

SOCIAL SURVEY QUICK REFERENCE GUIDE



Tony Mears

"SocialSurvey API provides access to import existing review data from the Social Survey system. This data can be used for business reporting and/or integration of the reviews content into your company websites."

TABLE OF CONTENTS

WHY USE SOCIAL SURVEY API'S	1
YOU MAY USE SOCIALSURVEY API TO:	2
DATA USAGE	2
COMPARE GET VS. POST	4
RESOURCE PATHS	5
SERVICE ENDPOINTS	5
USING THE API KEY TO ACCESS SOCIALSURVEY RESOURCES	5
CURL AUTHORIZATION	6
<i>Example curl Authorization Syntax</i>	6
<i>cURL Code Example</i>	6
PUT SURVEYS REQUEST BODY.....	7
<i>Submit Transactions to Survey</i>	7
<i>Resource URL</i>	7
<i>Method</i>	7
<i>PUT-Surveys Response Code</i>	7
PUT SURVEYS REQUEST BODY TABLE.....	9
<i>PUT-Surveys Response Code</i>	11
GET SURVEY REQUESTS TABLE	12
GET SURVEY REQUESTS FROM SOCIALSURVEY.....	13
<i>Resource URL</i>	13
<i>HTTP Method</i>	13
GET SURVEYS REQUEST BODY	13
<i>GET-Surveys Response Code</i>	14
SURVEYS RESPONSE TABLE.....	17
GET – INCOMPLETE SURVEYS BODY.....	21
<i>Resource URL</i>	21
<i>HTTP Method</i>	21
GET INCOMPLETE SURVEYS BODY	21
<i>GET –Incomplete Surveys Response Code</i>	22

INCOMPLETE SURVEYS RESPONSE TABLE	23
QUERY PARAMETERS.....	25
GET – SURVEY PARAMETERS	25
<i>Resource URL</i>	25
<i>HTTP Method</i>	25
GET – SURVEY PARAMETERS RESPONSE TABLE	26
GET – INCOMPLETE PARAMETERS	27
<i>Resource URL</i>	27
<i>HTTP Method</i>	27
<i>GET Incomplete Parameters Response Table</i>	27
USEFUL SERVER RESPONSES	29
<i>GET - Surveys Successful</i>	30
<i>GET Surveys Invalid Filter</i>	30
<i>GET Surveys/surveyId InvalidsurveyID</i>	30
<i>GET Invalid Parameter Passed</i>	30
<i>PUT Surveys Server Error</i>	31
<i>PUT Service Provider Mismatch</i>	31
<i>PUT Customer1Email is Null or Empty</i>	31
<i>PUT Customer has Already Been Surveyed</i>	32
<i>PUT Transaction is One Year or Older</i>	32
<i>PUT Invalid Date Format</i>	32
<i>PUT Customer Email Provided but No First name</i>	32
<i>PUT Successful With 1 Customer</i>	33
<i>PUT Successful With Customer</i>	33

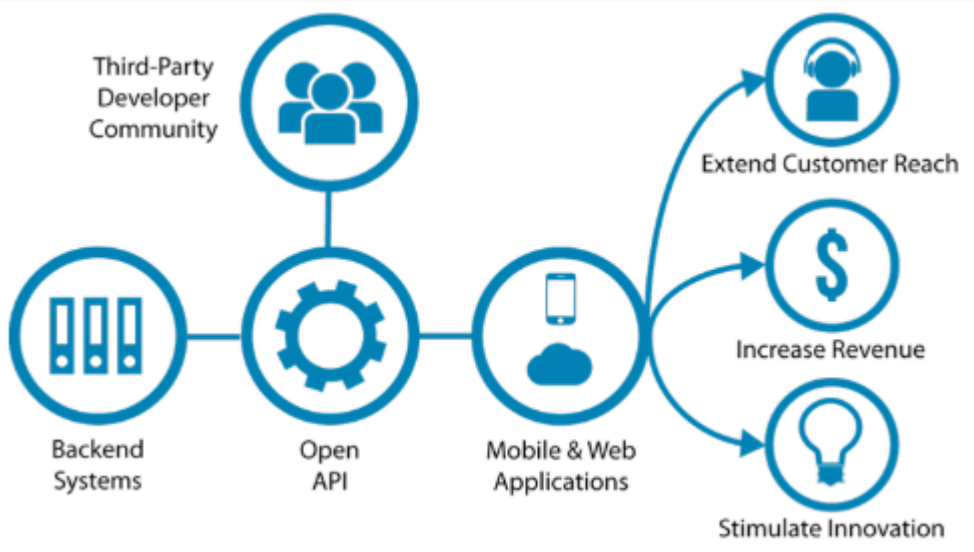
Version Number	Change Description	Effective Date
3.0	Creation of Quick Reference Guide	20-March-2019

WHY USE SOCIAL SURVEY API'S

SocialSurvey supports the principle belief that the survey/review data belongs to you and you may manage it in accordance to your company's operating policies and business rules. You may request for the survey/review data via API or a file export from SocialSurvey. The API can provide you data for a certain date range or a specified number of latest reviews

SocialSurvey is known for managing your online reputation, but that's not what makes us special. What makes SocialSurvey special is the use of data in motion and engaging a network of people via our platform. We want your customers to be a part of this network, where their true voice is amplified and can spark discovery and connect you to new people.

To help new customers discover and connect with you, we've built a Review API, using OAuth2 for authentication, through which you can seamlessly publish your reviews to your personal testimonial page.



To make it easy we have created this quick documentation on ways to consume our review API. There is no fuss to integrate, no messy builds to get up and running. You could use our API to customize how your reviews are displayed when you publish review data on your company site. You can also publish spotlight reviews that highlight the voice of your happiest customers. We're incredibly excited to help you unlock your creativity and bring even more customer testimonials and visibility to your business.

YOU MAY USE SOCIALSURVEY API TO:

POST transaction details to initiate survey sending

GET survey (and review) data collected for your account

DATA USAGE

Social Survey's surveys API allows you to create and maintain a replica database of your survey/review data that you may use to integrate with custom CRM's for analytics, Intranet portals for reporting, or to host reviews on your own public facing website on the Internet.

Caution: Data provided by the surveys API may not be used for real time dynamically generated content and is provided solely for the purpose of generating and updating your locally hosted database.

Note: If users from your organization connected their SocialSurvey profiles with Zillow, it is possible that you may receive some Zillow Reviews data through the SocialSurvey API. Please note that Zillow reviews are subjected to the terms of use of Zillow's data policy. Please be aware that the current policy from Zillow regarding their data usage permits you to dynamically display their reviews content but it does not permit you to store information locally. Click here to view details of the <https://www.zillow.com/howto/API/APITerms.htm> for their data usage.



COMPARE GET VS. POST

The following table compares the two HTTP methods: GET and POST.

Function	GET	POST
BACK button/Reload	Harmless	Data will be re-submitted (the browser should alert the user that the data are about to be re-submitted)
Bookmarked	Can be bookmarked	Cannot be bookmarked
Cached	Can be cached	Not cached
Encoding type	application/x-www-form-urlencoded	application/x-www-form-urlencoded or multipart/form-data. Use multipart encoding for binary data
History	Parameters remain in browser history	Parameters are not saved in browser history
Restrictions on data length	Yes, when sending data, the GET method adds the data to the URL; and the length of a URL is limited (maximum URL length is 2048 characters)	No restrictions
Restrictions on data type	Only ASCII characters allowed	No restrictions. Binary data is also allowed
Security	GET is less secure compared to POST because data sent is part of the URL. Never use GET when sending passwords or other sensitive information!	POST is a little safer than GET because the parameters are not stored in browser history or in web server logs
Visibility	Data is visible to everyone in the URL	Data is not displayed in the URL

RESOURCE PATHS

SocialSurvey APIs are supported on Social Survey’s Test and Production platforms. The resource path for each platform are as follows:

Environment	Resource path
Testing	http://api.socialsurvey.info/v2/
Production	http://api.socialsurvey.me/v2/

All API resources are prefixed with the Resource Paths listed above.

SERVICE ENDPOINTS

Method	HTTP Request	Description
surveys	GET / surveys	Get existing survey data
incompletesurveys	GET /incomplete surveys	Gets existing survey data which has a status of incomplete
surveys	PUT/surveys	Create a new transaction for processing into a survey

USING THE API KEY TO ACCESS SOCIALSURVEY RESOURCES

Authorization is ALWAYS REQUIRED
 “API KEY” tokens are different for Testing vs Production environments

There are two headers to be added in each API:

Name	Required	Location	Description
Content Type	Yes	Header	application/JSON
authorization	Yes	Header	Basic "API_Key"

CURL AUTHORIZATION

“curl” is a command line tool to test API. it is like “poster” addon available in “Google Chrome browser”. Or “Postman” app. Where you can test the API with URL/Request/Header parameters. In “curl” you need to pass the parameters in command line parameters as Linux commands. It is lightweight and easy to use. But the commands need to be prepared and pre-populated with all the parameters and you need to install the software manually.

Example curl Authorization Syntax

```
curl -header "Content-Type: application/json" --header "Authorization: Basic  
"API_KEY""-X GET "https://api.socialsurvey.me/v2/surveys"
```

cURL Code Example

```
curl -d "userid=1&filecomment=This is an image file" --data-binary  
@"/home/user1/Desktop/test.jpg" localhost/uploader.php
```

PUT SURVEYS REQUEST BODY

Submit Transactions to Survey

Submit Transactions data to Social Survey application for surveying clients

Resource URL

<https://api.socialsurvey.me/v2/surveys>

*ensure https is used due to secure connection

Method

PUT

PUT-Surveys Response Code

```
{  
  "transactionInfo": {  
    "transactionRef": string,  
    "transactionDate": Date/Time "yyyy-mm-dd hh:mm:ss",  
    "transactionCity": string,  
    "transactionState": string,  
    "transactionType": string,  
    "customer1FirstName": string,  
    "customer1LastName": string,  
    "customer1Email": string,  
    "customer2FirstName": string,
```

```
    "customer2LastName": string,  
    "customer2Email": string,  
    "buyerAgentFirstName": string,  
    "buyerAgentLastName": string  
    "buyerAgentEmail": string,  
    "sellerAgentFirstName": string  
    "sellerAgentLastName": string,  
    "sellerAgentEmail": string  
  },  
  "serviceProviderInfo": {  
    "serviceName": string,  
    "serviceProviderEmail": string  
  }  
}
```

PUT SURVEYS REQUEST BODY TABLE

Property Name	Value	Description
transactionInfo	nested object	The transaction data to be processed into a survey
transactionInfo.transactionRef	String (Mandatory)	The ID from your system that enables association to its original transaction (e.g. LoanNumber, InvoiceNumber, etc.)
transactionInfo.transactionDate	Date/Time (Mandatory) yyyy-mm-dd hh:mm:ss	The date and time the transaction occurred; typically associated with the closed or funding date of a transaction
transactionInfo.transactionCity	String (Optional)	The city where the transaction occurred
transactionInfo.transactionState	String (Optional)	The state where the transaction occurred
transactionInfo.transactionType	String (Optional)	The type of transaction provided (e.g. Purchase, Refinance, Rental, etc.)
transactionInfo.customer1FirstName	String (Mandatory)	The first name of the primary customer to be surveyed
transactionInfo.customer1LastName	String (Mandatory)	The last name of the primary customer to be surveyed
transactionInfo.customer1Email	String (Mandatory)	The email address of the primary customer to be surveyed
transactionInfo.customer2FirstName	String (Optional/mandatory)	The first name of the secondary customer (co-borrower) to be surveyed.
		NOTE: This field is mandatory if there is an email in "transactionInfo.customer2Email"
transactionInfo.customer2LastName	String (Optional)	The last name of the secondary customer (co-borrower) to be surveyed

transactionInfo.customer2Email	String (Optional)	The email address of the secondary customer (co-borrower) to be surveyed
transactionInfo.buyerAgentFirst Name	String (Optional/mandatory)	The first name of the Buyer's Agent to be Surveyed.
		NOTE: This field is mandatory if there is an email in "transactionInfo.buyerAgentEmail"
transactionInfo.buyerAgentLast Name	String (Optional)	The last name of the Buyer's Agent to be Surveyed
transactionInfo.buyerAgentEmail	String (Optional)	The email of the Buyer's Agent to be Surveyed
transactionInfo.sellerAgentFirst Name	String (Optional/mandatory)	The first name of the Seller's Agent to be Surveyed
		NOTE: This field is mandatory if there is an email in "transactionInfo.sellerAgentEmail"
transactionInfo.sellerAgentLast Name	String (Optional)	The last name of the Seller's Agent to be Surveyed
transactionInfo.sellerAgentEmail	String (Optional)	The email of the Buyer's Agent to be Surveyed
serviceProviderInfo	nested object	The object that contains the servicer information
serviceProviderInfo.serviceProviderName	String (Mandatory)	The name of the person who provided service for the transaction/person
serviceProviderInfo.serviceProviderEmail	String (Mandatory)	The email address of the person who provided service for the transaction/person

PUT-Surveys Response Code

```
{  
  "transactionInfo": {  
    "transactionRef": string,  
    "transactionDate": Date/Time "yyyy-mm-dd hh:mm:ss",  
    "transactionCity": string,  
    "transactionState": string,  
    "transactionType": string,  
    "customer1FirstName": string,  
    "customer1LastName": string,  
    "customer1Email": string,  
    "customer2FirstName": string,  
    "customer2LastName": string,  
    "customer2Email": string,  
    "buyerAgentFirstName": string,  
    "buyerAgentLastName": string,  
    "buyerAgentEmail": string,  
    "sellerAgentFirstName": string,  
    "sellerAgentLastName": string,  
    "sellerAgentEmail": string  
  },  
}
```

```

    "serviceProviderInfo": {
        "serviceProviderName": string,
        "serviceProviderEmail": string
    }
}

```

GET SURVEY REQUESTS TABLE

Property Name	Value	Description
msg	nested object	The server message, which includes the message and the server
msg.message	string	The server response message
msg.code	integer	The server response code
data	nested object	The object, which contains the surveyId object
data.surveyId	nested object	The object which contains the key value pairs of "customer1Email":"surveyId" and "customer2Email":"surveyId"
data.surveyId.customer1Email	long	The surveyId associated with customer1Email
data.surveyId.customer2Email	long	The surveyId associated with customer2Email

GET SURVEY REQUESTS FROM SOCIALSURVEY

GET requests include all required data in the URL. Forms in HTML can use either method by specifying method="POST" or method="GET" (default) in the <form> element.

Resource URL

<https://API.socialsurvey.me/v2/surveys>

HTTP Method

GET

GET SURVEYS REQUEST BODY

Do not supply a request body with this method
Provide parameters in URL (see parameters below)

GET-Surveys Response Code

```
{
  "msg": {
    "message": string,
    "code": integer
  },
  "data": {
    "surveys": [
      {
        "surveyId": long,
        "surveyTitle": string,
        "reviewId": string,
        "transactionInfo": {
          "transactionRef": string,
          "surveySentDateTime": Date/Time (UTC Format),
          "transactionDateTime": Date/Time (UTC Format),
          "transactionCity": string,
          "transactionState": string,
          "transactionType": string,
          "customerFirstName": string,
          "customerLastName": string,
          "customerEmail": string,
```

```
    "buyerAgentFirstName": string,  
    "buyerAgentLastName": string  
    "buyerAgentEmail": string,  
    "sellerAgentFirstName": string  
    "sellerAgentLastName": string,  
    "sellerAgentEmail": string  
  },  
  "serviceProviderInfo": {  
    "serviceName": string,  
    "serviceProviderEmail": string  
  },  
  "review": {  
    "source": string,  
    "reviewCompletedDateTime": Date/Time (UTC Format),  
    "reviewUpdatedDateTime": Date/Time (UTC Format),  
    "rating": string,  
    "summary": string,  
    "description": string,  
    "agreedToShare": boolean,  
    "verifiedCustomer": boolean,  
    "retakeSurvey": boolean,  
    "surveyResponses": [  

```

```
{
  "question": string,
    "type" string,
      "answer": string
}
],
  "reportedAbusive": boolean
},
  "reviewStatus": string
}
]
}
}
```

SURVEYS RESPONSE TABLE

Property Name	Value	Description
msg	nested object	The server message, which includes the message and the server
msg.message	string	The server response message
msg.code	integer	The server response code
data	nested object	The object which contains surveys
data.surveys[]	list	The list of surveyData
data.surveys.surveyId	long	The ID associated with that survey. NOTE: Not available for Zillow reviews
data.surveys.reviewId	string	The unique ID associated with the review NOTE: reviewId is only available after the customer completes the survey
data.surveys.transactionInfo	nested object	The object which contains the transaction information

data.surveys.transactionInfo.transactionRef	string	An ID from your system, given in the initial survey request, that enables association to its original transaction (eg. LoanNumber, Invoice Number, etc.)
data.surveys.transactionInfo.surveySentDateTime	Date/Time (UTC Format)	The timestamp when the survey was sent
data.surveys.transactionInfo.transactionDateTime	Date/Time (UTC Format)	The timestamp of when the transaction occurred. Typically associated with the funding date or closed date of a transaction
data.surveys.transactionInfo.transactionCity	string	The city where the transaction occurred
data.surveys.transactionInfo.transactionState	string	The state where the transaction occurred
data.surveys.transactionInfo.transactionType	string	The type of transaction (e.g. Purchase, Refinance, Rental, etc.)
data.surveys.transactionInfo.customerFirstName	string	The customer's first name
data.surveys.transactionInfo.customerLastName	string	The customer's last name
data.surveys.transactionInfo.customerEmail	string	The customer's email address

data.surveys.serviceProviderInfo	nested object	The object which contains servicer information
data.surveys.serviceProviderInfo.serviceProviderName	string	The name of the person who provided the service for the transaction / person being surveyed
data.surveys.serviceProviderInfo.serviceProviderEmail	string	The email address of the person who provided the service for the transaction / person being surveyed
data.surveys.review	nested object	The object that contains the review data
data.surveys.review.source	string	The source of the review <ul style="list-style-type: none"> • agent • admin • encompass • dotloop • lonewolf • ftp • Zillow • 3rd party
data.surveys.review.reviewCompletedDateTime	Date/Time (UTC Format)	The timestamp of when the survey was completed
data.surveys.review.reviewUpdatedDateTime	Date/Time (UTC Format)	The timestamp of when the survey was updated (e.g. survey re-take)

data.surveys.review.rating	string	The overall rating for the survey response
data.surveys.review.summary	string	The summary of the transaction (e.g. "Completed transaction on July 2017 in San Francisco, California")
data.surveys.review.description	string	The content of the review
data.surveys.review.agreedToShare	boolean	Whether the customer agreed to sharing on social media
data.surveys.review.verifiedCustomer	boolean	Whether the review has been verified against the originating transaction
data.surveys.review.retakeSurvey	boolean	Whether the review has been flagged as a re-take
data.surveys.review.surveyResponses	list	The survey responses
data.surveys.review.surveyResponses.question	string	The survey question
data.surveys.review.surveyResponses.type	string	The type of the response
		<ul style="list-style-type: none"> • Numeric • Text • Experience

data.surveys.review.surveyResponses.answer	string	The response provided by the customer
data.surveys.review.reportedAbusive	boolean	Whether the response has been marked as abusive
Data.surveys.reviewStatus	string	The state of the review (e.g. completed, incomplete)

GET – INCOMPLETE SURVEYS BODY

GET requests include all required data in the URL. Forms in HTML can use either method by specifying method="POST" or method="GET" (default) in the <form> element.

Resource URL

<https://API.socialsurvey.me/v2/surveys>

HTTP Method

GET

GET INCOMPLETE SURVEYS BODY

Do not supply a request body with this method
Provide parameters in URL (see parameters below)

GET –Incomplete Surveys Response Code

```
{
  "msg": {
    "message": string,
    "code": integer
  }
  "surveys": [
    {
      "surveyId": long,
      "reviewId": string,
      "transactionInfo": {
        "transactionRef": string,
        "surveySentDateTime": Date/Time (UTC Format),
        "transactionDateTime": Date/Time (UTC Format),
        "transactionCity": string,
        "transactionState": string,
        "transactionType": string,
        "customerFirstName": string,
        "customerLastName": string,
        "customerEmail": string,
      },
      "serviceProviderInfo": {
        "serviceProviderName": string,
        "serviceProviderEmail": string
      }
    }
  ]
}
```

INCOMPLETE SURVEYS RESPONSE TABLE

Property Name	Value	Description
msg	nested object	The server message, which includes the message and the server
msg.message	string	The server response message
msg.code	integer	The server response code
data	nested object	The object which contains surveys
data.surveys[]	list	The list of survey data
data.surveys.surveyId	long	The ID associated with that survey.
		NOTE: Not available for Zillow reviews
data.surveys.reviewId	string	The unique ID associated with the review NOTE: reviewId is only available after the customer completes the survey
data.surveys.transactionInfo	nested object	The object which contains the transaction information

data.surveys.transactionInfo.transactionRef	string	An ID from your system, given in the initial survey request, that enables association to its original transaction (eg. LoanNumber, Invoice Number, etc.)
data.surveys.transactionInfo.surveySentDateTime	Date/Time (UTC Format)	The timestamp when the survey was sent
data.surveys.transactionInfo.transactionDateTime	Date/Time (UTC Format)	The timestamp of when the transaction occurred. Typically associated with the funding date or closed date of a transaction
data.surveys.transactionInfo.transactionCity	string	The city where the transaction occurred
data.surveys.transactionInfo.transactionState	string	The state where the transaction occurred
data.surveys.transactionInfo.transactionType	string	The type of transaction (e.g. Purchase, Refinance, Rental, etc.)
data.surveys.transactionInfo.customerFirstName	string	The customer's first name
data.surveys.transactionInfo.customerLastName	string	The customer's last name
data.surveys.transactionInfo.customerEmail	string	The customer's email address
data.surveys.serviceProviderInfo	nested object	The object which contains service provider information

data.surveys.serviceProviderInfo.serviceProviderName	string	The name of the person who provided the service for the transaction / person being surveyed
data.surveys.serviceProviderInfo.serviceProviderEmail	string	The email address of the person who provided the service for the transaction / person being surveyed

QUERY PARAMETERS

Query parameters are the most common type of parameters. They appear at the end of the request URL after a question mark (?), with different name=value pairs separated by ampersands (&). Query parameters can be required and optional. Query parameters can be primitive values, arrays and objects. API provides several ways to serialize objects and arrays in the query string.

GET – SURVEY PARAMETERS

GET requests include all required data in the URL. Forms in HTML can use either method by specifying method="POST" or method="GET" (default) in the <form> element.

Resource URL

<https://API.socialsurvey.me/v2/surveys>

HTTP Method

GET

GET – SURVEY PARAMETERS RESPONSE TABLE

Parameter Name	Value	Description
Optional Path Parameters		
surveyId	long	The surveyId of the survey of interest
Optional Query Parameters		
count	string	The number of returned surveys requested Default: 1000
start	string	The starting index Default:0
state	string	A survey response <ul style="list-style-type: none"> • “Unpleasant” • “OK” • “Great”
status (Deprecated) <ul style="list-style-type: none"> • Use the incompleteSurveys endpoint 	string	The status of the survey <ul style="list-style-type: none"> • “complete” • “incomplete”
startSurveyID	long	The starting surveyId to retrieve survey responses
startReviewDateTime	string	The starting DateTime, of when a survey was completed, to retrieve survey responses

startTransactionDateTime	string	The starting DateTime, of when a transaction was completed, to retrieve survey responses
user	string	The serviceProviderEmail of the transaction
IncludeManagedTeam	string	The filter that includes or excludes members that are managed by the specific user given in the User parameter

GET – INCOMPLETE PARAMETERS

GET requests include all required data in the URL. Forms in HTML can use either method by specifying method="POST" or method="GET" (default) in the <form> element.

Resource URL

<https://API.socialsurvey.me/v2/surveys>

HTTP Method

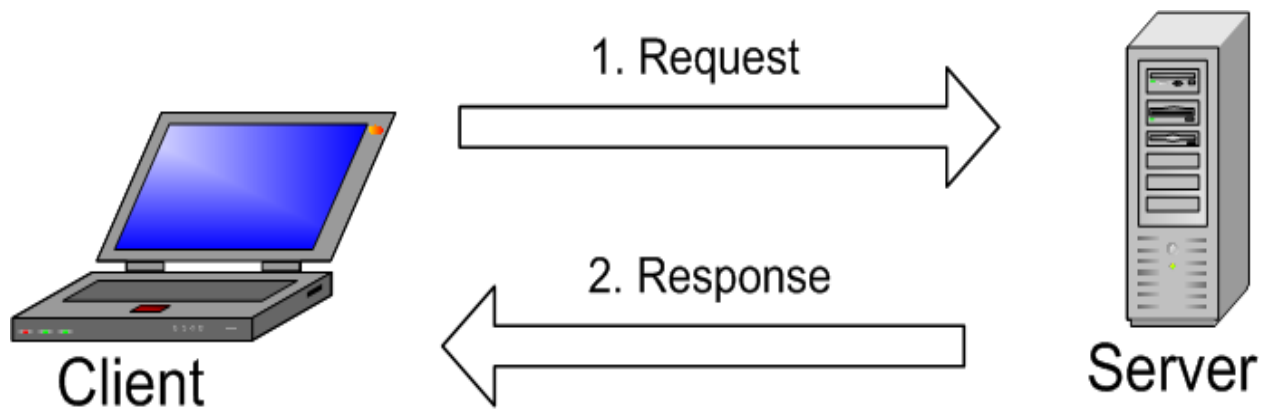
GET

GET Incomplete Parameters Response Table

Parameter Name	Value	Description
Optional Query Parameters		
count	string	The number of returned surveys requested Default: 1000
start	string	The starting index Default:0
startSurveyID	string	The starting surveyId to retrieve survey responses

startTransactionDateTime	string	The starting DateTime, of when a transaction was completed, to retrieve survey responses
user	string	The serviceProviderEmail of the transaction
IncludeManagedTeam	string	The filter that includes or excludes members that are managed by the specific user given in the User parameter

USEFUL SERVER RESPONSES



GET - Surveys Successful

```
{
  "msg": {
    "message": "Request Successfully processed",
    "code": 200
  },
  "data": {
    "surveys": [<SURVEY_DATA>]
  }
}
```

GET Surveys Invalid Filter

```
{
  "msg": {
    "message": "Unsupported filter parameter: <QUERY PARAMETER>",
    "code": 400
  }
}
```

GET Surveys/surveyId InvalidsurveyID

```
  "msg": {
    "message": "No record found for id",
    "code": 404
  }
}
```

GET Invalid Parameter Passed

```
{
  "msg": {
```

```
        "message": "Passed parameter <PASSED PARAMETER> is invalid",
        "code": 400
    }
}
```

PUT Surveys Server Error

```
{
    "msg": {
        "message": "Could not process request due to server error",
        "code": 500
    }
}
```

PUT Service Provider Mismatch

```
{
    "msg": {
        "message": "Can not process the record. No service provider found with
        email address :
        "serviceProviderEmail",
        "code": 406
    }
}
```

PUT Customer1Email is Null or Empty

```
{
    "msg": {
        "message": "Invalid input passed. Customer1Email can't be null or
        empty",
        "code": 400
    }
}
```

PUT Customer has Already Been Surveyed

```
{
  "msg": {
    "message": "Cannot process the record. A survey request for customer
    \"customer1FirstName\"
    has already been received",
    "code": 406
  }
}
```

PUT Transaction is One Year or Older

```
{
  "msg": {
    "message": "Can not process the record. Request for customer
    \"customer1FirstName\" is older
    than 365 days.",
    "code": 406
  }
}
```

PUT Invalid Date Format

```
{
  "msg": {
    "message": "Transaction Date with invalid format",
    "code": 400
  }
}
```

PUT Customer Email Provided but No First name

```
{
  "msg": {
    "message": "Invalid input passed. customer2FirstName can't be null or
    empty",
    "code": 400
  }
}
```

```
}
```

PUT Successful With 1 Customer

```
{  
    "message": "Survey successfully created",  
    "code": 201  
},  
  "data": {  
    "surveyId": {  
      "customer1Email": "surveyId"  
    }  
  }  
}
```

PUT Successful With Customer

```
{  
  "msg": {  
    "message": "Survey successfully created",  
    "code": 201  
  },  
  "data": {  
    "surveyId": {  
      "customer1Email": "surveyId",  
      "customer2Email": "surveyId"  
    }  
  }  
}
```