



twoPLUGS

# API Calls Documentation

Version 1.0



twoPLUGS

## Document Control Sheet

Document Information			
<b>Description</b>	This document contains a description of all the calls used in the twoPLUGSAPI along with all the parameters it needs and a short explanation of what its output consists of.		
<b>Issue Date</b>	July-11-2019	<b>Author</b>	Shafiq Shahin
<b>Approved By</b>	Hardeep Kukreja		
<b>Electronic Path</b>			

Distribution List		
Name	Title	Location

Authorizations				
Stakeholder 1	<b>Name</b>		<b>Title</b>	
	<i>Signature</i>		<i>Date</i>	
Quality Control				
<b>Quality</b>	<i>Signature</i>		<i>Date</i>	

## Notes:

Please note that several portions of this document will be updated as the project progresses.



twoPLUGS

# Contents

Profile Related Calls .....	4
Service Related Calls .....	7
User Related Calls .....	13
Message Related Calls.....	18
Review Related Calls .....	19
Servicematch Related Calls .....	20
Notification Related Calls.....	23
Transaction Related Calls.....	25
Other Calls.....	30



twoPLUGS

# Profile Related Calls

**GET**      `{{base_url}}/profiles/{id}`

**Returns:**                      **(Profile)** The profile record with the specified id

**Parameters:**              Projection                      (Default =ProfileDetail)

**Headers:**                      Authentication                      (bearer {{token}})

   Content-Type                      (Default = application/x-www-form-urlencoded)

**GET**      `{{base_url}}/profiles/search/getByUserId`

**Returns:**                      **(Profile)** The profile record with the specified user id

**Parameters:**                      **(Integer)** user\_id

   Projection                      (Default =ProfileDetail)

**Headers:**                      Authorization                      (bearer {{token}})

**GET**      `{{base_url}}/profiles/search/getByUserUsername`

**Returns:**                      **(Profile)** The profile record with the specified username

**Parameters:**                      **(String)** username

   Projection                      (Default =ProfileDetail)

**Headers:**                      Authorization                      (bearer {{token}})



**POST**      {{base\_url}}/profiles/transferreds

**Returns:**                    (Result) Transfers Eeds from one profile to another

**Body:**                    (Integer) fromuser

                              (Integer) touser

                              (Integer) amount

**Headers:**                Authorization            (bearer {{token}})

**POST**      {{base\_url}}/profiles/sendValidateSMS

**Returns:**                    (Result) Sends an SMS message with a validation code to the user

**Body:**                    (String) phoneNo

                              (Integer) user\_id

**Headers:**                Authorization            (bearer {{token}})

**POST**      {{base\_url}}/profiles/validateSMS

**Returns:**                    (Result) Validates the user's phone number if the message matches the one sent

**Body:**                    (String) message

                              (Integer) user\_id

**Headers:**                Authorization            (bearer {{token}})



**POST**      {{base\_url}}/profiles/refer

**Returns:**              (Result) Sends a referral email

**Body:**                (String) emails  
                            (Integer) user\_id

**POST**      {{base\_url}}/profiles/verifyno

**Headers:**            Authorization            (bearer {{token}})

**Returns:**              (Boolean) Weather the number can be used for referrals or if it was already taken

**Body:**                (String) phone  
                            (Integer) user\_id

**Headers:**            Authorization            (bearer {{token}})

**PATCH**      {{base\_url}}/profiles/{id}

**Returns:**              (Profile) Updates the profile record and returns the updated record

**Body:**                <ANY MEMBER VARIABLE OF PROFILE>

**Headers:**            Authorization            (bearer {{token}})



twoPLUGS

## Service Related Calls

**GET**

`{{base_url}}/services/search/getByCategoryIdInAndNetwork`

**Returns:** `(Service[])` The service record with the specified search criteria

**Parameters:** `(Integer)` category\_id

`(Integer)` network\_id

`(Enum)` service\_type

`(Enum)` type

**Headers:** Authentication (bearer `{{token}}`)

**GET**

`{{base_url}}/categories/search/findByParentId`

**Returns:** `(Category[])` All the subcategories under a specified parent



## twoPLUGS

**Parameters:** (Integer) parent\_id

**Headers:** Authentication (bearer {{token}})

### GET

{{base\_url}}/services/search/getTop5ByNetworkIdA

**Returns:** (Service[]) The service record with the specified search criteria

**Parameters:** (Integer) network\_id

(Integer) category\_id

Projection (Default =ServiceDetail)

### GET

{{base\_url}}/services/{id}

**Headers:** Authentication (bearer {{token}})

**Returns:** (Service) The service record with the specified id

**Headers:** Authentication (bearer {{token}})

Content-Type (Default = application/json)

### GET

{{base\_url}}/services/search/getByUserId

**Returns:** (Service) The service record with the specified user id

**Parameters:** (Integer) user\_id

**Headers:** Authentication (bearer {{token}})





## twoPLUGS

**GET**      `{{base_url}}/complaints/{id}`

**Returns:**                      **(Complaint)** The complaint record with the specified id

**Headers:**                      Authentication                      (bearer {{token}})

**GET**      `{{base_url}}/userdatas/search/getByServiceId`

**Returns:**                      **(Userdata[])** The Userdata objects associated with the service which contain images of the service

**Parameters:**                      **(Integer)** service\_id

Projection                      (Default =UserdataDetail)

**Headers:**                      Authentication                      (bearer {{token}})

**GET**                                      `{{base_url}}/services/search/getByNameIgnoreCaseContains`

**Returns:**                      **(Service)** The profile record with the specified id

**Parameters:**                      **(String)** keyword

**(Enum)** serviceType

**(Integer)** size

**Headers:**                      Authentication                      (bearer {{token}})



## twoPLUGS

**GET** {{base\_url}}/services/smartSearch

**Returns:** (Service[]) The services relevant to the search keyword

**Parameters:** (String) keyword  
(Enum) serviceType  
(Integer) size

**Headers:** Authentication (bearer {{token}})

**POST** {{base\_url}}/profiles/uploadprofileimage

**Returns:** (Result) Weather or not the profile image was uploaded correctly or if any issues were encountered

**Body:** (File) uploadingFiles

**Headers:** Authentication (bearer {{token}})

**POST** {{base\_url}}/services

**Returns:** (Complaint) Creates and returns the service record created

**Body:** (String) user  
(String) name  
(String) description  
(String) category



## twoPLUGS

(String) network  
(Integer) price  
(Integer) refund  
(Integer) refund\_valid  
(Integer) percentage  
(Enum) payment  
(Enum) service\_type  
(Integer) sales  
(Integer) reviews  
(Integer) type  
(String) image  
(Integer) videourl  
(Integer) active  
(Integer) bid

**Headers:** Authentication (bearer {{token}})

**POST** {{base\_url}}/complaints

**Returns:** (Complaint) Creates and returns the complaint record created

**Body:** (Integer) complainerId  
(Integer) userId  
(Integer) transactionId



## twoPLUGS

(Integer) serviceId

(String) subject

(String) Content

**Headers:**

Authentication	(bearer {{token}})
Content-Type	(Default = application/json)

**POST**      {{base\_url}}/services/setDefaultImage

**Returns:**      (Result) Whether the default image was changed correct of if any issues were encountered

**Body:**

(Integer) image\_id

(Integer) service\_id

**Headers:**

Authentication	(bearer {{token}})
----------------	--------------------

**POST**      {{base\_url\_unauth}}/services/servicearoundyou

**Returns:**      (Service) The profile record with the specified id

**Body:**      (String) lat



twoPLUGS

(String) log  
(Integer) user\_id  
(Integer) category\_id  
(Integer) limit

**PATCH**    {{base\_url}}/services/{id}

**Returns:**                (Service) Updates the service record and returns the updated record

**Body:**                    <ANY MEMBER VARIABLE OF PROFILE>

**Headers:**                Authorization                (bearer {{token}})

                              Content-Type                (Default = application/json)

**DELETE**    {{base\_url}}/services/

**Returns:**                (Result) Weather the service was deleted correctly or if any errors were encountered

**Parameters:**           (Integer) id

**Headers:**                Authorization                (bearer {{token}})

## UserRelated Calls



**GET**      `{{base_url}}/users/{id}`

**Returns:**                **(User)** The user record with the specified id

**Parameters:**        Projection                (Default = UserDetails)

**Headers:**            Authentication            (bearer {{token}})

                         Content-Type                (Default = application/json)

**GET**      `{{base_url}}/me`

**Returns:**                **(User)** The user record that is currently logged in

**Headers:**            Authentication            (bearer {{token}})

**GET**                                `{{base_url}}/users/search/getByUsernameIgnoreCas`

**Returns:**                **(User)** The user record with the specified id

**Parameters:**        **(String)** username

**(Integer)** id

**Headers:**            Authentication            (bearer {{token}})

**POST**      `{{base_url_unauth}}/checkemail`

                         Content-Type                (Default = application/json)

**Returns:**                **(Boolean)** Weather the email is valid for registration or if it is taken



## twoPLUGS

**Body:** (String) email

**POST** {{base\_url\_unauth}}/checkusername

**Returns:** (Boolean) Weather the username is valid for registration or if it is taken

**Body:** (String) username

**POST** {{base\_url\_unauth}}/users/createUser

**Returns:** (User) Creates and returns the user record created

**Body:** (String) username

(String) email

(String) password

(String) longitude

(String) latitude

**Headers:** Content-Type (Default = application/json)

**POST** {{base\_url\_unauth}}/oauth/token

**Returns:** (Token) An access token to call the API

**Body:** (String) username

(String) password

**Parameters:** grant\_type (Default = password)

scope (Default = read)



## twoPLUGS

**Headers:** Authentication (Basic bW9iaWxIY2xpZW50OnR3b3BsdWdzLmNvbQ==)

**POST** {{base\_url\_unauth}}/facebooklogin

**Returns:** (Result) An access token to call the API

**Body:** (String) email  
(String) facebook\_id  
(String) firstname  
(String) lastname  
(String) username  
(String) device\_token

**Body:** grant\_type (Default = password)  
scope (Default = read)

**POST** {{base\_url\_unauth}}/forgetPassword

**Returns:** (Result) Whether an email was sent to an account to change its password or if any issues were encountered

**Body:** (String) email





## twoPLUGS

### POST {{base\_url}}/resetpassword

**Returns:** (Result) Whether an email was sent to an account to change its password or if any issues were encountered

**Body:** (Integer) user\_id  
(String) oldpassword  
(String) newpassword

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/changeusername

**Returns:** (Result) Whether an email was sent to an account to change its password or if any issues were encountered

**Body:** (Integer) user\_id  
(String) newname

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/logout

**Returns:** (Result) Whether an email was sent to an account to change its password or if any issues were encountered

**Body:** (Integer) user\_id

**Headers:** Authentication (bearer {{token}})



twoPLUGS

**POST**      {{base\_url\_unauth}}/oauth/token

**Returns:**                      (Token) An access token to call the API

**Parameters:**              grant\_type                      (Default = refresh\_token)

                                 refresh\_token

                                 scope                      (Default = read)

**Headers:**                      Authentication                      (Basic bW9iaWxly2xpZW50OnR3b3BsdWdzLmNvbQ==)

## MessageRelated Calls

**GET**                      {{base\_url}}/messages/search/findBySender\_IdOrReceiver\_Id/

**Returns:**                      (Message) The message record with a matching sender or receiver

**Parameters:**                      (Integer) sid

                                 (Integer) rid

                                 (Integer) size

                                 Projection                      (Default =MessageDetail)

**Headers:**                      Authentication                      (bearer {{token}})



twoPLUGS

**GET**      `{{base_url}}/messages/{id}`

**Returns:**                      (Message) The message record with the specified id

**Headers:**                      Authentication              (bearer {{token}})

## ReviewRelated Calls

**GET**                      `{{base_url}}/reviewses/search/getReviewsByServiceId`

**Returns:**                      (Review) The review record with the matching service id

**Parameters:**                      (Integer) service\_id

Projection                      (Default =ReviewsDetail)

**Headers:**                      Authentication              (bearer {{token}})



**GET**      `{{base_url}}/reviewses/{Id}`

**Returns:**                      (Review) The review record with a matching id

**Parameters:**              Projection                      (Default =ReviewsDetail)

**Headers:**                      Authentication                      (bearer {{token}})

**POST**      `{{base_url}}/reviewses`

**Returns:**                      (Review) Creates and returns a review object

**Body:**                      (Integer) stars

                    (Integer) content

                    (String) service                      (Link to user i.e. "/users/{Id}")

                    (String) buyer                      (Link to user i.e. "/users/{Id}")

                    (String) seller                      (Link to user i.e. "/users/{Id}")

**Headers:**                      Authentication                      (bearer {{token}})

                    Content-Type                      (Default = application/json)

## Servicematch Related Calls

**GET**      `{{base_url}}/servicematches/{Id}`

**Returns:**                      (Servicematch) The Servicematch object with the matching id



## twoPLUGS

**Parameters:** Projection (Default =ServicematchDetail)

**Headers:** Authentication (bearer {{token}})

**GET** {{base\_url}}/servicematches/search/getByUserId

**Returns:** (Servicematch) The Servicematch object with the matching user id

**Parameters:** (Integer) user\_id

Projection (Default =ServicematchDetail)

**Headers:** Authentication (bearer {{token}})

**POST** {{base\_url}}/createinterest

**Returns:** (Servicematch) Creates and returns a new service match object

**Body:** (Integer) user\_id

(Integer[]) interest

**Headers:** Authentication (bearer {{token}})

Content-Type (Default = application/json)



twoPLUGS

## POST {{base\_url}}/deleteinterest

<b>Returns:</b>	(Result) Whether the interest was deleted successfully or if any issues were encountered	
<b>Body:</b>	(Integer) user_id	
	(Integer[]) interest	
<b>Headers:</b>	Authentication	(bearer {{token}})
	Content-Type	(Default = application/json)



twoPLUGS

# NotificationRelated Calls

**GET**      {{base\_url}}/notifications

**Returns:**                    (Notification) The notification record with a matching id

**Parameters:**            Projection                    (Default =NotificationDetail)

**Headers:**                Authentication                (bearer {{token}})

**GET**

    {{base\_url}}/notifications/search/findByFromuser\_IdOrTous

**Returns:**                    (Notification) The notification record with a matching sender or receiver

**Parameters:**            (Integer) fid

                              (Integer) tid

                              (Integer) size

                              Projection                    (Default =NotificationDetail)

**Headers:**                Authentication                (bearer {{token}})

**POST**      {{base\_url\_unauth}}/pushNotification

**Returns:**                    (String) The profile record with the specified id

**Body:**                    (String) recipient

                              (String) message



twoPLUGS

(String) title

**PATCH**    {{base\_url}}/notifications/2

**Returns:**                    (Notification) Updates the notification record and returns the updated record

**Body:**                        <ANY MEMBER VARIABLE OF NOTIFICATION>

**Headers:**                    Authorization                (bearer {{token}})  
                                  Content-Type                (Default = application/json)





## Transaction Related Calls

**GET** `{{base_url}}/transactions/{id}`

**Returns:** (Transaction) The transaction record with the specified id

**Headers:** Authentication (bearer {{token}})

```
GET {{base_url}}/transactions/search/retrieveTransactionbyUser
```

**Returns:** `(Transaction[])` The Transaction objects with a matching buyer or seller id

**Parameters:** (Integer) bid (Buyer ID)

(Integer) sid (Seller ID)

(Integer) steps

(Integer) size

Projection (Default =TransactionDetail)

**Headers:** Authentication (bearer {{token}})



twoPLUGS

**POST**      {{base\_url}}/transactions/buy

**Returns:**                      (Result) Weather or not the purchase was completed successfully

**Body:**                      (Integer) service\_id  
                                  (Integer) need\_id  
                                  (Integer) user\_id  
                                  (String) message  
                                  (Time) starttime            (ex. 2019-01-01 12:00:00)  
                                  (Time) endtime              (ex. 2019-01-01 12:00:00)

**POST**      {{base\_url}}/transactions/bid

**Headers:**                      Authentication            (bearer {{token}})

**Returns:**                      (Result) Weather or not the bid was created successfully

**Body:**                      (Integer) service\_id  
                                  (Integer) price  
                                  (Integer) user\_id  
                                  (Integer) need\_id  
                                  (Integer) refund  
                                  (Integer) refundvalid  
                                  (String) message  
                                  (Time) starttime            (ex. 2019-01-01 12:00:00)



## twoPLUGS

(Time) endtime (ex. 2019-01-01 12:00:00)

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/transactions/confirmNegotaiton

**Returns:** (Result) Weather or not the negotiation was confirmed successfully

**Body:** (Integer) transaction\_id

(Integer) service\_id

(Integer) price

(Integer) user\_id

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/transactions/makeoffer

**Returns:** (Result) Weather or not the offer was made successfully

**Body:** (Integer) transaction\_id

(Integer) need\_id

(Integer) user\_id

(Integer) price

(String) message

(Integer) service\_id

**Headers:** Authentication (bearer {{token}})



## twoPLUGS

### POST {{base\_url}}/transactions/cancelOrder

**Returns:** (Result) Weather or not the order was cancelled successfully

**Body:** (Integer) transaction\_id

(Integer) user\_id

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/transactions/processpurchasesteps

**Returns:** (Result) Weather or not the transaction was processed successfully

**Body:** (Integer) transaction\_id

(Integer) steps

(String) message

(Integer) user\_id

**Headers:** Authentication (bearer {{token}})

### POST {{base\_url}}/transactions/canceltransaction

**Returns:** (Result) Weather or not the transaction was cancelled successfully

**Body:** (Integer) transaction\_id

(Integer) user\_id

**Headers:** Authentication (bearer {{token}})



twoPLUGS

**POST**      {{base\_url}}/transactions/askRefund

**Returns:**                      (Result) Weather the request was completed correctly or if any errors were encountered

**Body:**                        (Integer) transaction\_id  
                                    (Integer) user\_id

**Headers:**                    Authentication              (bearer {{token}})

**POST**              {{base\_url}}/transactions/acceptRefund

**Returns:**                      (Result) Weather the refund was accepted correctly or if any errors were encountered

**Body:**                        (Integer) transaction\_id  
                                    (Integer) user\_id

**Headers:**                    Authentication              (bearer {{token}})

**POST**              {{base\_url}}/transactions/refuseRefund

**Returns:**                      (Result) Weather the refund was refused or if any errors were encountered

**Body:**                        (Integer) transaction\_id  
                                    (Integer) user\_id

**Headers:**                    Authentication              (bearer {{token}})



twoPLUGS

## Other Calls

**POST**      {{base\_url}}/uploadimage

**Returns:**              (Result) Weather the image was uploaded correctly or if any errors were encountered

**Body:**                (File[]) uploadingFiles

                         (Integer) type

                         (Integer) id

**Headers:**            Authentication            (bearer {{token}})



twoPLUGS

**POST**    {{base\_url}}/removeimage

**Returns:**            (Result) Whether the image was removed correctly or if any errors were encountered

**Body:**              (File[]) filename

                         (Integer) type

                         (Integer) id

**Headers:**            Authentication            (bearer {{token}})