



Бесплатная электронная книга

УЧУСЬ

Twilio

Free unaffiliated eBook created from
Stack Overflow contributors.

#twilio

.....	1
1: Twilio	2
.....	2
.....	2
Examples.....	2
SMS-	2
SMS-	3
2: SMS Ruby	5
Examples.....	5
Twilio Gem.....	5
3: SMS- Ruby on Rails	6
Examples.....	6
Twilio Gem.....	6
4: Python	8
.....	8
.....	8
.....	8
Examples.....	8
MMS.....	8
SMS.....	9
MMS Twilio.....	9
5: Node.js	11
.....	11
.....	11
.....	11
2.0 SDK	11
.....	12
Examples.....	12
.....	12
.....	12
.....	12

6: SMS- Bash cURL	13
.....	13
.....	13
Examples.....	13
, -	13
.....	15

Около

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

глава 1: Начало работы с Twilio

замечания

Twilio предоставляет платформу для обмена данными через IP-сообщения (чат), видео, голос, SMS и MMS. Этот простой пример демонстрирует, как отправить текстовое SMS-сообщение с [помощью модуля Twilio для Node.js](#).

Версии

Версия	Дата выхода
1.0.0	2016-04-12

Examples

Отправка SMS-сообщения из номера в США

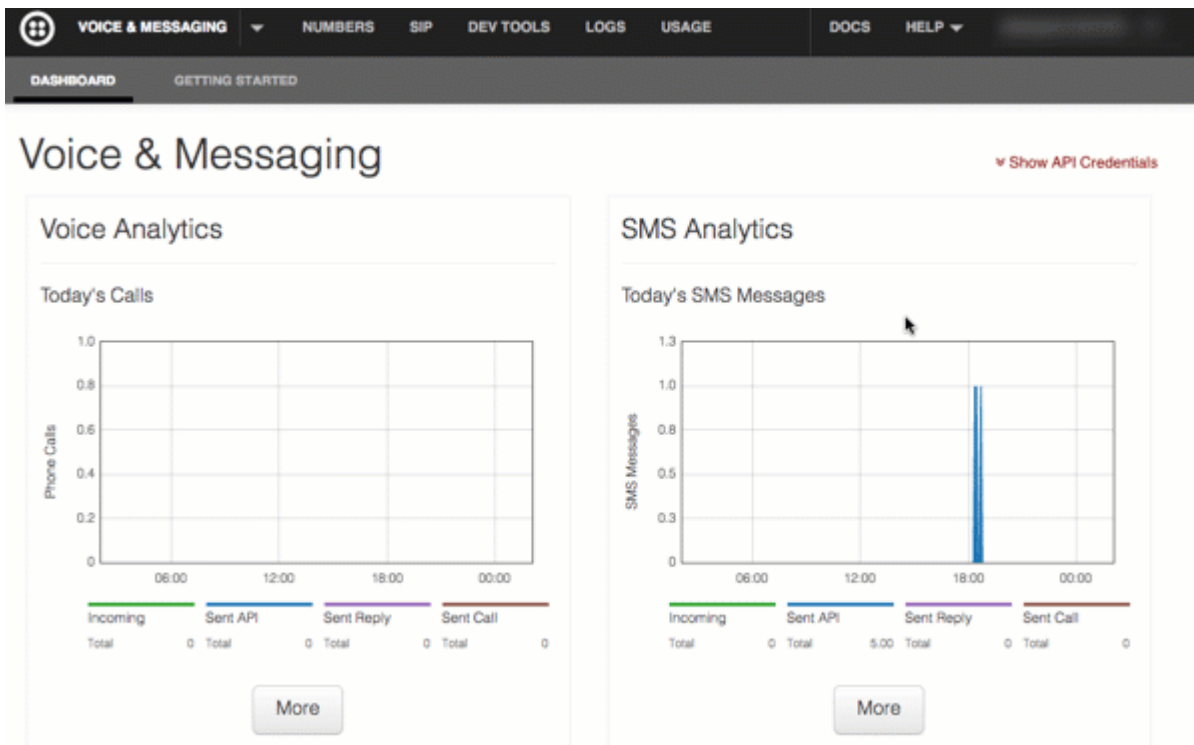
Вот как отправить текстовое SMS-сообщение из американского номера с помощью SDK Twilio's Node.js.

Сначала вам нужно установить клиент Node.js, используя:

```
npm install twilio
```

Затем вам необходимо [создать учетную запись на своем веб-сайте](#) .

После того, как у вас есть учетная запись, вам понадобится идентификатор SID учетной записи и токен аутентификации, который вы можете найти на онлайн-панели информационной панели.



В приведенном ниже примере кода замените [Account SID] и [Auth Token] на те, что указаны в вашей учетной записи.

```
// 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);
});
```

Отправка SMS-сообщения с помощью обещаний

API Node.JS от Twilio изначально поддерживает обещания, позволяя вам использовать обещания при отправке SMS-сообщений (этот пример был взят и адаптирован непосредственно из [документации API Twilio](#)).

```
// 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);
  });
```

Прочитайте Начало работы с Twilio онлайн: <https://riptutorial.com/ru/twilio/topic/443/начало-работы-с-twilio>

глава 2: Как отправить SMS с помощью Ruby

Examples

Использование Twilio Gem

Это предполагает, что у вас есть учетная запись twilio и вы купили / зарезервировали номер телефона ...

Если вы используете bundler, добавьте следующую строку в свой Gemfile, чтобы включить twilio-ruby в свой проект:

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

в противном случае введите `gem install twilio-ruby` в командной строке. Возможно, вам понадобится `sudo` если вы используете системный Ruby, а не ruby-env и т. Д.

и код

```
# 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!")
```

Прочитайте Как отправить SMS с помощью Ruby онлайн:

<https://riptutorial.com/ru/twilio/topic/1760/как-отправить-sms-с-помощью-ruby>

глава 3: Как получать SMS-сообщения с Ruby on Rails

Examples

Использование Twilio Gem

Предполагая, что у вас есть учетная запись Twilio и [учетные данные API](#), добавьте следующее в свой Gemfile:

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

В качестве альтернативы вы можете `gem install twilio-ruby`.

Чтобы Twilio отправляла входящие SMS-сообщения на конкретный маршрут в вашем приложении, вам необходимо настроить URL-адрес обмена сообщениями для вашего [номера телефона](#). Как только это будет сделано, вам нужно настроить маршрут в `config/routes.rb`:

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

Это создает набор маршрутов RESTful (`GET /messages` для отображения сообщений, `POST /messages` для создания сообщения и т. Д.), которые будут отправлять запросы в `MessagesController`. Определив эти маршруты, вы должны установить URL-адрес сообщений в панели инструментов Twilio в `https://your.site/messages` как HTTP POST Webhook.

Контроллер обрабатывает сообщение:

```
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
    end
  end
end
```

```
    head(403)
  end

end

end
```

Прочитайте [Как получать SMS-сообщения с Ruby on Rails онлайн](https://riptutorial.com/ru/twilio/topic/3673/как-получать-sms-сообщения-с-ruby-on-rails):

<https://riptutorial.com/ru/twilio/topic/3673/как-получать-sms-сообщения-с-ruby-on-rails>

глава 4: Отправить сообщения в Python

Синтаксис

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

параметры

параметр	подробности
<code>to (str)</code>	Номер телефона назначения.
<code>from_ (str)</code>	Номер телефона, отправляющий это сообщение (должен быть подтвержденным номером Twilio)
тело (str)	Сообщение, которое вы хотите отправить, ограничивается 160 символами.
<code>status_callback</code>	URL-адрес, по которому Twilio будет отправлять POST, когда ваше сообщение будет обработано.
<code>application_sid (str)</code>	Для обработки этого телефонного звонка должен использоваться 34 символа приложения Twilio.

замечания

Twilio отправит SMS-сообщения для длинных кодов (в США, десятизначное число) со скоростью одного сообщения в секунду. Для дополнительных пропускной способности, вы можете использовать пулы номеров с [Copilot](#) или применять для [шорткода](#), который может отправить 30 сообщений или более в секунду.

Examples

Отправка MMS

Вы можете добавить изображение в сообщение с помощью параметра `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'])
```

Отправка SMS

Чтобы отправить свое первое SMS-сообщение с Twilio и Python, вам просто понадобится вспомогательная библиотека [Twilio-Python](#) для начала.

```
# 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!")
```

Отправка сообщений и MMS через Twilio

Twilio помогает создавать приложения, которые общаются со всеми в мире. API для голоса и видео, сообщений и аутентификации для каждого приложения.

Вы можете получить ключ API бесплатно.

Чтобы отправить сообщение через Twilio, вашему приложению необходимо выполнить HTTP-запрос POST для Twilio со следующими вещами:

1. Номер телефона, на который вы хотите отправить сообщение.
2. Twilio, из которого вы пытаетесь отправить сообщение. (Будут работать только телефонные номера с поддержкой телефона Twilio)
3. Тело сообщения.

Чтобы отправить SMS, отправьте запрос HTTP POST на ресурс Messages.

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

Ниже приведен пример кода, показывающий, как будет работать отправка сообщений с Twilio API.

```
# 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 = "ACXXXXXXXXXXXXXXXXXX"
auth_token  = "YYYYYYYYYYYYYYYYYY"
client      = TwilioRestClient(account_sid, auth_token)

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

Если номер Twilio поддерживает MMS, вы также можете отправлять и получать MMS.

Ниже приведен код для отправки MMS через 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'
    ]
)
```

Прочитайте [Отправить сообщения в Python онлайн: https://riptutorial.com/ru/twilio/topic/579/отправить-сообщения-в-python](https://riptutorial.com/ru/twilio/topic/579/отправить-сообщения-в-python)

глава 5: Отправка сообщений в Node.js

Вступление

Twilio позволяет отправлять и получать текстовые сообщения с помощью Node.js, используя библиотеку twilio-node для совершения вызовов. В этом руководстве предполагается, что вы уже настроили учетную запись с Twilio и имеете SID учетной записи и токен аутентификации в консоли Twilio.

параметры

параметр	подробности
в	Действительный номер телефона для получения сообщения
от	Номер Twilio, присвоенный вам
тело	Тело текстового сообщения ограничено 1600 символами
StatusCallback	URL, который Twilio отправляет, когда изменяется статус сообщения
maxPrice	Установка максимальной цены сообщения
срок годности	Количество секунд, в течение которых сообщение останется в очереди Twilio
обеспечить обратную связь	Булево значение, если установлено значение true
mediaUrl	URL-адрес, содержащий gif, png или jpeg-контент, который будет отправлен с сообщением

замечания

Отказ от версии 2.0 SDK

Twilio имеет две версии модуля twilio-node, 2.0 SDK и 3.0 SDK. Рекомендуется использовать SDK 3.0, поскольку 2.0 SDK будет устаревшим на 8/31/2017.

Уведомление об изъятии: новая функциональность будет добавлена только в

новую библиотеку (Node Helper Library 3.x). Старая библиотека (2.x) будет официально поддерживаться до 31.08.2017. После этого дня Twilio перестанет исправлять ошибки, и поддержка может потребовать обновления до отладки.
<https://www.twilio.com/docs/libraries/node>

Ссылка на параметр

Более подробное описание вы можете найти в [документах Twilio REST](#) . Twilio также имеет [JSdocs](#), который можно использовать в качестве ссылки.

Examples

Отправка вашего первого сообщения

Установка модуля узла

Вы можете установить этот модуль, выполнив приведенную ниже команду в каталоге проекта:

```
npm install twilio
```

Отправка сообщения

```
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));
```

Прочитайте [Отправка сообщений в Node.js онлайн](#): <https://riptutorial.com/ru/twilio/topic/9955/отправка-сообщений-в-node-js>

глава 6: Отправьте SMS-сообщение с помощью Bash и cURL

Синтаксис

- [https://api.twilio.com/ {Api version} / Учетные записи / {AccountSid} /Messages.json](https://api.twilio.com/{Api version}/Учетные записи/{AccountSid}/Messages.json) {body}

параметры

параметр	подробности
{Аpi версия}	Дата версии API. например, 2010-04-01
{Account Sid}	Учетная запись Sid. Начинает AC
{ Тело}	Тело, закодированное URL, включая From, To, Body и MessagingServiceSid

Examples

Отправляйте одно сообщение, используя буквенно-цифровой идентификатор отправителя

Запрос HTTP POST отправляется по URL-адресу формата: « [https://api.twilio.com/2xxx-xx-xx/Accounts/\[AccountSid\]/Messages.json](https://api.twilio.com/2xxx-xx-xx/Accounts/[AccountSid]/Messages.json) »

В приведенном ниже примере используется алфавитно-цифровая строка в качестве отправителя. На момент написания идентификатор отправителя может быть добавлен только через служебный запрос Twilio.

Пример запроса:

```
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?}\" ;
```

Пример ответа (JSON, отформатирован):

```
{  
  "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"  
  }  
}
```

Прочитайте Отправьте SMS-сообщение с помощью Bash и cURL онлайн:

<https://riptutorial.com/ru/twilio/topic/4237/отправьте-sms-сообщение-с-помощью-bash-и-curl>

кредиты

S. No	Главы	Contributors
1	Начало работы с Twilio	Adam , cdock , Community , ddoria921 , Kevin Whinnery , msmanek , Sven
2	Как отправить SMS с помощью Ruby	derrek , Steve Graham
3	Как получать SMS-сообщения с Ruby on Rails	Benjamin Curtis
4	Отправить сообщения в Python	anthon , Henk Langeveld , Jacob Budin , Megan Speir , Shubham Gupta
5	Отправка сообщений в Node.js	Matt Shirley
6	Отправьте SMS-сообщение с помощью Bash и cURL	antonycc