



Kostenloses eBook

LERNEN

Twilio

Free unaffiliated eBook created from
Stack Overflow contributors.

#twilio

Inhaltsverzeichnis

Über.....	1
Kapitel 1: Erste Schritte mit Twilio.....	2
Bemerkungen.....	2
Versionen.....	2
Examples.....	2
Senden einer SMS-Nachricht von einer US-Nummer.....	2
Senden einer SMS-Nachricht mit Versprechen.....	3
Kapitel 2: Senden Sie eine SMS-Nachricht mit Bash und cURL.....	5
Syntax.....	5
Parameter.....	5
Examples.....	5
Senden Sie eine einzelne Nachricht mit einer alphanumerischen Absender-ID.....	5
Kapitel 3: Senden Sie Nachrichten in Python.....	7
Syntax.....	7
Parameter.....	7
Bemerkungen.....	7
Examples.....	7
MMS senden.....	7
SMS senden.....	8
Senden von Nachrichten und MMS über Twilio.....	8
Kapitel 4: Senden von Nachrichten in Node.js.....	10
Einführung.....	10
Parameter.....	10
Bemerkungen.....	10
2.0 SDK-Verfall.....	10
Parameterreferenz.....	11
Examples.....	11
Senden Sie Ihre erste Nachricht.....	11
Installieren des Knotenmoduls.....	11
Senden Sie Ihre Nachricht.....	11

Kapitel 5: So empfangen Sie SMS-Nachrichten mit Ruby on Rails	12
Examples.....	12
Verwendung des Twilio Gem.....	12
Kapitel 6: So senden Sie eine SMS mit Ruby	14
Examples.....	14
Verwendung des Twilio Gem.....	14
Credits	15



You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [twilio](#)

It is an unofficial and free Twilio ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official Twilio.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Kapitel 1: Erste Schritte mit Twilio

Bemerkungen

[Twilio](#) bietet eine Plattform für die Kommunikation über IP-Nachrichten (Chat), Video, Sprache, SMS und MMS. Dieses einfache Beispiel zeigt, wie Sie eine SMS-Textnachricht mit dem [Twilio-Modul für Node.js](#) senden.

Versionen

Ausführung	Veröffentlichungsdatum
1.0.0	2016-04-12

Examples

Senden einer SMS-Nachricht von einer US-Nummer

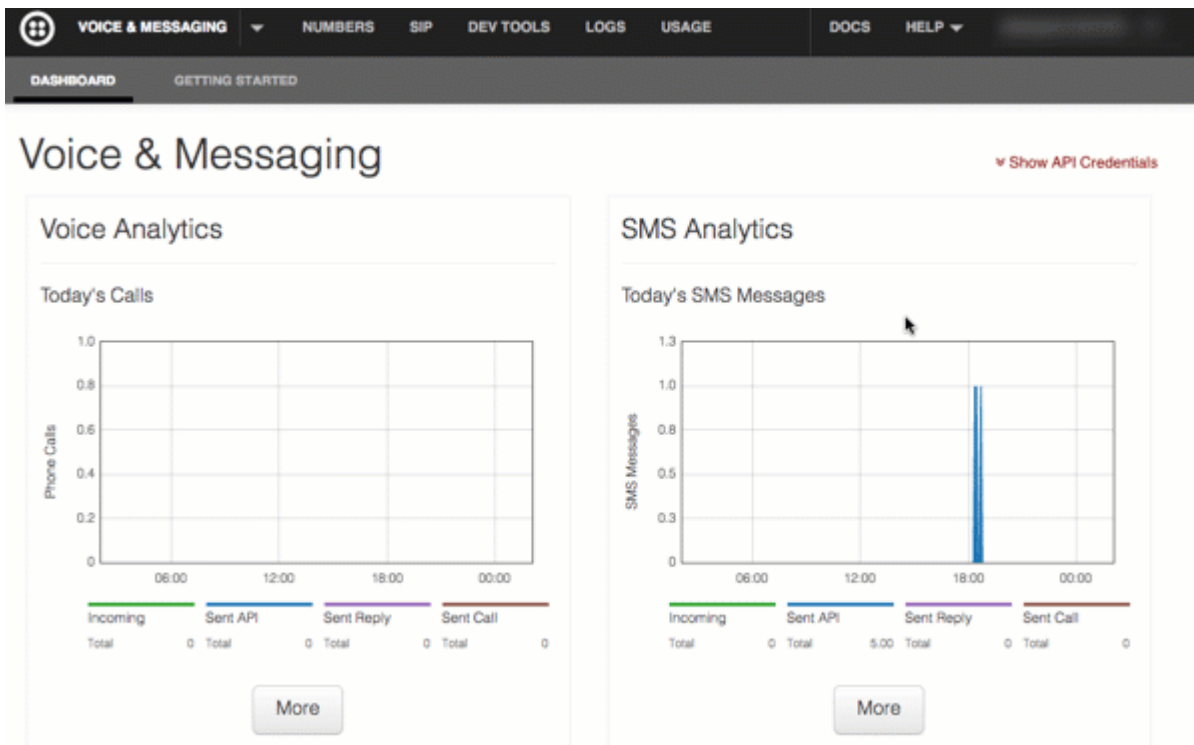
So senden Sie eine SMS-Textnachricht von einer US-amerikanischen Nummer mithilfe des Twilio Node.js SDK.

Zunächst müssen Sie den Node.js-Client mit folgendem Befehl installieren:

```
npm install twilio
```

Dann müssen Sie [ein Konto auf ihrer Website erstellen](#) .

Sobald Sie ein Konto haben, benötigen Sie die Konto-SID und das Authentifizierungs-Token, die Sie im Online-Dashboard finden können.



Ersetzen Sie im folgenden Codebeispiel `[Account SID]` und `[Auth Token]` durch die aus Ihrem Konto.

```
// Twilio Credentials
var accountSid = '[Account SID]';
var authToken = '[Auth Token]';

//require the Twilio module and create a REST client
var client = require('twilio')(accountSid, authToken);

client.messages.create({
  to: "+16518675309", // Any number Twilio can deliver to
  from: "+14158141829", // A number you bought from Twilio and can use for outbound
  communication
  body: "Hey Jenny, thanks for the pull request, will merge it right away."
}, function(err, message) {
  console.log(message.sid);
});
```

Senden einer SMS-Nachricht mit Versprechen

Twilio der Node.JS API unterstützt nativ verspricht, so dass Sie Versprechen verwenden , wenn SMS - Nachrichten zu senden (in diesem Beispiel wurde genommen und direkt aus angepasst [Twilio der API - Dokumentation](#)).

```
// Create an authenticated Twilio REST API client
var twilio = require('twilio');
var client = new twilio.RestClient('ACCOUNT_SID', 'AUTH_TOKEN');

// A simple example of sending an sms message using promises
var promise = client.makeCall({
  to: '+16515556667777', // a number to call
  from: '+16518889999', // a Twilio number you own
```

```
    body: 'Hello, world.' // A URL containing TwiML instructions for the call
  });

// You can assign functions to be called, at any time, after the request to
// Twilio has been completed. The first function is called when the request
// succeeds, the second if there was an error.
promise
  .then(function(sms) {
    console.log('Call success! SMS SID: ' + sms.sid);
  }, function(error) {
    console.error('Call failed! Reason: ' + error.message);
  });
```

Erste Schritte mit Twilio online lesen: <https://riptutorial.com/de/twilio/topic/443/erste-schritte-mit-twilio>

Kapitel 2: Senden Sie eine SMS-Nachricht mit Bash und cURL

Syntax

- <https://api.twilio.com/{API-Version}/Accounts/{AccountSid}/Messages.json> {body}

Parameter

Parameter	Einzelheiten
{API-Version}	Versionsdatum der API. zB 2010-04-01
{Account Sid}	Das Konto Sid. Startet AC
{Karosserie}	URL-kodierter Text einschließlich Von, An, Text und MessagingServiceSid

Examples

Senden Sie eine einzelne Nachricht mit einer alphanumerischen Absender-ID

Eine HTTP-POST-Anforderung wird an eine URL mit folgendem Format gesendet: "

[https://api.twilio.com/2xxx-xx-xx/Accounts/\[AccountSid\]/Messages.Json](https://api.twilio.com/2xxx-xx-xx/Accounts/[AccountSid]/Messages.Json)

Im folgenden Beispiel wird eine alphanumerische Zeichenfolge als Absender verwendet. Zum Zeitpunkt des Schreibens kann eine Absender-ID nur über eine Serviceanfrage Twilio hinzugefügt werden.

Beispielanfrage:

```
To="+447111111111" ;
From="Wxxxxxxxx" ;
MessagingServiceSid="MGxxxxxxxxxxxxxxxxxxxxxxxx" ;
Body="Test Message" ;
AccountSid="ACxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx" ;
AuthToken="[Auth Token]" ;
CMD="curl --request POST \"https://api.twilio.com/2010-04-01/Accounts/${AccountSid?}/Messages.json\" \" \" ;
CMD=\"${CMD?}\"--data-urlencode \"From=${From?}\" \" \" ;
CMD=\"${CMD?}\"--data-urlencode \"To=${To?}\" \" \" ;
CMD=\"${CMD?}\"--data-urlencode \"MessagingServiceSid=${MessagingServiceSid?}\" \" \" ;
CMD=\"${CMD?}\"--data-urlencode \"Body=${Body?}\" \" \" ;
CMD=\"${CMD?}\"-u \"${AccountSid?}:${AuthToken?}\" \" \" ;
echo \"${CMD?}\" \" ;
eval \"${CMD?}\" \" ;
```


Beispielantwort (JSON, formatiert):

```
{
  "sid": "SMxxxxxxxxxxxx",
  "date_created": "Tue, 26 Jul 2016 12:50:07 +0000",
  "date_updated": "Tue, 26 Jul 2016 12:50:07 +0000",
  "date_sent": null,
  "account_sid": "ACxxxxxxxx",
  "to": "+447111111111",
  "from": "Wxxxxxxxx",
  "messaging_service_sid": "MGxxxxxxxx",
  "body": "Test Message",
  "status": "accepted",
  "num_segments": "0",
  "num_media": "0",
  "direction": "outbound-api",
  "api_version": "2010-04-01",
  "price": null,
  "price_unit": null,
  "error_code": null,
  "error_message": null,
  "uri": "/2010-04-01/Accounts/ACxxxxxxxx/Messages/SMxxxxxxxx.json",
  "subresource_uris": {
    "media": "/2010-04-01/Accounts/ACxxxxxxxx/Messages/SMxxxxxxxx/Media.json"
  }
}
```

Senden Sie eine SMS-Nachricht mit Bash und cURL online lesen:

<https://riptutorial.com/de/twilio/topic/4237/senden-sie-eine-sms-nachricht-mit-bash-und-curl>

Kapitel 3: Senden Sie Nachrichten in Python

Syntax

- Klasse `twilio.rest.resources.Messages` (* args, ** kwargs)

Parameter

Parameter	Einzelheiten
<code>zu (str)</code>	Die Zieltelefonnummer
<code>from_ (str)</code>	Die Telefonnummer, die diese Nachricht sendet (muss eine bestätigte Twilio-Nummer sein)
<code>körper (str)</code>	Die Nachricht, die Sie senden möchten, ist auf 160 Zeichen begrenzt.
<code>status_callback</code>	Eine URL, an die Twilio POST sendet, wenn Ihre Nachricht verarbeitet wird.
<code>application_sid (str)</code>	Die 34-Zeichen-Seite der Anwendung, die Twilio für diesen Anruf verwenden soll.

Bemerkungen

Twilio sendet SMS-Nachrichten für lange Codes (in den USA eine normale zehnstellige Nummer) mit einer Geschwindigkeit von einer Nachricht pro Sekunde. Für zusätzlichen Durchsatz können Sie mit [Copilot](#) Nummernpools verwenden oder einen [Kurzcode](#) beantragen, der 30 oder mehr Nachrichten pro Sekunde sendet.

Examples

MMS senden

Sie können Ihrer Nachricht ein Bild mit dem Parameter `media_url` .

```
# Download the twilio-python library from http://twilio.com/docs/libraries
from twilio.rest import TwilioRestClient

# Find these values at https://twilio.com/user/account
account_sid = "ACXXXXXXXXXXXXXXXXXXXX"
auth_token = "YYYYYYYYYYYYYYYYYYYY"
client = TwilioRestClient(account_sid, auth_token)
message = client.messages.create(
    to="+12316851234",
    from_="TWILIO_NUMBER",
```

```
body="Hello there, StackOverflow!",
media_url=[
    'https://demo.twilio.com/owl.png',
    'https://demo.twilio.com/logo.png'])
```

SMS senden

Um Ihre erste SMS mit Twilio und Python zu senden, benötigen Sie nur die [Twilio-Python-Hilfsbibliothek](#), um loszulegen.

```
# Download the twilio-python library from http://twilio.com/docs/libraries
from twilio.rest import TwilioRestClient

# Find these values at https://twilio.com/user/account
account_sid = "ACXXXXXXXXXXXXXXXXXXXX"
auth_token = "YYYYYYYYYYYYYYYYYYYY"
client = TwilioRestClient(account_sid, auth_token)

message = client.messages.create(to="+12316851234", from_"TWILIO_NUMBER",
                                body="Hello there, StackOverflow!")
```

Senden von Nachrichten und MMS über Twilio

Twilio hilft bei der Erstellung von Apps, die mit jedem auf der Welt kommunizieren. APIs für Voice & Video, Messaging und Authentifizierung für jede Anwendung.

Sie können einen API-Schlüssel kostenlos erhalten.

Um eine Nachricht über Twilio zu senden, muss Ihre Anwendung eine HTTP-POST-Anforderung an Twilio mit den folgenden Dingen stellen:

1. Die Telefonnummer, an die Sie die Nachricht senden möchten.
2. Nummer, von der aus Sie die Nachricht senden möchten. (Nur für nachrichten aktivierte Telefonnummern von Twilio funktionieren)
3. Der Hauptteil der Nachricht

Um eine SMS zu senden, senden Sie eine HTTP-POST-Anforderung an die Nachrichtenressource.

```
POST https://api.twilio.com/20xx-xx-xx/Accounts/xxxxxxx/Messages
```

Nachfolgend finden Sie einen Beispielcode, der zeigt, wie das Senden von Nachrichten mit der Twilio-API funktioniert.

```
# Download the twilio-python library from http://twilio.com/docs/libraries
from twilio.rest import TwilioRestClient

# Find these values at https://twilio.com/user/account
account_sid = "ACXXXXXXXXXXXXXXXXXXXX"
auth_token = "YYYYYYYYYYYYYYYYYYYY"
```

```
client      = TwilioRestClient(account_sid, auth_token)

message = client.messages.create(
    to      = "+12316851234",
    from_   = "+15555555555",
    body    = "Hello there!"
)
```

Wenn die Twilio-Nummer MMS unterstützt, können Sie auch MMS senden und empfangen.

Nachfolgend finden Sie den Code zum Senden von MMS über die Twilio-API.

```
message = client.messages.create(
    to      = "+12316851234",
    from_   = "+15555555555",
    body    = "Hello there!",
    media_url=[
        'https://demo.twilio.com/owl.png',
        'https://demo.twilio.com/logo.png'
    ])
```

Senden Sie Nachrichten in Python online lesen: <https://riptutorial.com/de/twilio/topic/579/sendensie-nachrichten-in-python>

Kapitel 4: Senden von Nachrichten in Node.js

Einführung

Mit Twilio können Sie mit Node.js Textnachrichten senden und empfangen, indem Sie mit der Twilio-Node-Bibliothek Anrufe tätigen. In diesem Handbuch wird davon ausgegangen, dass Sie bereits ein Konto bei Twilio eingerichtet haben und Ihre Konto-SID und das Authentifizierungstoken von der Twilio Console haben.

Parameter

Parameter	Einzelheiten
zu	Eine gültige Telefonnummer, um die Nachricht zu erhalten
von	Eine Twilio-Nummer, die Ihnen zugewiesen wurde
Karosserie	Der Hauptteil der Textnachricht ist auf 1600 Zeichen begrenzt
Status Rückruf	Eine URL, an die Twilio veröffentlicht, wenn sich der Nachrichtenstatus ändert
maxPreis	Legen Sie den maximalen Preis einer Nachricht fest
Gültigkeitszeitraum	Die Anzahl der Sekunden, die die Nachricht in der Twilio-Warteschlange verbleibt
Rückmeldung geben	Boolescher Wert, wenn auf true gesetzt
mediaUrl	Eine URL, die einen GIF-, PNG- oder JPEG-Inhalt enthält, der mit der Nachricht gesendet wird

Bemerkungen

2.0 SDK-Verfall

Twilio verfügt über zwei Versionen des Twilio-Node-Moduls, ein 2.0 SDK und ein 3.0 SDK. Es wird empfohlen, das 3.0 SDK zu verwenden, da das 2.0 SDK am 31.08.2017 nicht mehr verwendet wird.

Hinweis zur Ablehnung: Neue Funktionen werden nur zur neuen Bibliothek hinzugefügt (Node Helper Library 3.x). Die alte Bibliothek (2.x) wird bis zum 31.08.2017 offiziell unterstützt. Nach diesem Tag stellt Twilio keine Fehlerbehebungen mehr bereit und der

Support fordert Sie möglicherweise auf, ein Upgrade durchzuführen, bevor Sie Probleme mit dem Debugging beheben. <https://www.twilio.com/docs/libraries/node>

Parameterreferenz

Eine ausführlichere Beschreibung finden Sie in den [Twilio-REST-Dokumenten](#) . Twilio hat auch ein [JS-Dokument](#) , das als Referenz verwendet werden kann.

Examples

Senden Sie Ihre erste Nachricht

Installieren des Knotenmoduls

Sie können dieses Modul installieren, indem Sie den folgenden Befehl in Ihrem Projektverzeichnis ausführen:

```
npm install twilio
```

Senden Sie Ihre Nachricht

```
const accountSID = ''; // Obtained from the Twilio Console
const authToken = ''; // Obtained from the Twilio Console

const client = require('twilio')(accountSID, authToken);

client.messages.create({
  to: '', // Number you want to send to
  from: '', // From a valid Twilio number
  body: 'Hello, from Stack Overflow', // What appears in the text message
})
.then((message) => console.log(message.sid));
```

[Senden von Nachrichten in Node.js online lesen:](#)

<https://riptutorial.com/de/twilio/topic/9955/sendenvonnachrichteninnodejs>

Kapitel 5: So empfangen Sie SMS-Nachrichten mit Ruby on Rails

Examples

Verwendung des Twilio Gem

Wenn Sie über ein Twilio-Konto und [API-Anmeldeinformationen](#) verfügen, fügen Sie der Gemfile-Datei Folgendes hinzu:

```
gem 'twilio-ruby'
```

Alternativ können Sie auch `gem install twilio-ruby`.

Damit Twilio eine eingehende SMS an eine bestimmte Route in Ihrer Anwendung senden kann, müssen Sie die Messaging-URL für Ihre [Telefonnummer](#) konfigurieren. Sobald dies erledigt ist, müssen Sie eine Route in `config/routes.rb`:

```
TwilioSample::Application.routes.draw do
  resources :messages
end
```

Dadurch werden RESTful-Routen (`GET /messages` zum Auflisten von Nachrichten, `POST /messages` zum Erstellen einer Nachricht usw.) erstellt, die Anforderungen an den `MessagesController` senden. Wenn diese Routen definiert sind, würden Sie die Messaging-URL im Twilio-Dashboard auf `https://your.site/messages` als HTTP POST-Webhook festlegen.

Ihr Controller verarbeitet die Nachricht:

```
class MessagesController < ApplicationController

  def create
    # First, check to see if this message really came from Twilio
    uri, signature = [request.protocol, request.host_with_port, request.path].join,
    request.env['HTTP_X_TWILIO_SIGNATURE']
    validator = Twilio::Util::RequestValidator.new(ENV['TWILIO_AUTH_TOKEN'])

    if validator.validate(uri, request.request_parameters, signature)
      # The params hash has the data from the Twilio payload
      Rails.logger.info "I just got this message from #{params[:From]}: #{params[:Body]}"

      # Send back a response that Twilio can handle
      render xml: { Sms: "Thanks, I got your message" }.to_xml(root: 'Response')

    else
      # This request didn't come from Twilio, so return a "Forbidden" response
      head(403)
    end
  end
end
```

```
end
```

```
end
```

So empfangen Sie SMS-Nachrichten mit Ruby on Rails online lesen:

<https://riptutorial.com/de/twilio/topic/3673/so-empfangen-sie-sms-nachrichten-mit-ruby-on-rails>

Kapitel 6: So senden Sie eine SMS mit Ruby

Examples

Verwendung des Twilio Gem

Dies setzt voraus, dass Sie ein Konto bei twilio haben und eine Telefonnummer gekauft / reserviert haben ...

Wenn Sie Bundler verwenden, fügen Sie Ihrem Gemfile die folgende Zeile hinzu, um `twilio-ruby` in Ihr Projekt aufzunehmen:

```
gem 'twilio-ruby'
```

Ansonsten geben Sie in der Kommandozeile `gem install twilio-ruby` . Möglicherweise benötigen Sie `sudo` wenn Sie das System Ruby und nicht Ruby-env usw. verwenden.

und der Code

```
# Instantiate a Twilio REST client with your account SID and auth token,  
# which can be found on your dashboard.  
  
client = Twilio::REST::Client.new("your_account_sid", "your_auth_token")  
  
client.messages.create(from: "+15556667777", to: "+15558883333", body: "Hello, world!")
```

So senden Sie eine SMS mit Ruby online lesen: <https://riptutorial.com/de/twilio/topic/1760/so-senden-sie-eine-sms-mit-ruby>

Credits

S. No	Kapitel	Contributors
1	Erste Schritte mit Twilio	Adam , cdock , Community , ddoria921 , Kevin Whinnery , msmanek , Sven
2	Senden Sie eine SMS-Nachricht mit Bash und cURL	antonycc
3	Senden Sie Nachrichten in Python	anthon , Henk Langeveld , Jacob Budin , Megan Speir , Shubham Gupta
4	Senden von Nachrichten in Node.js	Matt Shirley
5	So empfangen Sie SMS-Nachrichten mit Ruby on Rails	Benjamin Curtis
6	So senden Sie eine SMS mit Ruby	derrek , Steve Graham