateUsingListName**(string username, string password, string ListName, int ClientID, string name, string firstname, string lastname, string company, string address, string city, string state, string zipcode, string country, string voicephone, string secVoicephone, string homephone, string cellphone, string faxphone, string email, string website, string xmlData, string notes);

#### GET ALL LISTS:

**Get All Dialing Lists HTTP GET-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/GetAllLists`. This would change if the default URL of the service is modified

#### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive

**Below is the Response XML which the client would receive after the post**

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ClientItemCollection />
  <DialingListCollection>
    <DialingList>
      <ID>1</ID>
      <Name>ptest1</Name>
    </DialingList>
    <DialingList>
      <ID>2</ID>
      <Name>C1_Live_List</Name>
    </DialingList>
  </DialingListCollection>
  <ID>0</ID>
  <PostTime>2014-06-16T10:23:19.0329048-05:00</PostTime>
```





<Result>**Success**</Result>

</PostResult>

ID: 0 for success

PostTime: Timestamp

Result: Success if the lists were returned successfully else an error message describing the failure

### Example:

[http://192.168.2.76:8090/SpitFireAPI/GetAllLists?username=API@SpitFire.com&password=\\*\\*\\*](http://192.168.2.76:8090/SpitFireAPI/GetAllLists?username=API@SpitFire.com&password=***)  
\*\*\*\*\*

### CREATE DIALING LIST:

**Create Dialing List using HTTP GET-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/CreateDialingListUsingListName`. This would change if the default URL of the service is modified

### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. ListName: name of the list that is being created

### Below is the Response XML which the client would receive after the post

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ClientItemCollection />
  <DialingListCollection />
  <ID>778</ID>
  <PostTime>2014-06-16T10:37:48.3297798-05:00</PostTime>
  <Result>Success</Result>
</PostResult>
```

ID: 0 if the list is successfully created

PostTime: Timestamp

Result: Success or failure based on result returned

### Example:

[http://192.168.2.76:8090/SpitFireAPI/CreateDialingListUsingListName?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&listName=NewList](http://192.168.2.76:8090/SpitFireAPI/CreateDialingListUsingListName?username=API@SpitFire.com&password=*****&listName=NewList)



### **DELETE DIALING LIST:**

**Delete Dialing List using HTTP GET-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/DeleteDialingListUsingListName`. This would change if the default URL of the service is modified

### **Following is the list of parameters which would be appended to the URL:**

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. ListName: name of the list that has to be deleted

### **Below is the Response XML which the client would receive after the post**

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ClientItemCollection />
  <DialingListCollection />
  <ID>0</ID>
  <PostTime>2014-06-16T14:42:38.281625-05:00</PostTime>
  <Result>Success</Result>
</PostResult>
```

ID: 0 if list is successfully deleted

PostTime: Timestamp

Result: Success or failure based on result returned

### **Example:**

`http://192.168.2.76:8090/SpitFireAPI/DeleteDialingListUsingListName?username=API@SpitFire.com&password=SpitFireAPI&listName=NewList`

### **DELETE ALL LEADS IN DIALING LIST:**

**Delete all leads in list using HTTP GET-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/DeleteLeadsUsingListName`. This would change if the default URL of the service is modified

### **Following is the list of parameters which would be appended to the URL:**

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. ListName: name of the list from which all leads are to be deleted



Below is the Response XML which the client would receive after the post

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <ClientItemCollection />
    <DialingListCollection />
  </PostResult>
</ID>0</ID>
<PostTime>2014-06-16T15:16:09.29725-05:00</PostTime>
<Result>Success</Result>
</PostResult>
```

ID: 0 if all records in the list are deleted

PostTime: Timestamp

Result: Success or failure based on what was returned

### Example:

[http://192.168.2.76:8090/SpitFireAPI/DeleteLeadsUsingListName?username=API@SpitFire.com  
&password=\\*\\*\\*\\*\\*&listName=NewList](http://192.168.2.76:8090/SpitFireAPI/DeleteLeadsUsingListName?username=API@SpitFire.com&password=*****&listName=NewList)

### SEARCH CALL ITEM:

**Search for Call Item using HTTP GET-** The base URL for clients using HTTP GET method would be <http://{service ip address}:8090/SpitFireAPI/SearchCallItem>. This would change if the default URL of the service is modified

### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. VoicePhone: Voicephone of the lead that is used for searching
4. ListName: Name of the list in which the voicephone is searched

Below is the Response XML which the client would receive after the post

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ClientItemCollection />
  <DialingListCollection />
</ID>0</ID>
<PostTime>2014-06-16T13:28:50.5016548-05:00</PostTime>
```



```
<Result>Success</Result>  
</PostResult>
```

ID: 0 for successful call  
PostTime: Timestamp  
Result: Success or failure based on result returned

### Example:

[http://192.168.2.76:8090/SpitFireAPI/SearchCallItem?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&voicePhone=9722673279&ListName=testCampaign Live List](http://192.168.2.76:8090/SpitFireAPI/SearchCallItem?username=API@SpitFire.com&password=*****&voicePhone=9722673279&ListName=testCampaign Live List)

### EXPORT CALL HISTORY DATA:

**Export Call History Data using HTTP GET-** The base URL for clients using HTTP GET method would be `http://{service ip address}:8090/SpitFireAPI/ ExportHistoryData`. This would change if the default URL of the service is modified

### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive
3. CampaignName
4. StartDate
5. EndDate
6. Skip
7. ListNameList
8. DispositionNameList
9. AgentNameList
10. CustomerName
11. FirstName
12. LastName
13. Company
14. Address
15. City
16. State
17. ZipCode
18. VoicePhone: Voicephone of the lead that is used for searching
19. SecondaryVoicePhone



- 20. HomePhone
- 21. CellPhone
- 22. Email
- 23. IsHit

Below is the Response XML which the client would receive after the post

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <ClientItemCollection>
  <ClientItem>
    <Address>3939 Beltline Road</Address>
    <CallLength>19</CallLength>
    <CallStartTime>2014-03-04T15:04:00</CallStartTime>
    <CampaignName>testCampaign</CampaignName>
    <Cellphone />
    <City />
    <Company>OPC Marketing</Company>
    <Country />
    <CustomerName>John Doe</CustomerName>
    <DispositionName>fourth</DispositionName>
    <EMail>test@email.com</EMail>
    <Faxphone />
    <Fields>
    <Field>
      <Name>id</Name>
      <Value>d143925a-e48d-e311-ba35-00155d5d2601</Value>
    </Field>
    </Fields>
    <FirstName>John</FirstName>
    <Homephone />
    <IsHit>True</IsHit>
    <LastName>Doe</LastName>
    <ListName>Captorra</ListName>
    <Notes />
    <SecVoicephone />
    <State />
    <UserName>Agent 2</UserName>
    <Voicephone>8008595924</Voicephone>
    <Website />
    <WrapUpTime>0</WrapUpTime>
    <Zipcode />
```



```
</ClientItem>
</ClientItemCollection>
<DialingListCollection />
<ID>0</ID>
<PostTime>2014-06-16T13:20:23.2516548-05:00</PostTime>
<Result>Success</Result>
</PostResult>
```

ID: 0 for success

PostTime: Timestamp

Result: Success or failure based on result returned

### Example:

[http://192.168.2.76:8090/SpitFireAPI/ExportHistoryData?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*&campaignName=testCampaign&startDate=2014/02/28](http://192.168.2.76:8090/SpitFireAPI/ExportHistoryData?username=API@SpitFire.com&password=*****&campaignName=testCampaign&startDate=2014/02/28)

00:00:00&endDate=2014/03/29

00:00:00&skip=0&listNameList=&dispositionNameList=&agentNameList=&customerName=&firstname=&lastname=&company=&address=&city=&state=&zipcode=&voicephone=&secVoicephone=&homephone=&cellphone=&email=&isHit=false

### GET ALL CAMPAIGNS:

**Get All Campaigns HTTP GET-** The base URL for clients using HTTP GET method would be <http://{serviceipaddress}:8090/SpitFireAPI/GetAllCampaigns>. This would change if the default URL of the service is modified

### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive

### Below is the Response XML which the client would receive after the post

```
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
<CampaignCollection>
<Campaign>
<ID>2</ID>
<Name>testCampaign</Name>
</Campaign>
<Campaign>
<ID>3</ID>
<Name>MyCampaign</Name>
```



```
</Campaign>
<Campaign>
  <ID>69</ID>
  <Name>Preview</Name>
</Campaign>
<Campaign>
  <ID>1</ID>
  <Name>C1</Name>
</Campaign>
<Campaign>
  <ID>104</ID>
  <Name>PreviewTest1</Name>
</Campaign>
</CampaignCollection>
<ClientItemCollection />
<DialingListCollection />
<ID>0</ID>
<PostTime>2014-08-25T15:24:34.921875-05:00</PostTime>
<Result>Success</Result>
</PostResult>
```

ID: 0 for success

PostTime: Timestamp

Result: Success if the campaigns are returned successfully else an error message describing the failure

### Example:

[http://192.168.2.76:8090/SpitFireAPI/GetAllCampaigns?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*](http://192.168.2.76:8090/SpitFireAPI/GetAllCampaigns?username=API@SpitFire.com&password=*****)

### GET ALL CUSTOM FIELDS:

**Get All CustomFields HTTP GET-** The base URL for clients using HTTP GET method would be `http://{serviceipaddress}:8090/SpitFireAPI/GetAllCustomFields`. This would change if the default URL of the service is modified

### Following is the list of parameters which would be appended to the URL:

1. Username: username given to use SpitFire API, this field is not case sensitive
2. Password: Password given to use SpitFire API, this field is not case sensitive



Below is the Response XML which the client would receive after the post

```
<?xml version="1.0"?>
<PostResult xmlns="http://schemas.datacontract.org/2004/07/API.Business"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <CampaignCollection/>
  <ClientItemCollection/>
  <CustomFieldCollection>
    <CustomField>
      <ID>9</ID>
      <Name>Age</Name>
    </CustomField>
    <CustomField>
      <ID>10</ID>
      <Name>Gender</Name>
    </CustomField>
    <CustomField>
      <ID>11</ID>
      <Name>Balance</Name>
    </CustomField>
  </CustomFieldCollection>
  <DialingListCollection />
</ID>0</ID>
<PostTime>2014-08-25T15:24:34.921875-05:00</PostTime>
<Result>Success</Result>
</PostResult>
```

ID: 0 for success

PostTime: Timestamp

Result: Success if the custom fields are returned successfully else an error message describing the failure

### Example:

[http://192.168.2.76:8090/SpitFireAPI/GetAllCustomFields?username=API@SpitFire.com&password=\\*\\*\\*\\*\\*](http://192.168.2.76:8090/SpitFireAPI/GetAllCustomFields?username=API@SpitFire.com&password=*****)

### List of Methods





URL	Method	Description
CreateDialingListUsingListName	GET	Create a new dialing list using ListName
DeleteDialingListUsingListName	GET	Delete a dialing list using ListName
DeleteLeadsUsingListName	GET	Delete all leads using ListName
ExportHistoryData	GET	Export History
GetAllCampaigns	GET	Get all Campaigns
GetAllCustomFields	GET	Get all CustomFields
GetAllLists	GET	Get All Lists
GetStatus	POST	Service at <a href="http://localhost:8090/SpitFireAPI/getStatus">http://localhost:8090/SpitFireAPI/getStatus</a>
LeadDeletefromDNC	GET	Delete a Live Lead from DNC
LeadInsertintoDNC	GET	Post a live lead into DNC
LiveLeadDeleteUsingCampaignName	GET	Delete a Live Lead using Campaign Name
LiveLeadDeleteUsingListName	GET	Delete a Live Lead using ListName
LiveLeadInsert	GET	Post a live lead with HTTP GET method using Campaign Name
LiveLeadInsert_CallerID	GET	Overloaded Function - Post a live lead with HTTP GET method using Campaign Name and CallerID
LiveLeadInsertUsingListName	POST	Post a live lead with HTTP GET method using ListName
LiveLeadInsertUsingListName_CallerID	GET	Overloaded Function - Post a live lead with HTTP GET method using ListName and CallerID
LiveLeadPost	GET	Post a live lead using HTTP POST method
LiveLeadUpdateUsingCampaignName	GET	Update a Live Lead using Campaign Name
LiveLeadUpdateUsingCampaignName_CallerID	GET	Overloaded method - Update a Live Lead using Campaign Name and CallerID
LiveLeadUpdateUsingListName	GET	Update Live Lead using List Name



LiveLeadUpdateUsingListName_CallerID	GET	Overloaded Function - Update Live Lead using List Name and CallerID
SearchCallItem	GET	Search Call Item