TEXTGRID

Textgrid Breeze API

General Note

1. All Post requests must be formatted as JSON
2. The API returns responses in JSON
3. The Base API URL is https://api.textgrid.com/2010-04-01. All API calls below should be appended to the Base API URL.

API Authorization

When you sign up for an account with Textgrid, you will get an Account Sid and Auth Token.

These values should be passed in the Request Header “Authorization”, for every API call, as a Bearer token - {AccountSid}:{AuthToken} encoded in Base64

So an account with a sid “ABCD12345” and AuthToken “1234567890” would send:
Bearer QUJDRDEyMzQ1OjEyMzQ1Njc4OTA=

Example code (c#):
request.Headers.Add(“Authorization”, “Bearer QUJDRDEyMzQ1OjEyMzQ1Njc4OTA=”);

Account Functions

This section describes the methods pertaining to Account management

Get Account Details

GET /Accounts/{accountSid}.json

Path Parameters:

accountSid= "{Account Sid}"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Account successfully found:

  HttpStatus 200 OK

  Return Schema:

  ```
  {
     "sid": "{sub account sid}",
     "auth_token": "{auth token}",
     "friendly_name": "{sub account name}",
     "date_created": "{time sub account was created}",
     "date_updated": "{time sub account was updated}",
     "status": "{status of sub account}",
     "owner_account_sid": "{account sid of master account}",
     "uri":  "{URI of account details endpoint}",
     "type":  "{sub account type}"
  }
  ```

List All Sub Accounts

GET /Accounts.json

Query Parameters:

```
Page= "{Page Number [0 indexed]}"
PageSize= "{Number of records to show per page}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Accounts successfully found:

  HttpStatus 200 OK

  Return Schema:

  ```
  {
     "start": 0,
     "end": 1,
     "uri": "{URI of this page of results}",
     "first_page_uri": "{URI of first page of results}",
     "next_page_uri": "{URI of next page of results}",
  ```

```
  "previous_page_uri": "{URI of previous page of results}",
  "page": 0,
  "page_size": 2,
  "accounts": [
    {
      "status": "{Status of sub account}",
      "date_updated": "{Date Account Updated}",
      "auth_token": "{Auth Token}",
      "friendly_name": "{Sub account name}",
      "owner_account_sid": "{Account Sid of master account}",
      "uri": "{URI of specific account result}",
      "sid": "{Account Sid of sub account}",
      "date_created": "{Date Account Created}",
      "type": "{Sub Account type}"
    },
    {
      "status": "{Status of sub account}",
      "date_updated": "{Date Account Updated}",
      "auth_token": "{Auth Token}",
      "friendly_name": "{Sub account name}",
      "owner_account_sid": "{Account Sid of master account}",
      "uri": "{URI of specific account result}",
      "sid": "{Account Sid of sub account}",
      "date_created": "{Date Account Created}",
      "type": "{Sub Account type}"
    }
  ]
}
```

Create Sub Account

POST /Accounts.json

Form Parameters:

```
friendlyName = "{name of sub account}"
```

Possible responses:

- If parent account was not recognized or enabled:

HttpStatus 401 Unauthorized

- When sub account created successfully:

  HttpStatus 200 OK

  Return Schema:

```
{
   "sid": "{sub account sid}",
   "auth_token": "{auth token}",
   "friendly_name": "{sub account name}",
   "date_created": "{time sub account was created}",
   "date_updated": "{time sub account was updated}",
   "status": "active",
   "owner_account_sid": "{account sid of master account}",
   "uri":  "{URI of account details endpoint}",
   "type":  "{sub account type}",
   "webhook_secret":  "{webhook secret}"
}
```

Change Sub Account Status

POST /Accounts/{subAccountSid}.json

Path Parameters:

subAccountSid= "{Sub Account Sid}"

Form Parameters:

status = "closed/active"

Content-type:

"application/x-www-form-urlencoded"

Possible responses:

- If parent account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When sub account successfully closed:

HttpStatus 200 OK

Return Schema:

```
{
   "sid": "{sub account sid}",
   "auth_token": "{auth token}",
   "friendly_name": "{sub account name}",
   "date_created": "{time sub account was created}",
   "date_updated": "{time sub account was updated}",
   "status": "closed/active",
   "owner_account_sid": "{account sid of master account}",
   "uri":  "{URI of account details endpoint}",
   "type":  "{sub account type}",
   "webhook_secret":  "{webhook secret}"
}
```

Update Sub Account Name

POST /Accounts/{subAccountSid}.json

Path Parameters:

```
subAccountSid= "{Sub Account Sid}"
```

Form Parameters:

```
friendlyName = "{New Name of Sub Account}"
```

Content-type:

"application/x-www-form-urlencoded"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When sub account successfully renamed:

  HttpStatus 200 OK

  Return Schema:

```
{
        "sid": "{sub account sid}",
        "auth_token": "{auth token}",
        "friendly_name": "{sub account name}",
        "date_created": "{time sub account was created}",
        "date_updated": "{time sub account was updated}",
        "status": "closed",
        "owner_account_sid": "{account sid of master account}",
        "uri":  "{URI of account details endpoint}",
        "type":  "{sub account type}",
        "webhook_secret":  "{webhook secret}"
    }
```

Voice Call Functions

The following are methods that can be used to interact with voice calls

Initiate a Call

Please speak to support for the documentation for this endpoint

Incoming Calls

Please see below under “Webhooks and Callbacks” how to receive incoming calls

Retrieve Call Information

GET /Accounts/{accountSid}/Calls/{callSid}.json

Path Parameters:

```
accountSid= "{Account Sid}"
callSid= "{Call Sid}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Call successfully found:

  HttpStatus 200 OK

Return Schema:

```
{
   "sid": "{Call Sid}",
   "date_created": "Wed, 17 Nov 2021 13:17:02 +0000",
   "date_updated": "Wed, 17 Nov 2021 13:17:02 +0000",
   "account_sid": "{Account Sid}",
   "status": “in-progress”,
   "duration": 35,
   "api_version": "2010-04-01",
   "price": "-0.00000",
   "price_unit": "USD",
   "uri": "/2010-04-01/Accounts/{Account Sid}/Calls/{Call Sid}.json",
   "parent_call_sid": “{Call Sid Of Call which initiated this call, if any}”,
   "caller_name": null,
   "from": "{from Number}",
   "to": "{to Number}",
   "annotation": null,
   "answered_by": null,
   "queue_time": "Wed, 17 Nov 2021 13:17:02 +0000",
   "direction": "outbound-api",
   "start_time": "Wed, 17 Nov 2021 13:17:02 +0000",
   "from_formatted": “{from Number Formatted}”,
    "end_time": “Wed, 17 Nov 2021 13:20:58 +0000",
   "to_formatted":  “{to Number Formatted}”,
   "phone_number_sid": "{Phone Number Sid}"
}
```

Retrieve All Call Information

GET /Accounts/{accountSid}/Calls.json

Path Parameters:

accountSid= "{Account Sid}"

Query Parameters:

Page= "{Page Number [starting with 1]}"

PageSize= "{Number of records to show per page}"

To= "{Filter results by number called}"

From= "{Filter results by number called from}"

StartTime= "{Filter results by date calls started. Must be in ‘yyyy-MM-dd’ format. Use StartTime=’ to get all messages starting from specified date at 12 am, or use StartTime>=’ and ‘StartTime<=’ for calls started in the specified range of dates}"
EndTime="{Filter results by date calls ended. Must be in ‘yyyy-MM-dd’ format. Use EndTime>=’ and/or ‘EndTime<=’ for calls ended before or after 12:00 am on chosen date or in a range of dates}"
Status= "{Filter results by call status. Can be ‘queued’, ‘in-progress’, ‘failed’, ‘completed’, ‘busy’ or ‘no-answer’}"
ParentCallSid= "{Filter results by Call Sid of parent call which initiated this call}"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Calls successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
  "start": {position of first result in this response (will vary when paging)},
  "end": {position of last result in this response (will vary when paging)},
  "uri": "{URI of this page of results}",
  "first_page_uri": "{URI of first page of results}",
  "next_page_uri": "{URI of next page of results}",
  "previous_page_uri": "{URI of previous page of results}",
  "page": {current page number},
  "page_size": {amount returned with this page},
  "calls": [
    {
  "sid": "{Call Sid}",
  "date_created": "Wed, 17 Nov 2021 13:17:02 +0000",
  "date_updated": "Wed, 17 Nov 2021 13:17:02 +0000",
  "account_sid": "{Account Sid}",
  "status": “in-progress”,
  "duration": 35,
  "api_version": "2010-04-01",
  "price": "-0.00000",
  "price_unit": "USD",
  "uri": "/2010-04-01/Accounts/{Account Sid}/Calls/{Call Sid}.json",
  "parent_call_sid": “{Call Sid Of Call which initiated this call, if any}”,
  "caller_name": null,
  "from": "{from Number}",
        "to": "{to Number}",
        "annotation": null,
        "answered_by": null,
        "queue_time": "Wed, 17 Nov 2021 13:17:02 +0000",
        "direction": "outbound-api",
        "start_time": "Wed, 17 Nov 2021 13:17:02 +0000",
        "from_formatted": “{from Number Formatted}”,
         "end_time": “Wed, 17 Nov 2021 13:20:58 +0000",
        "to_formatted":  “{to Number Formatted}”,
        "phone_number_sid": "{Phone Number Sid}"
      },
      {
        "sid": "{Call Sid}",
        "date_created": "Wed, 17 Nov 2021 13:17:02 +0000",
        "date_updated": "Wed, 17 Nov 2021 13:17:02 +0000",
        "account_sid": "{Account Sid}",
        "status": “in-progress”,
        "duration": 35,
        "api_version": "2010-04-01",
        "price": "-0.00000",
        "price_unit": "USD",
        "uri": "/2010-04-01/Accounts/{Account Sid}/Calls/{Call Sid}.json",
        "parent_call_sid": “{Call Sid Of Call which initiated this call, if any}”,
        "caller_name": null,
        "from": "{from Number}",
        "to": "{to Number}",
        "annotation": null,
        "answered_by": null,
        "queue_time": "Wed, 17 Nov 2021 13:17:02 +0000",
        "direction": "outbound-api",
        "start_time": "Wed, 17 Nov 2021 13:17:02 +0000",
        "from_formatted": “{from Number Formatted}”,
         "end_time": “Wed, 17 Nov 2021 13:20:58 +0000",
        "to_formatted":  “{to Number Formatted}”,
        "phone_number_sid": "{Phone Number Sid}"
      }
    ]
}
```

Sending & Receiving Messages

The following are methods that can be used to interact with SMS messages

Send Message

POST /Accounts/{accountSid}/Messages.json

Path Parameters:

accountSid= "{Account Sid}"

Content-type:

"application/json"

Request Schema

```
{
   "body" : "{body of SMS/MMS to send}",  - required
   "from" : "{number to send SMS from, eg, +1xxxxxxxxxx}", - required
   "to" : "{number to send SMS to in the E164 format, eg, +1xxxxxxxxxx}", - required
   "statusCallback" :"{URL for receiving status callback}", - optional
   "mediaUrl" : ["{media URL for MMS}","{media URL for MMS}"] - optional
}
```

Notes:

- If there is at least 1 MediaUrl in the request, then the message sent will be MMS. Otherwise, the message will be sent as SMS
- If the To number is not in the E 164 format, the Opt Out status of the recipient may not be recognized
- The character limit for a single SMS is 2048 characters. Sending an SMS longer than that will fail
- The maximum size for an MMS media file is 525 kb
- The maximum size of all combined MMS media files together is 525 kb
- The maximum amount of media files allowed per MMS is 10
-

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- If account is not authorized to send MMS but media URL was included:

  HttpStatus 400 Bad Request

- If recipient has opted out of receiving SMS from “from number”:

  HttpStatus 400 Bad Request

- When SMS successfully sent:

  HttpStatus 200 OK

  Return Schema:

```
{
   "body": "{body of SMS sent}",
   "direction": "outbound-api",
   "from":  "+1234567890",
   "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
   "price": null,
   "error_message": null,
   "uri":”/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json”,
   "account_sid": "{account sid}",
   "num_media": {number of Media elements sent in MMS},
   "to": "+1234567890",
   "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
   "status": "{queued/sent/failed}",
   "sid":  "{message sid}",
   "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
   "messaging_service_sid": null,
   "error_code": null,
   "price_unit": "USD",
   "api_version": "2010-04-01",
   "subresource_uris": null
}
```

Send Bulk Messages

POST /Accounts/{accountSid}/BulkMessages.json

This endpoint allows a client to make 1 API call that contains multiple messages to send. This is useful for clients that need to send out a burst of messages at one. They can send them all out in 1 API call. Note, in order to facilitate bulk messages, all the messages will be initially placed in a Queue, so the status of every accepted message will be returned as “queued”. The messages will be sent out in a separate process according to the campaign limits.

Instead of returning HTTP errors, the errors, if there are, will be returned in the specific message response.

The responses returned will be in the same order as they were received.

This endpoint needs special permission to use. Please contact Textgrid Support if interested.

Path Parameters:

```
accountSid= "{Account Sid}"
```

Content-type:

```
"application/json"
```

Request Schema

```
{
    “messages” : [
                {
                   "body" : "{body of SMS/MMS to send}",  - required
                   "from" : "{number to send SMS from, eg, +1xxxxxxxxxx}", - required
                   "to" : "{number to send SMS to, eg, +1xxxxxxxxxx}", - required
                   "statusCallback" :"{URL for receiving status callback}", - optional
                   "mediaUrl" : ["{media URL for MMS}","{media URL for MMS}"] - optional
                },
                {
                   "body" : "{body of SMS/MMS to send}",  - required
                   "from" : "{number to send SMS from, eg, +1xxxxxxxxxx}", - required
                   "to" : "{number to send SMS to, eg, +1xxxxxxxxxx}", - required
                   "statusCallback" :"{URL for receiving status callback}", - optional
                   "mediaUrl" : ["{media URL for MMS}","{media URL for MMS}"] - optional
                },
                …
          ]
}
```

Notes:

- If there is at least 1 MediaUrl in the request, then the message sent will be MMS. Otherwise, the message will be sent as SMS
- The character limit for a single SMS is 2048 characters. Sending an SMS longer than that will fail
- The maximum size for an MMS media file is 525 kb

- The maximum size of all combined MMS media files together is 525 kb
- The maximum amount of media files allowed per MMS is 10

Possible responses:

- If account was not recognized or enabled, or doesn’t have the special permission for the Bulk endpoint:

  HttpStatus 401 Unauthorized

- When the messages were successfully queued:

  HttpStatus 200 OK

  Return Schema:

```
[
        {
          "body": "{body of SMS sent}",
          "direction": "outbound-api",
          "from":  "+1234567890",
          "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
          "price": null,
          "error_message": null,
          "uri":”/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json”,
          "account_sid": "{account sid}",
          "num_media": {number of Media elements sent in MMS},
          "to": "+1234567890",
          "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
          "status": "{queued/sent/failed}",
          "sid":  "{message sid}",
          "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
          "messaging_service_sid": null,
          "error_code": null,
          "price_unit": "USD",
          "api_version": "2010-04-01",
          "subresource_uris": null
        },
        {
          "body": "{body of SMS sent}",
          "direction": "outbound-api",
          "from":  "+1234567890",
          "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
          "price": null,
          "error_message": null,
          "uri":"/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json",
          "account_sid": "{account sid}",
          "num_media": {number of Media elements sent in MMS},
          "to": "+1234567890",
          "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
          "status": "{queued/sent/failed}",
          "sid":  "{message sid}",
          "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
          "messaging_service_sid": null,
          "error_code": null,
          "price_unit": "USD",
          "api_version": "2010-04-01",
          "subresource_uris": null
        }
        …
]
```

Retrieve Message

GET /Accounts/{accountSid}/Messages/{messageSid}.json

Path Parameters:

```
accountSid= "{Account Sid}"
messageSid= "{Message Sid}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Message successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
   "body": "{body of SMS}",
   "num_segments": "1",
   "direction": "outbound-api",
   "from":  "+1234567890",
   "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
   "price": "{Total Price of message- message cost + surcharge}",
   "surcharge": "{surcharge part of cost}",
   "error_message": null,
   "uri": "/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json",
   "account_sid": "{account sid}",
   "num_media": "{amount of media attached. If more than 0, then this message is an MMS}",
   "to": "+1234567890",
   "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
   "status": "{status of msg: queued/sent/delivered/undelivered/failed}",
   "sid":  "{SMS sid}",
   "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
   "error_code": null,
   "price_unit": "USD",
   "api_version": "2010-04-01",
   "email": {true/false},
    "carrierNetwork": "{carrier of destination number}",
    "messageClass": "{campaign class of message}"
    "subresourceuris": {
             "media":  "/2010-04-01/Accounts/{accountSid}/Messages/{messageSid}/Media.json"
      }
}
```

Get Media URLs in Message

GET /Accounts/{accountSid}/Messages/{messageSid}/Media.json

Path Parameters:

```
accountSid= "{Account Sid}"
messageSid= "{Message Sid}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Message successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
     "Medias": [
              {
                     "sid":  "{mediaSid}",
                     "date_created":  "Fri, 08 Sep 2023 09:38:17 +0000",
                "date_updated":  "Fri, 08 Sep 2023 09:38:17 +0000",
                "account_sid":  "{accountSid}",
                "parent_sid":  "{messageSid}",
                "content_type": "image/jpeg",
                "uri":  "/2010-04-01/Accounts/{accountSid}/Messages/{messageSid}/Media/{mediaSid}.json"
              }
            ],
            "end":  1,
            "num_pages":  0,
            "page":  0,
            "page_size": 0,
            "start":  0,
            "total":  1,
            "uri":  "https://api.textgrid.com/2010-04-01/Accounts/{accountSid}/Messages/{messageSid}/Media.json"
    }
```

Retrieve All Messages

GET /Accounts/{accountSid}/Messages.json

Path Parameters:

accountSid= "{Account Sid}"

Query Parameters:

Page= "{Page Number [starting with 0]}"

PageSize= "{Number of records to show per page}"

To= "{Filter results by number sent to}"

From= "{Filter results by number sent from}"

DateSent= "{Filter results by date sent. Must be in ‘yyyy-MM-dd’ format. Use ‘DateSent=’ for exact date or ‘DateSent>=’ and/or ‘DateSent<=’ for messages sent before or after 12:00 am on chosen date or in a range of dates}"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When Messages successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
   "start": {position of first result in this response (will vary when paging)},
   "end": {position of last result in this response (will vary when paging)},
   "uri": "{URI of this page of results}",
   "first_page_uri": "{URI of first page of results}",
   "next_page_uri": "{URI of next page of results}",
   "previous_page_uri": "{URI of previous page of results}",
   "page": {current page number},
   "page_size": {amount returned with this page},
   "messages": [
      {
          "body": "{body of SMS}",
          "num_segments": "1",
          "direction": "outbound-api",
          "from":  "+1234567890",
          "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
           "price": "{Total Price of message- message cost + surcharge}",
           "surcharge": "{surcharge part of cost}",
          "error_message": null,
          "uri": "/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json",
          "account_sid": "{account sid}",
          "num_media": "0",
          "to": "+1234567890",
          "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
          "status": "{status of msg: queued/sent/delivered/undelivered/failed}",
          "sid":  "{SMS sid}",
          "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
          "error_code": null,
          "price_unit": "USD",
          "api_version": "2010-04-01",
          "email": {true/false},
          "carrierNetwork": "{carrier of destination number}",
          "messageClass": "{campaign class of message}",
          "subresource_uris": {
                    "media": "/2010-04-01/Accounts/{account sid}Messages/{message sid}/Media.json"
         }
      },
      {
         "body": "{body of SMS}",
          "num_segments": "1",
          "direction": "outbound-api",
          "from":  "+1234567890",
          "date_updated":"Sun, 21 Aug 2022 15:45:15 +0000",
          "price": "{Total Price of message- message cost + surcharge}",
          "surcharge": "{surcharge part of cost}",
          "error_message": null,
          "uri": "/2010-04-01/Accounts/{account sid}/Messages/{message sid}.json",
          "account_sid": "{account sid}",
          "num_media": "0",
          "to": "+1234567890",
          "date_created": "Sun, 21 Aug 2022 15:45:15 +0000",
          "status": "{status of msg: queued/sent/delivered/undelivered/failed}",
          "sid":  "{SMS sid}",
          "date_sent": "Sun, 21 Aug 2022 15:45:15 +0000",
          "error_code": null,
          "price_unit": "USD",
          "api_version": "2010-04-01",
          "email": {true/false},
          "carrierNetwork": "{carrier of destination number}",
          "messageClass": "{campaign class of message}"
          "subresource_uris": {
                    "media": "/2010-04-01/Accounts/{account sid}Messages/{message sid}/Media.json"
         }
      }
   ]
}
```

User Opt In / Out

Opt User In/Out

You can explicitly opt a recipient in or out of messages from a specific number. This is useful if for example you may have an explicit opt in/out on your website that the end user can click.

GET /optin/{end user number}/{in/out}/{sender number}

End User number and sender number should both be formatted in the E 164 format, ie +1234567890
Sending “in” will re-opt the user in if the user had previously opted out. Sending “out” will opt the user out

Response code: 200 OK

Return Schema:

```
{
   "action": “[optedin/optedout/nothing]”
```

```
    }
```

The “action” in the response specifies the action that was taken for this API call. It can be:

- optedin (if the previous status was not Opted In)
- optedout (if the previous status was not Opted Out)
- nothing (if the requested action is the same as the previous status)

Multiple Opt User In/Out

You can explicitly opt a list of recipients in or out of messages from a specific number.

POST /optin/{in/out}/{sender number}/{notifyRecipient}

Content-type:

    "application/json"

Request Schema

```
[
  "number1",
  "number2",
  "number3"
]
```

End User numbers and sender number should both be formatted in the E 164 format, ie +1234567890
Sending “in” will re-opt the in if the user had previously opted out. Sending “out” will opt the user out

Send “true” for the NotifyRecipient parameter to send the end users a notification that they were opted in or out, or “false” to not send a notification

Response code: 200 OK

```
    Return Schema:
    {
      "action": [
              “[optedin/optedout/nothing]”,
              “[optedin/optedout/nothing]”,
              “[optedin/optedout/nothing]”
    }
```

Each value in the “action” array in the response corresponds to the recipient number located at the identical index in the request.

The “action” in the response specifies the action that was taken for each recipient number. It can be:

- optedin (if the previous status was not Opted In)

- optedout (if the previous status was not Opted Out)
- nothing (if the requested action is the same as the previous status)

User Opt In/Out Status

You can receive the last opt in/out a user has done from a specific number.

GET optin/status/{end user number}/{sender number}/{method}

End User number and sender number should both be formatted in the E 164 format, ie +1234567890

“Method” is an optional parameter:

- “phone” will return the last time this number opted in or out through texting START or STOP. API opt ins or outs will be ignored
- “api” will return the last time this number was opted in or out through the API. Opt ins/outs through texting START or STOP will be ignored.
- Anything else, or leaving it out altogether, will return the last status, whether through API or through text.

Response code: 200 OK

Return Schema:

```
{
   "action": “[optedin/optedout/nothing]”
   "date": “[date of last action if not “nothing”]”
   "varb": “[trigger of last action if not “nothing”. eg, Start, Stop or API]”
}
```

End-user Opt Outs

We strongly recommend that you append the following language to the end of each message sent out: “Reply STOP to unsubscribe. Msg&Data Rates May Apply”

Opt-out

The end user can opt-out of receiving text messages by texting back the words “STOP”, “QUIT”, “END” or “UNSUBSCRIBE”. Note, these words must be the sole content of the message. If any other words are present then the opt-out request will not be recognized. It is the responsibility of the client to honor the opt-out request and to no longer send messages to that recipient.

When Textgrid receives an Opt-out request, an automatic opt-out message will be sent to the user: “You have successfully been unsubscribed. You will not receive any more messages from this number. Reply

START to resubscribe”. If you would like a custom message to be sent, please be in touch with support@textgrid.com to request it.

Opt-in

The end user can opt back in using the keywords “START”, “YES” and “UNSTOP”. When Textgrid receives that opt-in message, the end user will receive an automated message “You have successfully been re-subscribed to messages from this number. Reply HELP for help. Reply STOP to unsubscribe”. If you would like a custom message to be sent, please be in touch with support@textgrid.com to request it.

Help

When Textgrid receives the “HELP” message, the end user will receive an automated message “Reply STOP to unsubscribe. Msg&Data Rates May Apply”. If you would like a custom message to be sent, please be in touch with support@textgrid.com to request it.

Phone Numbers Functions

This section describes the methods pertaining to Phone Numbers management

List Available Local Phone Numbers

GET /Accounts/{accountSid}/AvailablePhoneNumbers/{isoCountryCode}/Local.json?{query parameter}

Path Parameters:

accountSid= "{Account Sid}"

isoCountryCode= "{Two-character ISO country code (US or CA)}"

Query Parameters (please use at least 1 parameter to get numbers):

inPostalCode = "{Limit results to a particular postal code}"

inRegion = "{Limit results to a particular region (i.e. State/Province}"

areaCode = "{Find phone numbers in the specified Area Code}"

contains = "{A 3 to 7 alphanumeric character long string for pattern matching of phone numbers}"

inLata = "{Limit results to a specific Local access and transport area (LATA)}"A

inRateCenter = "{Limit results to a specific rate center}”

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When phone numbers successfully found:

HttpStatus 200 OK

Return Schema:

```
{
   "available_phone_numbers": [
      {
         "friendly_name": "{A nicely-formatted version of the phone number}",
         "phone_number": ""{The phone number, in E.164 (i.e. '+1') format}",
         "lata": "{The LATA of this phone number}",
         "rate_center": "{The rate center of this phone number}",
         "region": "{The two-letter state or province abbreviation of this phone number}",
         "iso_country": "{The ISO country code of this phone number}",
         "beta": false,
         "capabilities": {
            "voice": true,
            "SMS": true,
            "MMS": true
         }
      },
      {
         "friendly_name": "{A nicely-formatted version of the phone number}",
         "phone_number": ""{The phone number, in E.164 (i.e. '+1') format}",
         "lata": "{The LATA of this phone number}",
         "rate_center": "{The rate center of this phone number}",
         "region": "{The two-letter state or province abbreviation of this phone number}",
         "iso_country": "{The ISO country code of this phone number}",
         "beta": false,
         "capabilities": {
            "voice": true,
            "SMS": true,
            "MMS": true
         }
      }
   ],
   "uri": "{URI of this request}"
}
```

Add Incoming Phone Number

POST /Accounts/{accountSid}/IncomingPhoneNumbers.json

Note, the phone number to purchase must have been returned by a call to the List Available Local Phone Numbers endpoint.

Please note, a credit card payment method must be set up in the Dashboard before you will be able to purchase a phone number

Path Parameters:

accountSid= "{Account Sid}"

Form Parameters:

phoneNumber = "{The phone number you want to purchase. The number should be formatted starting with a '+' followed by the country code and the number in E.164 format}"   - required

friendlyName = "{A human readable description of the new incoming phone number}" - optional

voiceUrl = "{The URL that TextGrid should request when somebody calls the new phone number}" - optional

voiceMethod = "{The HTTP method that should be used to request the VoiceUrl. GET/POST. Defaults to POST}" - optional

voiceFallbackUr = "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}" - optional

voiceFallbackMethod = "{The HTTP method that should be used to request the VoiceFallbackUrl. GET/POST. Defaults to POST.}" - optional

smsUrl = "{The URL that TextGrid should request when somebody sends an SMS to the phone number}" - optional

smsMethod = "{The HTTP method that should be used to request the SmsUrl. GET/POST. Defaults to POST.}" - optional

smsFallbackUrl = "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}" - optional

smsFallbackMethod = "{The HTTP method that should be used to request the SmsFallbackUrl. GET/POST. Defaults to POST.}" - optional

statusCallback = "{The URL that TextGrid will request to pass status parameters (such as call ended or SMS status updated) to your application}" - optional

statusCallbackMethod = "{The HTTP method TextGrid will use to make requests to the StatusCallback URL. GET/POST Defaults to POST.}" - optional

Content-type:

"application/x-www-form-urlencoded"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- If phone number is missing:

HttpStatus 400 Bad Request

- When phone number successfully purchased:

  HttpStatus 200 OK

  Return Schema:

```
 {
          "sid": "{Phone Number Sid}",
          "account_sid": "{Account Sid}",
          "friendly_name": "{Human readable description of the new incoming phone number}",
          "phone_number": "{The phone number you have purchased}",
          "voice_url": "{The URL that TextGrid should request when somebody calls the new phone number}",
          "voice_method": "{The HTTP method that should be used to request the VoiceUrl}",
          "voice_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}",
          "voice_fallback_method": "{The HTTP method that should be used to request the VoiceFallbackUrl.}",
           "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
          "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
          "sms_url": "{The URL that TextGrid should request when somebody sends an SMS to the phone number: https://your.website/url}",
          "sms_method": "{The HTTP method that should be used to request the SmsUrl: POST/GET}",
          "sms_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}",
          "sms_fallback_method": "{The HTTP method that should be used to request the SmsFallbackUrl.}",
          "beta": false,
          "capabilities": {
             "voice": true,
             "sms": true,
             "mms": true
          },
          "status_callback": "{The URL that TextGrid will request to pass status parameters to your application}",
          "status_callback_method": "{The HTTP method TextGrid will use to make requests to the StatusCallback URL}",
          "api_version": "2010-04-01",
          "uri": "{URI of this specific phone number result}",
          "status": "{Status of Phone Number}"
       }
```

Update Incoming Phone Number

POST /Accounts/{accountSid}/IncomingPhoneNumbers/{phoneNumberSid}.json

Path Parameters:

accountSid= "{Account Sid}"

phoneNumberSid= "{Phone Number Sid}"

Form Parameters:

friendlyName = "{A human readable description of the incoming phone number}" - optional

voiceUrl = "{The URL that TextGrid should request when somebody calls the phone number}" - optional

voiceMethod = "{The HTTP method that should be used to request the VoiceUrl. Must be either GET or POST. Defaults to POST}" - optional

voiceFallbackUr = "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}" - optional

voiceFallbackMethod = "{The HTTP method that should be used to request the VoiceFallbackUrl. Either GET or POST. Defaults to POST.}" - optional

smsUrl = "{The URL that TextGrid should request when somebody sends an SMS to the phone number}" - optional

smsMethod = "{The HTTP method that should be used to request the SmsUrl. Must be either GET or POST. Defaults to POST.}" - optional

smsFallbackUrl = "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}" - optional

smsFallbackMethod = "{The HTTP method that should be used to request the SmsFallbackUrl. Must be either GET or POST. Defaults to POST.}" - optional

statusCallback = "{The URL that TextGrid will request to pass status parameters (such as call ended or SMS status updated) to your application}" - optional

statusCallbackMethod = "{The HTTP method TextGrid will use to make requests to the StatusCallback URL. Either GET or POST. Defaults to POST.}" - optional

Content-type:

"application/x-www-form-urlencoded"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When phone number successfully updated:

  HttpStatus 200 OK

  Return Schema:

```
{
        "sid": "{Phone Number Sid}",
        "account_sid": "{Account Sid}",
        "friendly_name": "{Human readable description of the new incoming phone number}",
        "phone_number": "{The phone number you have purchased}",
        "voice_url": "{The URL that TextGrid should request when somebody dials the new phone number}",
        ,
        "voice_method": "{The HTTP method that should be used to request the VoiceUrl}",
        "voice_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}",
        "voice_fallback_method": "{The HTTP method that should be used to request the VoiceFallbackUrl.}",
 "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
        "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
        "sms_url": "{The URL that TextGrid should request when somebody sends an SMS to the phone number: https://your.website/url}",
        "sms_method": "{The HTTP method that should be used to request the SmsUrl: POST/GET}",
        "sms_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}",
        "sms_fallback_method": "{The HTTP method that should be used to request the SmsFallbackUrl.}",
        "beta": false,
        "capabilities": {
          "voice": true,
          "sms": true,
          "mms": true
        },
        "status_callback": "{The URL that TextGrid will request to pass status parameters to your application}",
        "status_callback_method": "{The HTTP method TextGrid will use to make requests to the StatusCallback URL}",
        "api_version": "2010-04-01",
        "uri": "{URI of this specific phone number result}",
        "status": "{Status of Phone Number}"
      }
```

Delete Incoming Phone Number

DELETE /Accounts/{accountSid}/IncomingPhoneNumbers/{phoneNumberSid}.json

Path Parameters:

accountSid= "{Account Sid}"

phoneNumberSid= "{Phone Number Sid}"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When phone number successfully deleted:

  HttpStatus 204 No Content

Get Incoming Phone Number

GET /Accounts/{accountSid}/IncomingPhoneNumbers/{phoneNumberSid}.json

Path Parameters:

```
accountSid= "{Account Sid}"
phoneNumberSid= "{Phone Number Sid}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When phone number successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
        "sid": "{Phone Number Sid}",
        "account_sid": "{Account Sid}",
        "friendly_name": "{Human readable description of the incoming phone number}",
        "phone_number": "{The phone number you have purchased}",
        "voice_url": "{The URL that TextGrid should request when somebody calls the phone number}",
        "voice_method": "{The HTTP method that should be used to request the VoiceUrl}",
        "voice_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}",
        "voice_fallback_method": "{The HTTP method that should be used to request the VoiceFallbackUrl.}",
        "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
        "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
        "sms_url": "{The URL that TextGrid should request when somebody sends an SMS to the phone number: https://your.website/url}",
        "sms_method": "{The HTTP method that should be used to request the SmsUrl: POST/GET}",
        "sms_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}",
        "sms_fallback_method": "{The HTTP method that should be used to request the SmsFallbackUrl.}",
        "beta": false,
        "capabilities": {
          "voice": true,
          "sms": true,
          "mms": true
        },
        "status_callback": "{The URL that TextGrid will request to pass status parameters to your application}",
        "status_callback_method": "{The HTTP method TextGrid will use to make requests to the StatusCallback URL}",
        "api_version": "2010-04-01",
        "uri": "{URI of this specific phone number result}",
        "status": "{Status of Phone Number"
      }
```

List Incoming Phone Numbers

GET /Accounts/{accountSid}/IncomingPhoneNumbers.json

Path Parameters:

```
accountSid= "{Account Sid}"
```

Query Parameters:

```
Page= "{Page Number [0 indexed]}"
PageSize= "{Number of records to show per page}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- When phone numbers successfully found:

  HttpStatus 200 OK

  Return Schema:

```
{
  "start": {position of first result in this response (will vary when paging)},
  "end": {position of last result in this response (will vary when paging)},
  "uri": "{URI of this page of results}",
  "first_page_uri": "{URI of first page of results}",
  "next_page_uri": "{URI of next page of results}",
  "previous_page_uri": "{URI of previous page of results}",
  "page": {current page number},
  "page_size": {amount returned with this page},
  "incoming_phone_numbers": [
    {
      "sid": "{Phone Number Sid}",
      "account_sid": "{Account Sid}",
      "friendly_name": "{Human readable description of the incoming phone number}",
      "phone_number": "{The phone number you have purchased}",
      "voice_url": "{The URL that TextGrid should request when somebody calls the phone number}",
      "voice_method": "{The HTTP method that should be used to request the VoiceUrl}",
      "voice_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}",
      "voice_fallback_method": "{The HTTP method that should be used to request the VoiceFallbackUrl.}",
      "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
      "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
      "sms_url": "{The URL that TextGrid should request when somebody sends an SMS to the phone number: https://your.website/url}",
      "sms_method": "{The HTTP method that should be used to request the SmsUrl: POST/GET}",
      "sms_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}",
      "sms_fallback_method": "{The HTTP method that should be used to request the SmsFallbackUrl.}",
      "beta": false,
      "capabilities": {
          "voice": true/false,
          "outgoing_voice_only": true/false,
          "sms": true/false,
          "outgoing_sms_only": true/false,
          "mms": true/false
       },
      "status_callback": "{The URL that TextGrid will request to pass status parameters to your application}",
      "status_callback_method": "{The HTTP method TextGrid will use to make requests to the StatusCallback URL}",
      "api_version": "2010-04-01",
      "uri": "{URI of this specific phone number result}",
      "status": "{Status of Phone Number}"
    },
    {
      "sid": "{Phone Number Sid}",
      "account_sid": "{Account Sid}",
      "friendly_name": "{Human readable description of the incoming phone number}",
      "phone_number": "{The phone number you have purchased}",
      "voice_url": "{The URL that TextGrid should request when somebody calls the phone number}",
      "voice_method": "{The HTTP method that should be used to request the VoiceUrl}",
      "voice_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML at Url.}",
      "voice_fallback_method": "{The HTTP method that should be used to request the VoiceFallbackUrl.}",
       "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
            "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
            "sms_url": "{The URL that TextGrid should request when somebody sends an SMS to the phone number: https://your.website/url}",
            "sms_method": "{The HTTP method that should be used to request the SmsUrl: POST/GET}",
            "sms_fallback_url": "{The URL that TextGrid will request if an error occurs requesting or executing the TwiML defined by SmsUrl}",
            "sms_fallback_method": "{The HTTP method that should be used to request the SmsFallbackUrl.}",
            "beta": false,
            "capabilities": {
                "voice": true/false,
                "outgoing_voice_only": true/false,
                "sms": true/false,
                "outgoing_sms_only": true/false,
                "mms": true/false
             },
            "status_callback": "{The URL that TextGrid will request to pass status parameters to your application}",
            "status_callback_method": "{The HTTP method TextGrid will use to make requests to the StatusCallback URL}",
            "api_version": "2010-04-01",
            "uri": "{URI of this specific phone number result}",
            "status": "{Status of Phone Number"
        },
    ]
}
```

Add Caller ID

POST /Accounts/{accountSid}/CallerID.json

This endpoint allows you to add any number to your account as a valid From number for calling.

Note, this endpoint is restricted. If you'd like access to it, please contact support@textgrid.com

Please note, a credit card payment method must be set up in the Dashboard before you will be able to provision the phone number

Path Parameters:

accountSid= "{Account Sid}"

Json Body Parameters:

phoneNumber = "{The phone number you want to add.}"   - required

friendlyName = "{A human readable description of the new phone number}" - optional

Content-type:
"application/json"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- If phone number is missing:

  HttpStatus 400 Bad Request

- When phone number successfully added:

  HttpStatus 200 OK

  Return Schema:

```
{
        "sid": "{Phone Number Sid}",
        "account_sid": "{Account Sid}",
        "friendly_name": "{Human readable description of the new Caller ID phone number}",
        "phone_number": "{The phone number}",
        "date_created": "Tue, 12 Apr 2022 10:15:04 +0000",
        "date_updated": "Tue, 12 Apr 2022 10:15:04 +0000",
         "beta": false,
        "capabilities": {
           "voice": true,
           "sms": false,
           "mms": false
        },
        "api_version": "2010-04-01",
        "uri": "{URI of this specific phone number result}",
        "status": "{Status of Phone Number}"
 }
```

Set Auto Forward

POST /Accounts/{accountSid}/IncomingPhoneNumbers/{phoneNumberSid}/forward.json

This endpoint allows you to set a forwarding number that incoming calls to the specified number will be forwarded to. Note, this Forwarded call is charged as a new outgoing call, so there will be 2 simultaneous calls - the incoming call and the Forwarded call.

Path Parameters:

accountSid= "{Account Sid}"

phoneNumberSid= "{phone number Sid}"

Json Body Parameters:

```
{
    "forwardEnabled": {true/false, to enable or disable the auto forward},
    "forwardTo": "{phone number to forward to, in E. 164 format, ie +1xxxxxxxxxx}",
    "forwardRecord" : {true/false, whether to record the forwarded call}
}
```

Content-type:

"application/json"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- If phone number is missing, or not found in the specified account:

  HttpStatus 404 Not Found

- When Forward successfully set:

  HttpStatus 200 OK

Get Auto Forward

GET /Accounts/{accountSid}/IncomingPhoneNumbers/{phoneNumberSid}/forward.json

This endpoint retrieves the Auto Forward settings for the specified phone number.

Path Parameters:

accountSid= "{Account Sid}"

phoneNumberSid= "{phone number Sid}"

Content-type:

"application/json"

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

- If phone number is missing:

  HttpStatus 400 Bad Request

- When phone number successfully added:

  HttpStatus 200 OK

  Return Schema:

```
 {
     "forwardEnabled": {true/false, whether the auto forward is enabled or disabled },
     "forwardTo": "{phone number to forward to, in E. 164 format, ie +1xxxxxxxxxx}",
     "forwardRecord" : {true/false, whether the forwarded call is recorded or not}
}
```

Email Functions

This section describes the methods pertaining to email management

Send Email

POST /Accounts/{accountSid}/email

Path Parameters:

```
        accountSid= "{Account Sid}"
```

Possible responses:

- If account was not recognized or enabled:

  HttpStatus 401 Unauthorized

Parameters can be sent either using JSON format or application/x-www-form-urlencoded.

Note, one of To, CC or BCC addresses must be set.

Form Parameters:

fromName="{The name' to be displayed as the sender of the email.}" - optional

from= "{The 'from address' to be displayed as the sender of the email. If this parameter is null then the 'from address' will be the default as set for the account}" - optional

replyto= "{The email address used when replying to an email. If this parameter is null then the 'from' address will be used}" - optional

to= "{The address to send the email to. May have multiple addresses separated by commas}" - optional

cc= "{The address to 'cc' the email to. May have multiple addresses separated by commas}" - optional

bcc= "{The address to 'bcc' the email to. May have multiple addresses separated by commas}" - optional

subject= "{Subject of the email}" - optional

body= "{Body of the email}" - required

firstName= "{First Name of the recipient. If only one 'To' address is set, then name of recipient will be added to email }" - optional

lastName= "{Last Name of the recipient. If only one 'To' address is set, then name of recipient will be added to email }" - optional

isBodyHtml= Boolean value representing whether body of Email is in HTML or not. Default is true - optional

Return Schema:

```
{
   "Success": boolean,
   "ErrorMessage": "string",
   "EmailID": "string"
}
```

Lookup API

The Lookup API provides a way to retrieve additional information about a phone number.

Request

GET https://lookups.textgrid.com/v1/PhoneNumbers/{PhoneNumber}

{PhoneNumber} is the phone number you are requesting information about. Phone numbers can be specified either in national formatting or in standard E.164 format. If providing a number in local national format, please also specify the country as an optional parameter. If no country is provided, this will default to US.

Optional Query Parameters:

CountryCode=The ISO country code of the phone number to fetch. This is used to specify the country when the phone number is provided in a national format. If no country is provided, this will default to US

Type= The type of information to return. Can be: carrier or caller-name. The default is null. Caller Name information is currently available only in the US. To retrieve both types of information, specify this parameter twice; once with carrier and once with caller-name as the value.

Response

Return Schema:

```
{
  "caller_name": {
    "caller_name": "{string}",
    "caller_type": "{string}",
    "error_code":  "{string}",
  },
  "carrier": {
    "error_code":  "{string}",
    "mobile_country_code": "{string}",
    "mobile_network_code": "{string}",
    "name": "{string}",
    "type": "{string}"
  },
  "country_code":  "{string}",
  "national_format":  "{string}",
  "phone_number":  "{string}",
  "url": "{string}"
}
```

The following properties are always returned:

country_code : The ISO country code for the phone number.

phone_number : The phone number in E.164 format, which consists of a + followed by the country code and subscriber number.

national_format : The phone number in national format.

url : The absolute URL of the resource.

Carrier Information

As detailed above, you can add the query parameter “type=carrier” to the request in order to retrieve the telecom company that provides the phone number

Note, calling this API will incur charges as detailed on the website located at https://textgrid.com/pricing/.

Example Request:

```
GET https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?type=carrier
```

Example Response:

```
{
   "caller_name": null,
   "country_code": "US",
   "phone_number": "+15108675310",
   "national_format": "(510) 867-5310",
   "carrier": {
      "mobile_country_code": "310",
      "mobile_network_code": "800",
      "name": "T-Mobile USA, Inc.",
      "type": "mobile",
      "error_code": null
   },
   "url": "https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?Type=carrier"
}
```

Properties:

mobile_country_code: The three digit mobile country code of the carrier, used with the mobile network code to identify a mobile network operator.

mobile_network_code: The two/three digit mobile network code of the carrier, used with the mobile country code to identify a mobile network operator (only returned for mobile numbers).

name: The name of the carrier; often subject to change.

type: The phone number type. Possible values are landline, mobile, or voip. See below for more information.

error_code: The error code, if any, associated with your request.

The following are the possible values for the 'Type' property.

landline: The phone number is a landline number; generally not capable of receiving SMS messages.

mobile: The phone number is a mobile number; generally capable of receiving SMS messages.

voip: An internet based phone number that may or may not be capable of receiving SMS messages. For example, Google Voice. Returned for U.S. numbers only.

null: The phone number is valid but no information was returned from our data providers

Caller Name Information

As detailed above, you can add the query parameter “type=caller-name” to the request in order to retrieve the name of the phone number’s owner.

Note, calling this API will incur charges as detailed on the website located at https://textgrid.com/pricing/.

Example Request:

```
GET https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?type=caller-name
```

Example Response:

```
{
  "caller_name": {
    "caller_name": "Delicious Cheese Cake",
    "caller_type": "CONSUMER",
    "error_code": null
  },
  "carrier": null,
  "country_code": "US",
  "national_format": "(510) 867-5310",
  "phone_number": "+15108675310",
  "url": "https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?Type=caller-name"
}
```

Properties:

caller_name: String indicating the name of the owner of the phone number. If not available, this will return null.

caller_type: String indicating whether this caller is a business or consumer. Possible values are BUSINESS, CONSUMER. If not available, this will return null.

error_code: The error code, if any, associated with your request.

Caller name lookups for US numbers are billed per lookup, even if data is not available. Requesting Caller name lookups for non-US will return null values and will not be billed.

Caller Name and Carrier Information

As detailed above, you can add both the query parameter “type=caller-name” and “type=carrier” to the request.

Note, calling this API will incur charges for **both Caller Name and Carrier Information**. Charges for each type can be seen on the website located at https://textgrid.com/pricing/.

Example Request:

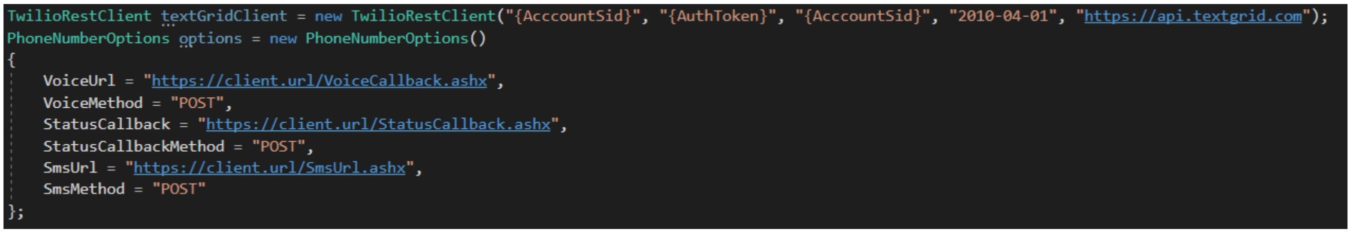
GET https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?type=caller-name&type=carrier

Example Response:

```
{
  "caller_name": {
    "caller_name": "Delicious Cheese Cake",
    "caller_type": "CONSUMER",
    "error_code": null
  },
  "carrier": {
    "mobile_country_code": "310",
    "mobile_network_code": "800",
    "name": "T-Mobile USA, Inc.",
    "type": "mobile",
    "error_code": null
  },
  "country_code": "US",
  "national_format": "(510) 867-5310",
  "phone_number": "+15108675310",
  "add_ons": null,
  "url": "https://lookups.textgrid.com/v1/PhoneNumbers/+15108675310?Type=caller-name"
}
```

Webhooks and Callbacks

Every phone number should have webhook URLs defined for dealing with incoming calls, incoming messages and status callbacks. See the section Update Incoming Phone Number for how to set them up, or follow this example for using the C# SDK:



```
TwilioRestClient textGridClient = new TwilioRestClient("{AcccountSid}", "{AuthToken}", "{AcccountSid}", "2010-04-01", "https://api.textgrid.com");
PhoneNumberOptions options = new PhoneNumberOptions()
{
    VoiceUrl = "https://client.url/VoiceCallback.ashx",
    VoiceMethod = "POST",
    StatusCallback = "https://client.url/StatusCallback.ashx",
    StatusCallbackMethod = "POST",
    SmsUrl = "https://client.url/SmsUrl.ashx",
    SmsMethod = "POST"
};
```

Webhook Sender Verification

To verify that the caller to webhooks is indeed TextGrid, all webhook calls sent out from TextGrid will have a HMAC-SHA1 signed "X-TextGrid-Signature" authentication header, signed using the Webhook Secret found on the Dashboard of every account/sub account

Steps to verify the signed signature:

1. Read the input stream into a string.
2. Build string to be signed, stringToSign = webhook URL + Request Body.
3. AuthKey = subscriber's TextGrid account Webhook Secret
4. Apply HMAC-SHA1 algorithm on the stringToSign with authKey and get the signedString.
5. Compare the X-TextGrid-Signature value with the signedString to determine the authenticity of the webhook notification.

Example (C#):

```
public void ProcessRequest(HttpContext context)
{
    string requestBody = new System.IO.StreamReader(context.Request.InputStream).ReadToEnd();
    string stringToSign = _webhookUrl + requestBody;
    string signedString = CalculateSignature(stringToSign, _secretKey);
    if (signedString != context.Request.Headers["X-TextGrid-Signature"].ToString())
        return;

    continue to process the Request....
}

private static string CalculateSignature(string payload, string appSecret)
{
    payload = EncodeNonAsciiCharacters(payload);

    byte[] secretKey = System.Text.Encoding.UTF8.GetBytes(appSecret);
    var hmac = new System.Security.Cryptography.HMACSHA1(secretKey);
    hmac.Initialize();
    byte[] bytes = System.Text.Encoding.UTF8.GetBytes(payload);
    byte[] rawHmac = hmac.ComputeHash(bytes);

    return Convert.ToBase64String(rawHmac);
}

private static string EncodeNonAsciiCharacters(string value)
{
    var sb = new System.Text.StringBuilder();
    foreach (char c in value)
    {
        if (c > 127)
        {
            string encodedValue = "\\u" + ((int)c).ToString("x4");
            sb.Append(encodedValue);
        }
        else
        {
            sb.Append(c);
        }
    }
    return sb.ToString();
}
```

Incoming Message

The URL that TextGrid will call when somebody sends an SMS/MMS to the phone number. This url is set with the “smsUrl” parameter either with the Add Incoming Phone Number method or the Update Incoming Phone Number method.

Note, the information is sent through the body of the call, encoded as application/x-www-form-urlencoded

Schema:

```
AccountSid= "{Account Sid}"
MessageSid= "{Message Sid}"
Body= "{Body of message}"
From= "+1234567890"
To="+1234567890"
ApiVersion="2010-04-01"
SmsMessageSid= "{Message Sid}"
NumMedia= "{Amount of media files attached to message}"
SmsSid= "{Message Sid}"
SmsStatus= "received"
NumSegments= "{Segment count of message}"
```

If there are 1 or more media files attached to the message (ie MMS) the following parameters will also be sent:

```
MediaContentType0="{Media Type of attachment 1}"
MediaUrl0= "{Url to retrieve attachment 1}"

MediaContentType1="{Media Type of attachment 2}"
MediaUrl1= "{Url to retrieve attachment 2}"

…
```

Responding to an incoming text message

An XML response in the form of <Response></Response> or <Response /> is expected to indicate that the request was received successfully, even if no automated response is being sent back.

If you want to send an automated response to the incoming SMS/MMS, your server should respond to the request with the following XML structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
   <Message>
      <Body>{Reply text}</Body>
   </Message>
```

```
</Response>
```

Alternatively, you can leave out the <Body> element and place the text of the reply directly in the <Message> element:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
   <Message>{Reply text}</Message>
</Response>
```

Alternatively, you can set a different URL where we can find the response (which must be in the same format as described above):

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
   <Redirect>{Redirect Url}</Redirect>
</Response>
```

Incoming Voice Call

The URL that TextGrid will call when somebody calls the phone number. This url is set either with the Add Incoming Phone Number method or the Update Incoming Phone Number method.

If the method selected is POST, the parameters will be in the body as x-www-form-urlencoded parameters; if the method is GET, they will be sent as query parameters

```
CallSid= "{Call Sid}"
AccountSid= "{Account Sid}"
From = "+1234567890"
To= "+1234567890"
CallStatus = "ringing"
ApiVersion="2010-04-01"
Direction= "inbound"
```

We expect to receive TWIML from your URL, for example:

```
<Response>
   <Say voice="alice" language="en-US">Hi. Thank you for calling.</Say>
   <Gather action="https://client.url/VoiceCallback.ashx" numDigits="10" timeout="3">
      <Say voice="alice" language="en-US">Please enter your account number</Say>
   </Gather>
```

```
    <Say voice="alice" language="en-US">I'm sorry, I didn't hear the key press.</Say>
    <Redirect>https:/client.url/VoiceCallback.ashx</Redirect>
</Response>
```

OR

```
<Response>
    <Say voice="alice" language="en-US">Thank you for calling. We will connect you to an agent now</Say>
    <Dial>+14151234567</Dial>
</Response>
```

We will make the call and follow all TWIML instructions just like Twilio until the call is terminated, at which stage we will call your Status Callback with the “completed” status.

Call Status Callback

This is the URL that TextGrid will call to pass status parameters (such as call ended) to your application. This url is set either with the Add Incoming Phone Number method or the Update Incoming Phone Number method.

If the method selected is POST, the parameters will be in the body as x-www-form-urlencoded parameters; if the method is GET, they will be sent as query parameters

```
CallSid= "{Call Sid}"
AccountSid= "{Account Sid}"
From = "+1234567890"
To= "+1234567890"
CallDuration= "{Call duration in seconds: 135}"
CallStatus  = "{completed/busy/no-answer/failed}"
ApiVersion="2010-04-01"
Direction= "inbound"
```

SMS Status Callback

TextGrid will call a Status Callback Url to pass status parameters when an SMS delivery status is updated. By default, it uses the “statusCallback” specified with the Add Incoming Phone Number method or the Update Incoming Phone Number method. Alternatively, it can be overridden on a per-SMS basis by sending the “statusCallback” parameter in the Send SMS method.

The SmsStatusDetail parameter will detail any errors which occurred.

Note, only the “delivered” and “undelivered” statuses will be received here. “Sent” and “Failed” are returned with the response to the Send SMS call.

```
SmsSid= "{SMS Sid}"
MessageSid= "{SMS Sid}"
AccountSid= "{Account Sid}"
From = "+1234567890"
To= "+1234567890"
ApiVersion="2010-04-01"
MessageStatus="{delivered/undelivered}"
SmsStatus="{delivered/undelivered}"
SmsStatusDetail="{SmsStatusDetail}"
Segments="{SegmentCount}"
```