Products & Pricing

Use Cases (https://www.nexmo.com/use-cases/)

Docs (https://docs.nexmo.com/)   Support (https://help.nexmo.com/hc/en-us)

Why Nexmo (https://www.nexmo.com/why-nexmo/)

SIGN IN (https://dashboard.nexmo.com/sign-in)

Try it free (https://dashboard.nexmo.com/sign-up)

Blog

Search   **GO**

(https://www.nexmo.com/blog/2016/10/19/how-to-send-sms-messages-with-node-js-and-express-dr/)

How to Send SMS Messages with Node.js and Express (https://www.nexmo.com/blog/2016/10/19/how-to-send-sms-messages-with-node-js-and-express-dr/)

October 19, 2016 Published by <u>Tomomi Imura (https://www.nexmo.com/blog/author/tomomi/)</u>

This is the first article in a series of "Getting Started with Nexmo and Node.js" tutorials.

Categories

<u>Award (https://www.nexmo.com/</u>
<u>Carriers (https://www.nexmo.com/</u>
<u>Chat App (https://www.nexmo.com/</u>
app/)
<u>Cloud Communications (https://www.nexmo.com/</u>
communications/)
<u>Customer Support (https://www.nexmo.com/</u>
support/)
<u>Customers (https://www.nexmo.com/</u>
<u>Dashboard (https://www.nexmo.com/</u>
<u>Developer Challenge (https://www.nexmo.com/</u>

The [Nexmo SMS API (https://docs.nexmo.com/messaging/sms-api)](https://docs.nexmo.com/messaging/sms-api) allows you to send and receive a high volume of SMS anywhere in the world. Once you get your virtual phone number, you can use the API to manage outbound messages ("sending") and inbound messages ("receiving"). In this article, you will learn how to send SMS messages with Node.js and express.

All the step-by-step articles I am going to post in this *Getting Started* series are written from my experiences as a new employee at Nexmo! Whenever I try a new thing, technical or not, I tend to write down how I did it whether I succeed or fail. Working with Nexmo APIs is not an exception – I have been writing down every step I took to work with each API from scratch. Now I am posting my notes with a bunch of screenshots to share with you, so I hope you find them helpful. So let's walk through with me!

View [the source code on GitHub (https://github.com/nexmo-community/nexmo-node-quickstart/blob/master/sms/send-express.js)](https://github.com/nexmo-community/nexmo-node-quickstart/blob/master/sms/send-express.js)

Prerequisites

Before starting this tutorial, make sure you have:

- the basic understanding of JavaScript and Node.js
- [Node.js (https://nodejs.org/en/)](https://nodejs.org/en/) installed on your machine

Getting Your API Keys

To get started with the SMS API, [sign up for a Nexmo account (https://dashboard.nexmo.com/sign-up)](https://dashboard.nexmo.com/sign-up) to get your virtual number, as well as your API credentials. Once you signed up, go to the [Nexmo Dashboard (https://dashboard.nexmo.com)](https://dashboard.nexmo.com) to get your number on [Numbers (https://dashboard.nexmo.com/your-numbers)](https://dashboard.nexmo.com/your-numbers), and API key and secret on [Settings (https://dashboard.nexmo.com/settings)](https://dashboard.nexmo.com/settings) section.
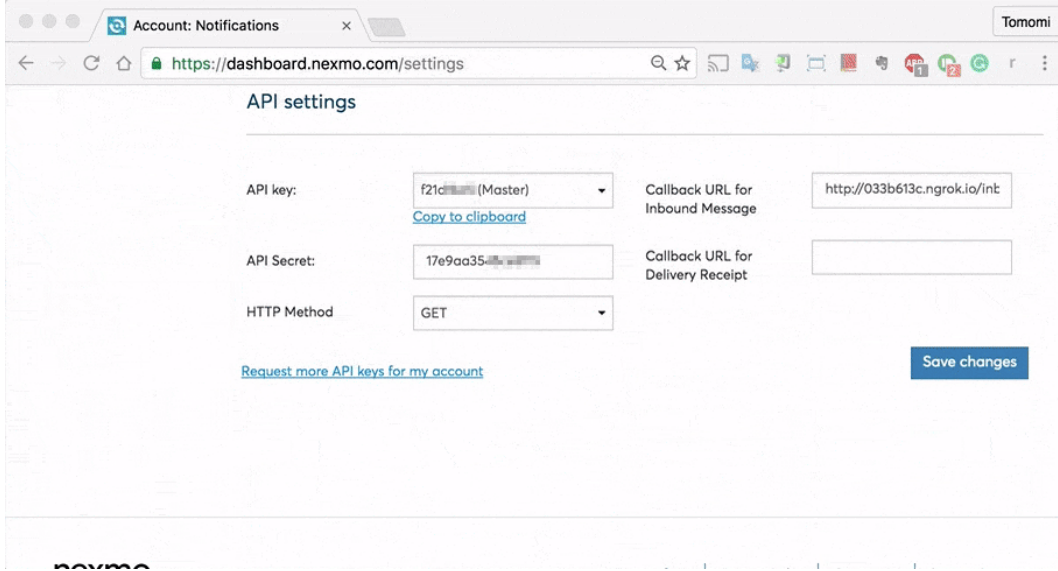
Popular

Using the Nexmo REST API Client for Node.js

First, use npm to install nexmo, the REST API client for Node.js in your working directory:

```
$ npm install nexmo --save
```

Create a `.js` file, let's call it `index.js`, and in the file, initialize a Nexmo instance with your credentials:

```
1  const Nexmo = require('nexmo');
2  const nexmo = new Nexmo({
3    apiKey: YOUR_API_KEY,
4    apiSecret: YOUR_API_SECRET
5  });
6
```

There are also optional params such as, `applicationId`, `privateKey`, and `options` object. You can find out more on the **node-nexmo** library repo on GitHub (https://github.com/Nexmo/nexmo-node).

Send SMS Messages with Node.js

To send a message, use the `nexmo.sms.sendSms` function and pass your virtual number you are sending the message from, a recipient number, and the message to

Top Contributo

Archives

be sent.

Also, you can pass optional params
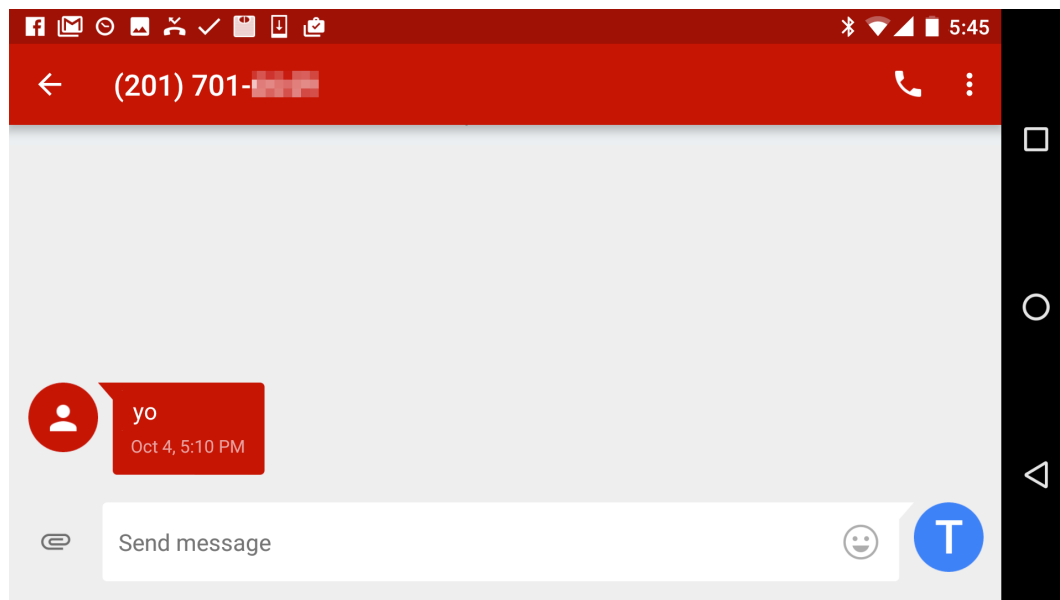(https://docs.nexmo.com/messaging/sms-api/api-
reference#request), and a callback.

Let's try hard-code the phone number (which should
start with a country code, e.g. "15105551234") and a
message for now to try the API:

```
1  nexmo.message.sendSms(
2    YOUR_VIRTUAL_NUMBER, '15105551234', 'yo',
3      (err, responseData) => {
4        if (err) {
5          console.log(err);
6        } else {
7          console.dir(responseData);
8        }
9      }
10 );
11
```

Let's run this, and see if you get a SMS to your mobile
phone.

$ node index.js

I hope it worked! You have learned how to send an SMS
message with Nexmo Node.js Library.

You can stop right here, or proceed to play with
Express.js to be able to take the queries dynamically
from POST requests!

Building a Bare Minimal SMS App with Express.js

Let's write a very simple app using Express (http://expressjs.com/) to send an SMS.

Install Express and body-parser as a dependency:

```
1  $ npm install express body-parser --save
2
```

In the **index.js**, add the following code to start a server and listens on port 3000 for connections:

```
1  const express = require('express');
2  const bodyParser = require(body-parser);
3  const app = express();
4  app.use(bodyParser.json());
5  app.use(bodyParser.urlencoded({ extended: true }));
6  const server = app.listen(3000);
7
```

Now, wrap the `nexmo.message.sendSms()` with the Express `post` route method. Let's set the `type` to `'unicode'` so you can send some emoji too! Also, print out the success response at the callback.
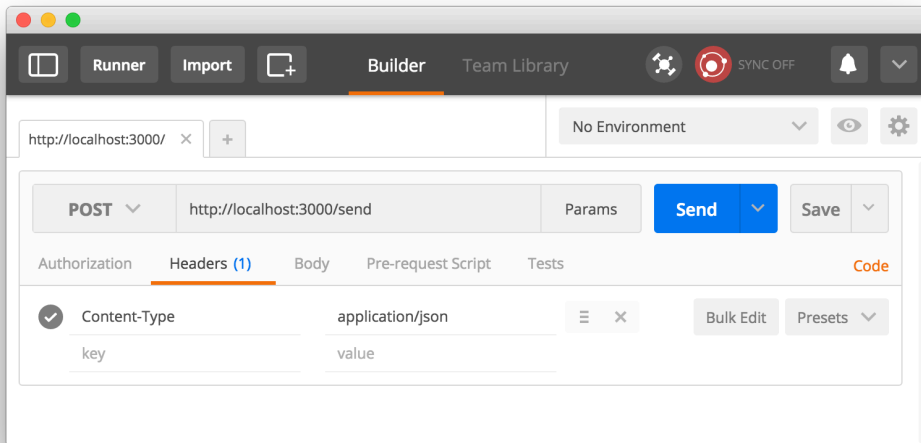
```
1  app.post('/send', (req, res) =>; {
2    // Send SMS
3    nexmo.message.sendSms(
4      config.number, req.body.toNumber, req.body.messo
5      (err, responseData) => {if (responseData) {conso
6    );
7  });
8
```

Now, try sending an SMS to any mobile phone number (including Google Voice numbers) using your app.
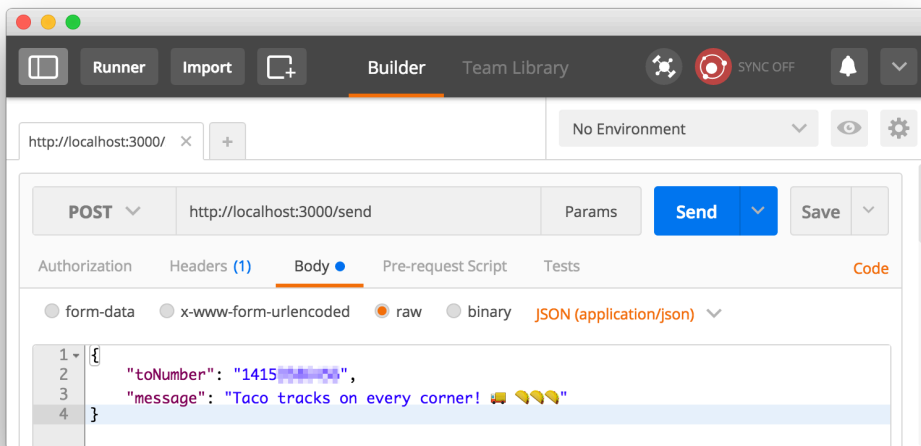
In this tutorial, we are not going to create an HTML with the form UI where a user can fill out a phone number and message (I will write a full tutorial including a front-end code sometimes!), so let's pretend we are sending data from a web interface by using Postman

(https://www.getpostman.com/) to make requests to your app. Postman is a good tool to have when you develop apps with REST APIs!

1. Launch Postman and Select **POST**, and enter *http://localhost:3000/send*.

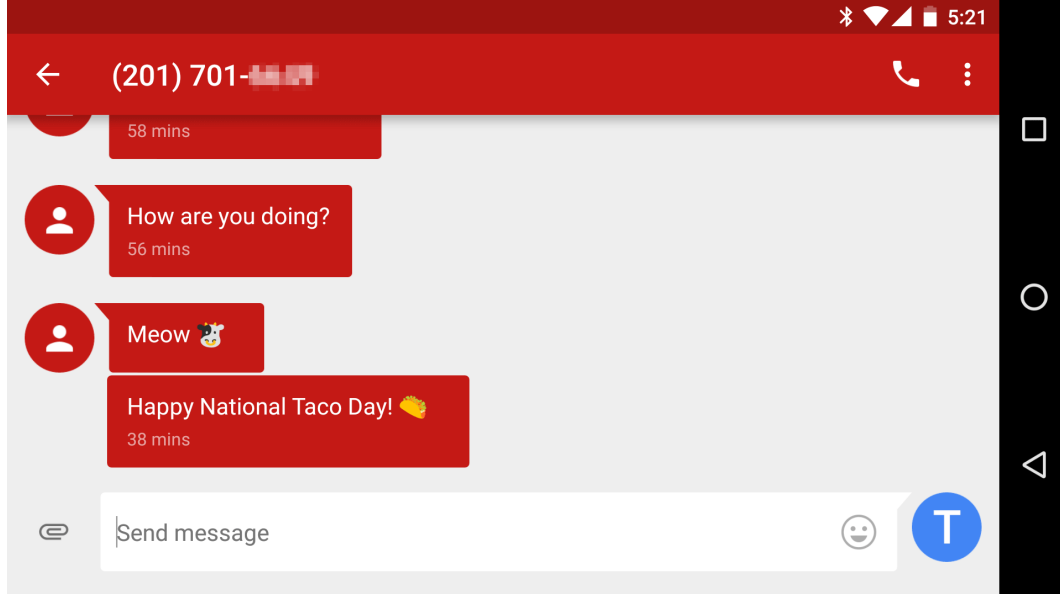2. At **Headers**, Set *Content-Type: application/json*

3. At **Body**, type a valid JSON with "toNumber" and its value (use your mobile phone number! To receive an SMS message!), also "message" and its value.

4. Press the blue **Send** button

Once you made a successfully POST to your app, you will get a text message to your phone from the virtual number!

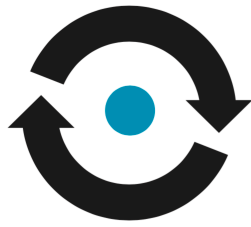Also, you will see the response printed on your terminal.

You can view the code sample (https://github.com/nexmo-community/nexmo-node-quickstart/blob/master/sms/send-express.js) used in this tutorial on GitHub.

In the next tutorial, you will learn how to receive SMS messages to your virtual number. Stay tuned!

References

- Nexmo SMS REST API
  https://docs.nexmo.com/messaging/sms-api
  (https://docs.nexmo.com/messaging/sms-api)

- Nexmo REST API client for Node.js
  https://github.com/Nexmo/nexmo-node
  (https://github.com/Nexmo/nexmo-node)

- ExpressJS (https://expressjs.com)

Tags: es6 (https://www.nexmo.com/blog/tag/es6/),
expressjs (https://www.nexmo.com/blog/tag/expressjs/),
javascript (https://www.nexmo.com/blog/tag/javascript/),
nodejs (https://www.nexmo.com/blog/tag/nodejs/), SMS
(https://www.nexmo.com/blog/tag/sms/)

Categorised in: Developers
(https://www.nexmo.com/blog/category/developers-2/),
Messaging
(https://www.nexmo.com/blog/category/messaging/),
SMS (https://www.nexmo.com/blog/category/sms/),
Tutorial
(https://www.nexmo.com/blog/category/developers-
2/tutorial/)

This post was written by Tomomi Imura
(https://www.nexmo.com/blog/author/tomomi/)

GET IN TOUCH WITH SALES

North America: **+1 855 830 8862 (tel:1-855-830-8862)**

Asia Pacific: **+852 3796 7261 (tel:852-3796-7261)**

Europe Middle East and Africa **+44 20 3198 0560 (tel:44-20-3198-0560)**

Contact Sales (https://www.nexmo.com/contact-sales/)