

Connect to Twilio over SIP

1

*Using your Web Browser,
Open this URL:*

<http://mlhlocal.host/lhd-resources>

2

Click on the workshop you're attending, and find:

- Setup Instructions
- The Code Samples
- A demo project
- A Workshop FAQ
- These Workshop Slides
- More Learning Resources



***Our Mission** is to Empower Hackers.*

65,000+
HACKERS

12,000+
PROJECTS CREATED

3,000+
SCHOOLS

We hope you learn something awesome today!
Find more resources: <http://mlh.io/>

What will you **learn** today?

- 1 How to configure a Twilio phone number
- 2 How to set up automatic SMS responses with TwiML
- 3 How to create a Twilio App!

Table of Contents

-  **1. What's Twilio?**
- 2. MLH Memes App**
- 3. Twilio Console**
- 4. Test your App**
- 5. Next Steps**

Who has heard of Twilio?



What does Twilio do?

It's a **cloud communication** Platform as a Service (PaaS). Twilio can automate sending and receiving:

- SMS (texts)
- Phone calls
- Video
- ... And more!

Key Term

Platform as a Service (PaaS): A platform for developers to build internet applications without having to build and maintain the underlying infrastructure (e.g. servers) themselves.

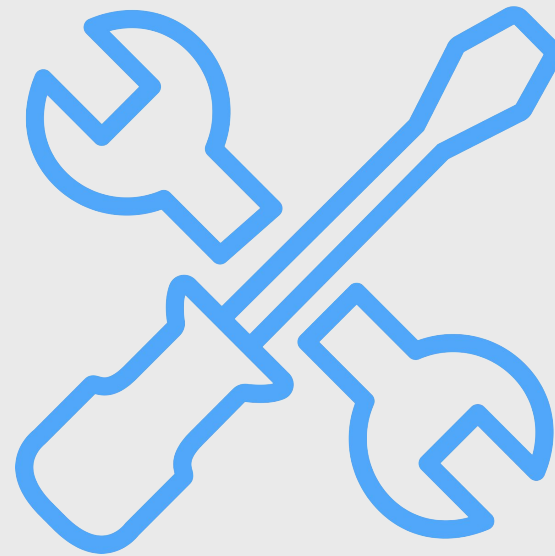
Table of Contents

1. What's Twilio?
-  2. MLH Memes App
3. Twilio Console
4. Test your App
5. Next Steps

What are we going to build with Twilio?

We're going to build an MLH Memes App with these tools:

- NodeJS
- Glitch
- The Twilio SMS API



Cool, let's try the app!

Text any word to this number:

+1 (312) 967 9068

What happens?



Woah, what happened?

It kind of feels like magic, right?

You text a number, and you get back
a meme or a gif?

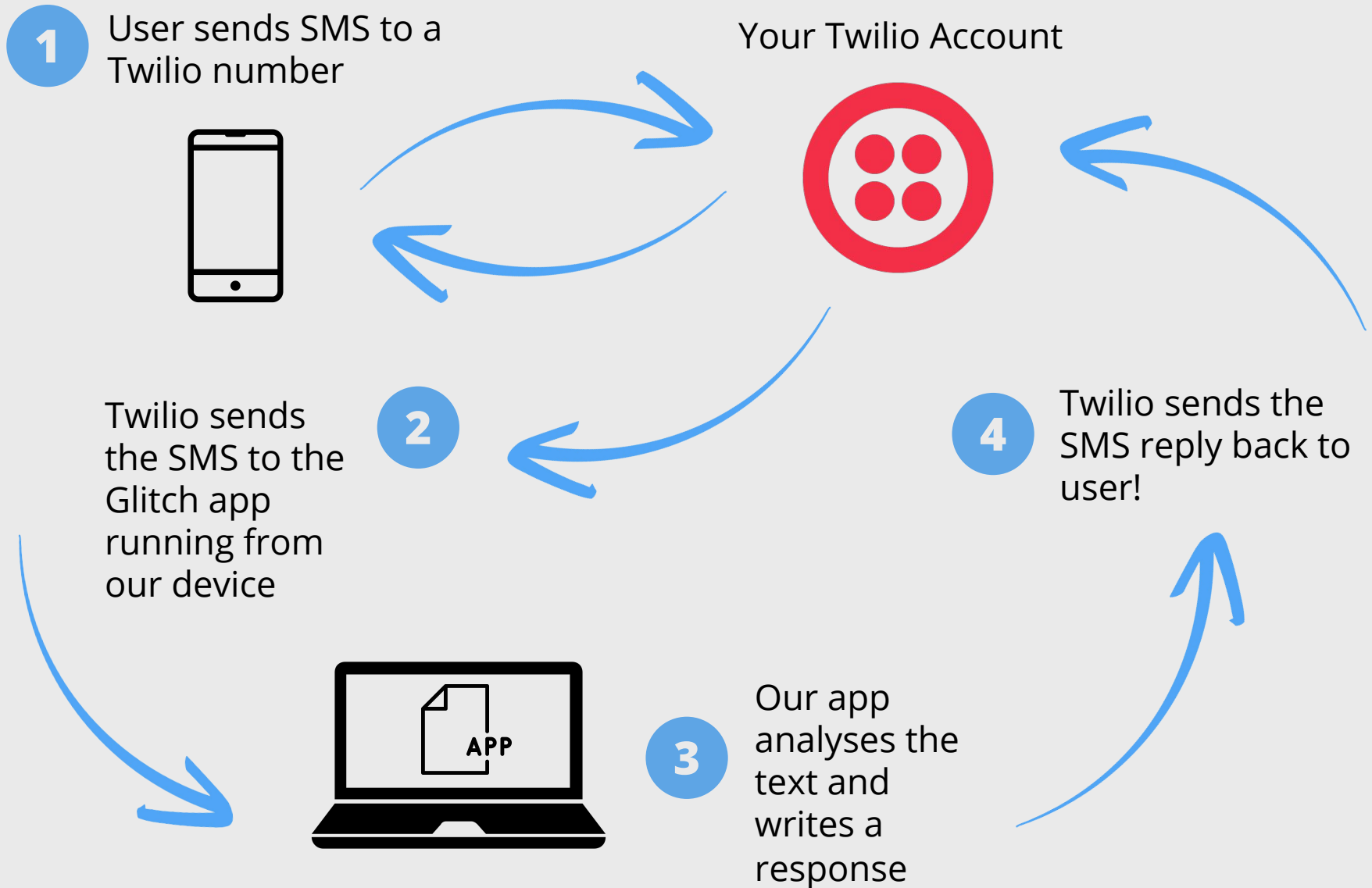
Well, SOME of it's magic. Let's learn
how it works.



How does it work?

- User sends a text to Twilio number
- Twilio receives the text and sends it to your app
- Our app selects a random image URL from a hard coded list
- Our app compiles a Twilio message containing the URL and sends it to the user.

Let's see how it looks graphically!



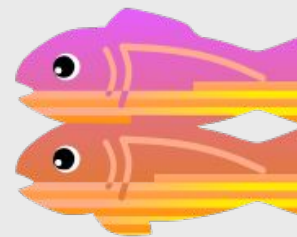
Getting Set Up

We're going to use a few services to make developing this app SUPER easy:



Twilio

Twilio allows us to create SMS and Voice apps. You send a text and it starts running an app behind the scenes.



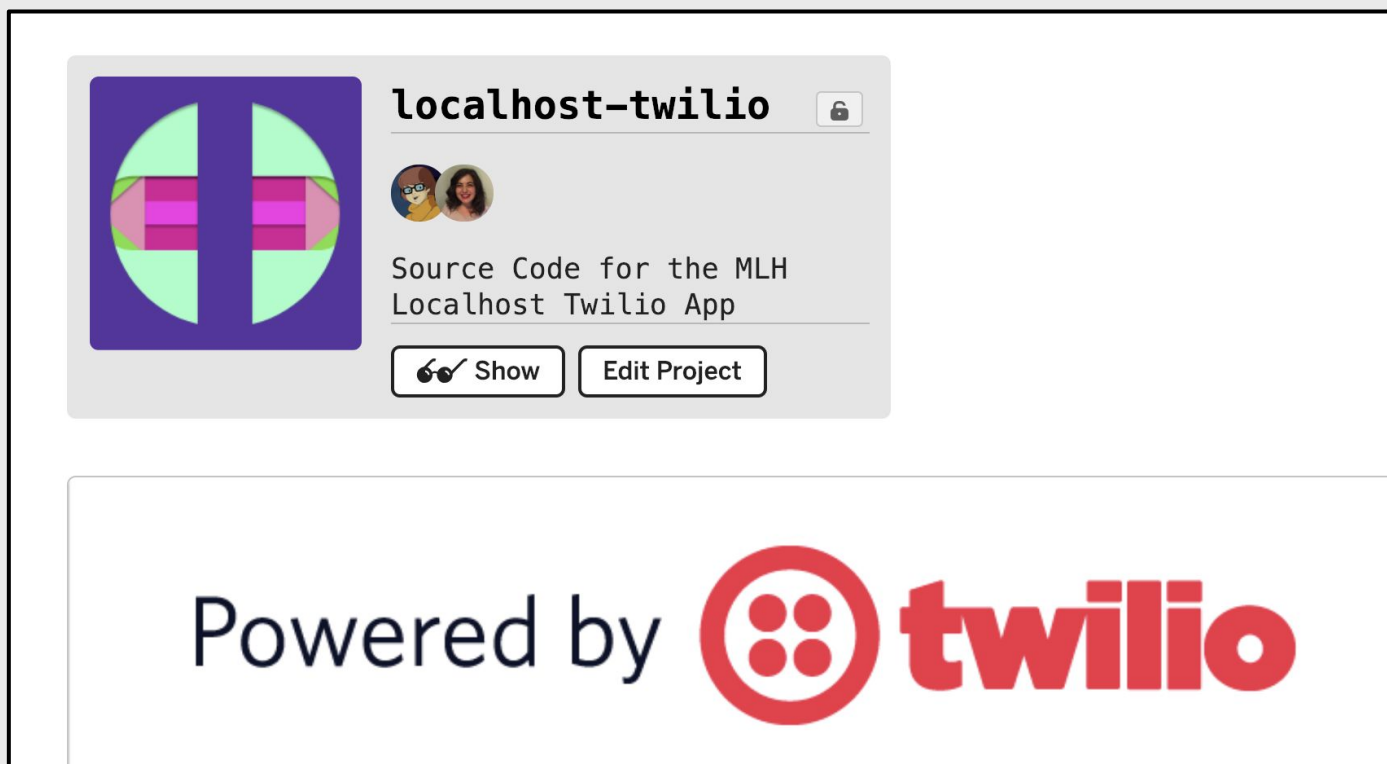
Glitch

Glitch lets you code right in the browser and deploy it instantly!

Great, now let's see the code!

Head to the URL below to see the code on Glitch!

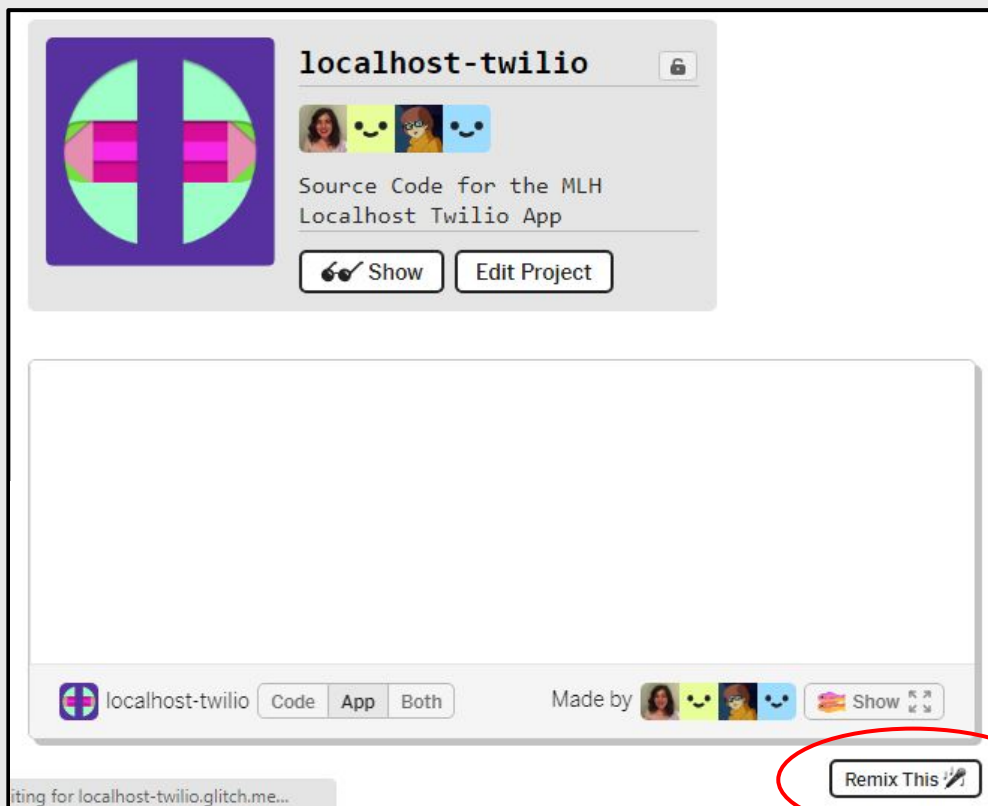
<http://mlhlocal.host/twilioapp-code>



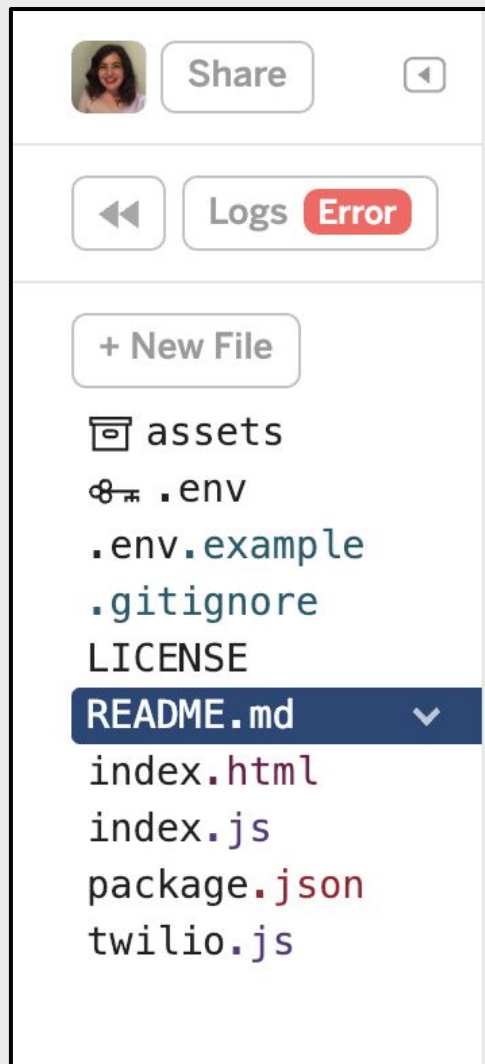
Remix the Project

Click the **Remix This** button to create a copy of the code.

We're going to make some edits and get the app running on our own devices.



Code Review



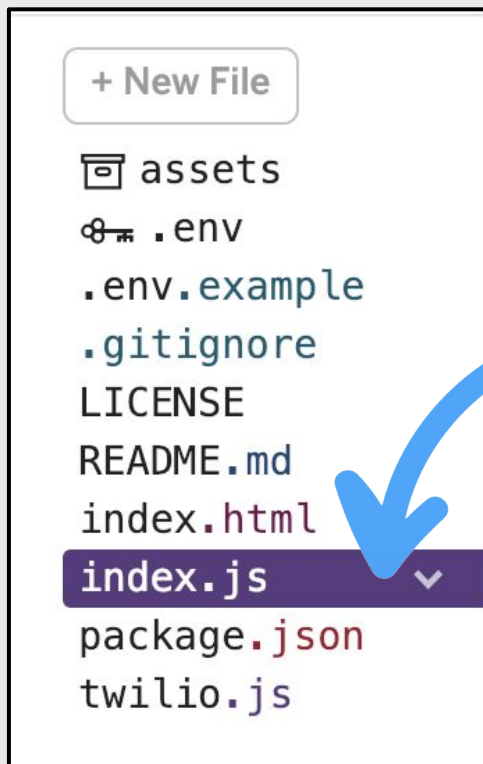
The left panel in Glitch shows all the files we have in the project.

We have some **Javascript** files which communicate with **Twilio**.

This way, we can send SMS messages to users containing anything we want.

Let's go under the hood!

index.js



Select **index.js** and we'll take a look at it line by line.

index.js

```
1 require('dotenv').config()
2 const express = require('express')
3 const bodyParser = require('body-parser');
4 const app = express()
5 const twilio = require('./twilio.js')
6
```

Lines 1-5: Loads the modules needed by the app. They are listed in the **dependencies** of the `package.json` file.

Key Term

Dependencies: When one piece of software relies on another piece to run.

index.js

```
7~ app.use(bodyParser.urlencoded({  
8   extended: true  
9 }));  
10  
11 const port = process.env.PORT || 3000  
12
```

Lines 7-9: Activates bodyParser, a module that allows us to read SMS messages that users send to Twilio.

Line 14: Sets the network port to either the environment variable PORT or 3000.

index.js

```
13~ const arrayMessage = [  
14   'http://www.gifs.eco.br/wp-content/uploads/2017/11/0lha-pra-mim.gif',  
15   'https://media.giphy.com/media/Cprj15D9UMHde/giphy.gif',  
16   'https://media.giphy.com/media/cfuL5gqFDreXxkWQ4o/giphy-downsized-large.gif',  
17   'http://gif-finder.com/wp-content/uploads/2017/09/There-are-two-types-of-dogs.gif',  
18   'https://i.gifer.com/wau.gif',  
19   'https://i.pinimg.com/originals/c6/8e/e9/c68ee9b4782e31405269eca1c07b5c39.gif',  
20   'https://gifimage.net/wp-content/uploads/2017/11/gif-de-tecnologia-4.gif',  
21   'https://media.giphy.com/media/U0k61gfRIIsA/giphy.gif',  
22   'http://meowgifs.com/wp-content/uploads/2013/02/dancing_cat.gif',  
23   'https://i.pinimg.com/originals/9e/75/9f/9e759fd37ccd98da121b670249f34afa.gif',  
24   'http://gifs.eco.br/wp-content/uploads/2017/12/Gifs-Legais-Baixar-3.gif',  
25   'http://happynewyearsms2017.com/wp-content/uploads/2018/10/Happy-New-Year-2019-GIF-15-1.gif',  
26   'https://rockntech.com.br/wp-content/uploads/2014/06/gifs-eventos-incriveis-natureza_4.gif',  
27   'https://www.oficinadanet.com.br/imagens/post/15974/gif_7.gif',  
28   'https://www.thinkgeek.com/images/products/frontsquare/jqmn_lightup_pikachu.gif',  
29   'http://i.giphy.com/I0w8pC7LX2ZP0.gif',  
30   'https://i.gifer.com/ALRo.gif',  
31   'https://happynewyear2019.us/wp-content/uploads/2018/10/2019-Happy-New-Year-gif.gif',  
32   'https://i.pinimg.com/originals/9d/10/c5/9d10c5eb0254c2a6742150579ffb2207.gif',  
33   'http://i.imgur.com/foyXPJ.gif',  
34 ]
```

Lines 13-34 This is the hard-coded list of items that we could send back to the user. Feel free to add your own!

index.js

```
43~ app.post('/', (req, res) => {  
44   const phoneNumber = req.body.From  
45   var index = Math.floor(Math.random() * (arrayMessage.length - 1));  
46   twilio.sendMessage(phoneNumber, arrayMessage[index]);  
47  
48   res.writeHead(200, {'Content-Type': 'image/gif'});  
49 })  
50  
51 app.listen(port, () => console.log(`Example app listening on port ${port}!`))  
52
```

Line 43: When our app gets a POST request (when someone sends a message to your Twilio number), it does the following:

Line 44: Retrieve the phone number that sent the message

Line 45: generate a random index inside our arrayMessage list

index.js

```
43~ app.post('/', (req, res) => {  
44   const phoneNumber = req.body.From  
45   var index = Math.floor(Math.random() * (arrayMessage.length - 1));  
46   twilio.sendMessage(phoneNumber, arrayMessage[index]);  
47  
48   res.writeHead(200, {'Content-Type': 'image/gif'});  
49 })  
50  
51 app.listen(port, () => console.log(`Example app listening on port ${port}!`))  
52
```

Line 46: Use the Twilio API to send the message back to the phone number

Line 48: sends an 'ok' status to our local server to finish.

Line 51: Starts the app running on the user's network port and prints a message to the console to notify us.

.env.example

Now click on **.env.example** and let's look at the environment variables.



.env.example

.env.example contains examples of the environment variables our app needs, like API keys.

When we need to use these credentials, we can use the variable name instead of writing the values again and again.

.env.example

```
1 TWILIO_ACCOUNT_SID=  
2 TWILIO_AUTH_TOKEN=  
3 TWILIO_PHONE_NUMBER=  
4
```

Like in **twilio.js**!

```
1 const ACCOUNT_SID = process.env.TWILIO_ACCOUNT_SID
```

Soon, we'll be replacing these values with your own credentials.

.env.example

Where do we get these variables?

.env.example

```
1 TWILIO_ACCOUNT_SID=  
2 TWILIO_AUTH_TOKEN=  
3 TWILIO_PHONE_NUMBER=  
4
```

These come from Twilio

twilio.js

Now click on `twilio.js` and let's look at the code!



twilio.js

```
1  const ACCOUNT_SID = process.env.TWILIO_ACCOUNT_SID
2  const AUTH_TOKEN = process.env.TWILIO_AUTH_TOKEN
3  const TWILIO_NUMBER = process.env.TWILIO_PHONE_NUMBER
4
5  const client = require('twilio')(ACCOUNT_SID, AUTH_TOKEN);
6
```

Lines 1-3: Creates the credentials we need to configure Twilio: an SID (security identifier), an authentication token, and our Twilio phone number.

Line 5: Loads the Twilio module and gives it the SID and authentication token.

twilio.js

```
7  const sendMessage = (to, message, callback) => {  
8    client.messages  
9    .create({  
10     body: message,  
11     from: TWILIO_NUMBER,  
12     to,  
13   })  
14   .then(callback)  
15   .done();  
16 };  
17
```

Lines 7-16: Creates a function to craft a new SMS message:

- **to:** A phone number to send it to
- **message:** the content of the SMS
- **callback:** something to do after the message is sent


twilio.js

```
18 module.exports = {  
19   sendMessage,  
20 };  
21
```

Lines 18-20: This allows the sendMessage function to be used in other files in the app!

Now that you are familiar with the code, let's set up Twilio so it can handle the messaging function for us.

Table of Contents

1. What's Twilio?
2. MLH Memes App
-  3. Twilio Console
4. Test your App
5. Next Steps

Setting up Twilio

Let's register for Twilio so we can set up our phone number and get the credentials we need to run the app!

Firstly, head to this URL:

`mlhlocal.host/twilio`

Register for Twilio

Enter your name, email address, and create a password.

Click **Get Started**.

Get started with a free Twilio account.
No credit card required.

CHOOSE YOUR LANGUAGE

JavaScript



Get Started

By clicking the button, you agree to our [legal policies](#).

Already have an account? [Login](#)

Register for Twilio

Enter your mobile phone number, wait for an SMS, and enter the verification code you receive.

1

Verify Your Identity

Verify Phone Number

NUMBER



+1

Phone Number

Verify

We will contact you at the number above with a verification code

- ☐ The phone number you provide will be used for authentication. A Twilio specialist may also use this number to reach out with free onboarding assistance. If you have a business phone number, please check this box.

2

Verify Phone Number

Please enter the verification code we sent to <7555467524>

Verification Code

Submit

[Want to verify with a Call instead of SMS?](#)

Didn't receive a code?

Resend Code (14)

Register for Twilio

You can go ahead and click [Skip to dashboard](#).

3

`<Message>Welcome! Let's customize your experience!</Message>`



Do you write code?

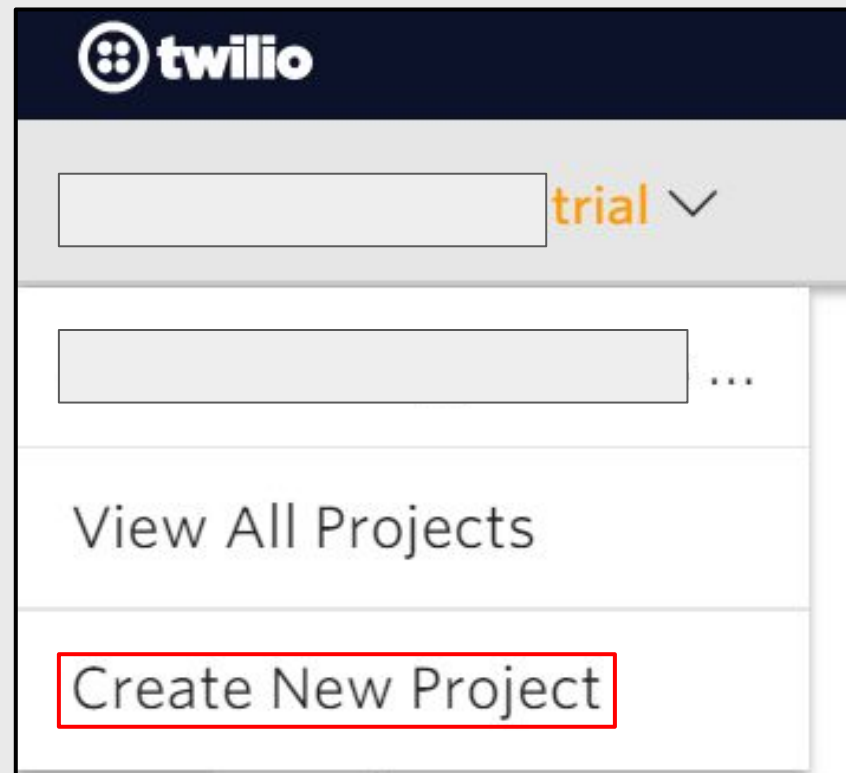
Yes

No

Skip to dashboard

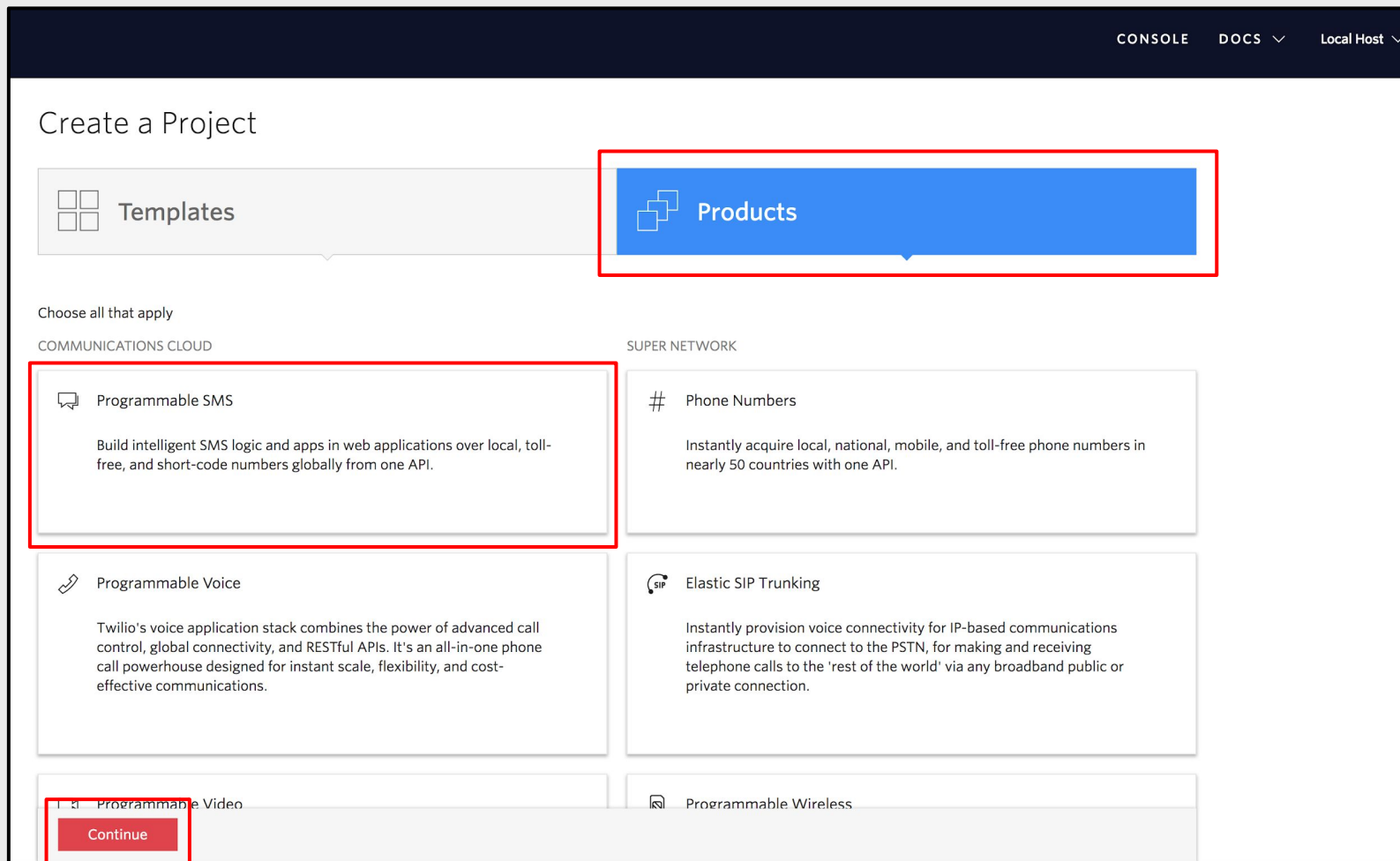
Create a New Project

Click the upside arrow icon next to your email, select **Create New Project**



Create a New Project

Click **Products**, select **Programmable SMS**, then click **Continue**.



CONSOLE DOCS Local Host

Create a Project

Templates

Products

Choose all that apply

COMMUNICATIONS CLOUD

Programmable SMS

Build intelligent SMS logic and apps in web applications over local, toll-free, and short-code numbers globally from one API.

Phone Numbers

Instantly acquire local, national, mobile, and toll-free phone numbers in nearly 50 countries with one API.

Programmable Voice

Twilio's voice application stack combines the power of advanced call control, global connectivity, and RESTful APIs. It's an all-in-one phone call powerhouse designed for instant scale, flexibility, and cost-effective communications.

Elastic SIP Trunking

Instantly provision voice connectivity for IP-based communications infrastructure to connect to the PSTN, for making and receiving telephone calls to the 'rest of the world' via any broadband public or private connection.

Programmable Video

Programmable Wireless

Continue

Create a New Project

Type in a name for your project and click [Continue](#).

Custom Project

Give your project a name

You can make changes later if you need to.

PROJECT NAME

Create a New Project

Select [Skip This Step](#).

Custom Project

Who do you want to build it with?

You don't have any teammates yet

Invite your team to this project. If you think other users would be interested in tracking your progress or can help you build it, you can invite them to Twilio with a variety of permissions. You can also manage the users associated with your project at any time from the Project Settings page.

For more information about managing users and project roles please visit our [Help center](#) ↗

Invite a Teammate

[< Go Back](#)

Invite Selected

Skip This Step

Create a New Project

Click **Get Started**.

First Project Dashboard

Project Info

We've customized your dashboard based on the products you selected. Use the product get started guides to get up and running.

We can't wait to see what you build!

PROJECT NAME First Project [edit](#)
 ACCOUNT SID AC9b41201066aceb6bf8e443f118f767db
 AUTH TOKEN [view](#)
 Owner 1 [manage](#)
 2FA Disabled [edit](#)



Programmable SMS

Build intelligent SMS logic and apps in web, mobile, toll-free, and short-code numbers globally from one API

Get Started

[Programmable SMS](#)

[Developer Docs](#)



Invite Your Team

Developer

Invite

[Manage Users](#)

[I don't have developers](#)



Errors & Warnings (Debugger)

You have no recent errors or warnings.

[View Debugger](#)

Register a Phone Number

We're going to register a telephone number to use in our project.

Select [Get a number](#).

Build with Programmable SMS

[You have a Trial Account](#) >

First let's get a Twilio phone number

In order to make calls or send messages through the Twilio API, you need to get a Twilio phone number.

[Get a number](#)

Register a Phone Number

Select **Choose this Number**. Make a note of your number!

Your first Twilio Phone Number

(754) 219-9755

Don't like this one? [Search for a different number](#)

 This United States phone number has the following capabilities:

 **Voice:** This number can receive incoming calls and make outgoing calls.

 **SMS:** This number can send and receive text messages to and from mobile numbers.

 **MMS:** This number can send and receive multi media messages to and from mobile numbers.

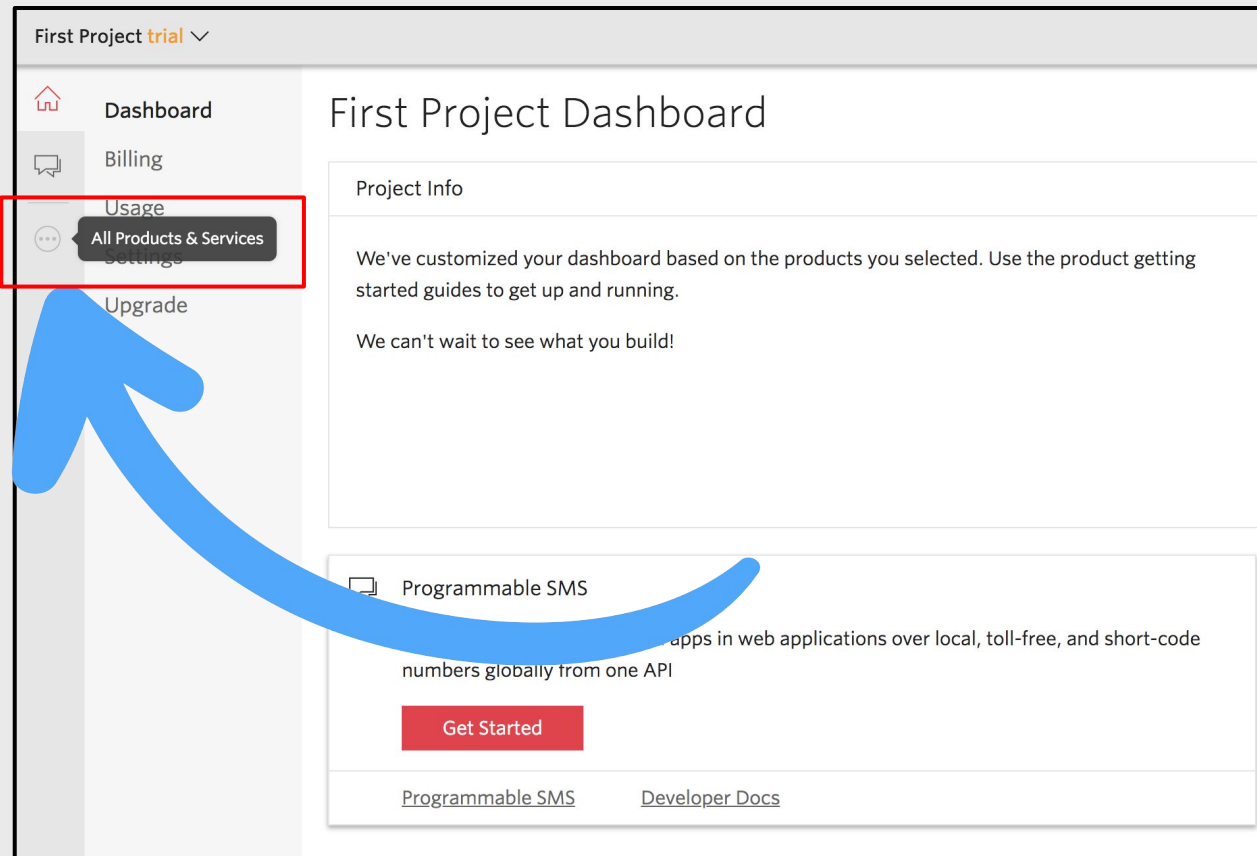
Cancel

Choose this Number

43

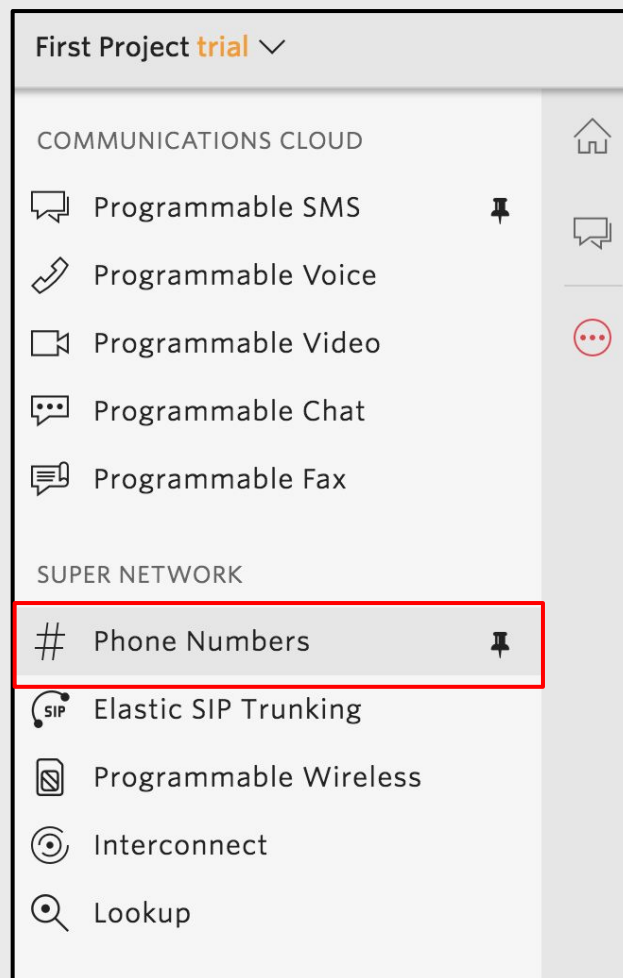
Configure the Phone Number

Click the **All Products and Services** icon on the left panel (it has three dots).



Configure the Phone Number


Then click [Phone Numbers](#).



Configure the Phone Number

Click on your new number.

Active Numbers


CLICK + TO BUY NEW NUMBER




Number

▼

Voice URL

▼

Filter

NUMBER	FRIENDLY NAME	ABILITIES	CONFIGURATION
<div> <div>+1 754 219 9755</div> <div>Hollywood, FL</div> </div>	219-9755	  	Voice POST: https://demo.twilio.com/welcome/voice/ Messaging POST: https://demo.twilio.com/welcome/sms/reply/

* Can send/receive calls to domestic numbers only

† Can send/receive sms to domestic numbers only

‡ This number does NOT support SIP Trunking

(national) A non-geographic number

(beta) This number is new to the Twilio Platform

(hosted) This number is hosted on the Twilio Platform

Webhooks

Here is where we configure what happens when someone sends an SMS or a call to your new phone number.

Messaging

CONFIGURE WITH	Webhooks, TwiML Bins, Functions, Studio, or Proxy ▼		
A MESSAGE COMES IN	Webhook ▼	<input type="text"/>	HTTP POST ▼
PRIMARY HANDLER FAILS	Webhook ▼	<input type="text"/>	HTTP POST ▼

Leave this webpage open for now - we'll come back to it soon. Now, we're going to set up the response.

Authentication Tokens

Let's find our authentication tokens.

Head to the Twilio Console:

<http://twilio.com/console>

Authentication Tokens

Your Dashboard shows your SID. Click [view](#) to see your Auth token too.

My First Project Dashboard

Project Info

We've customized your dashboard based on the products you selected. Use the product getting started guides to get up and running.

We can't wait to see what you build!

PROJECT NAME My First Project [edit](#)

ACCOUNT SID AC94ff5593721806ad14034782b2e7a5b3

AUTH TOKEN [view](#)

 Owner  1 [manage](#)  2FA Disabled [edit](#)

Adding Your Authentication Tokens

Go back to Glitch, open `.env` **(NOT `.env.example`)** and add your phone number and auth tokens.

`.env`

```
4 TWILIO_ACCOUNT_SID=29B234b7AS2328J  
5 TWILIO_AUTH_TOKEN=Ac sdf234765922af02df  
6 TWILIO_PHONE_NUMBER=+447554218678  
7
```

Your codes will be unique to you!

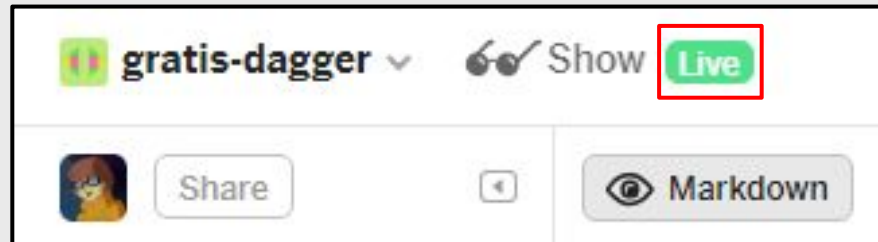
Adding Twilio Webhook

Almost there!

Now all we have to do is launch our app in Glitch, and give Twilio the URL.

Then, when we send an SMS to our number, it will redirect to our app and generate a response.

In Glitch, click [Live](#).



Adding Twilio Webhook




Your app should launch in a new tab. Copy the URL - it will be something like `https://random-name.glitch.me`.

Head back to your Twilio Phone number dashboard:

`https://twilio.com/console/phone-numbers/`

Adding Twilio Webhook

Click on your registered number.

NUMBER	FRIENDLY NAME	CAPABILITIES	CONFIGURATION
+44 1431 905010 ↑ Helmsdale, Scotland	441431905010	  	Voice POST: https://c4337d4a.ngrok.io/call Messaging POST: https://gratis-dagger.glitch.me/

Scroll down to **Messaging** and add your app url to the Message Webhook.

Click **Save**.

Messaging

CONFIGURE WITH Webhooks, TwiML Bins, Functions, Studio, or Proxy

A MESSAGE COMES IN Webhook HTTP POST

PRIMARY HANDLER FAILS Webhook HTTP

Save

Table of Contents

1. What's Twilio?
2. MLH Memes App
3. Twilio Console
-  4. Test your App
5. Next Steps

Testing Your App

Trying sending an SMS saying '**show me a meme!**' to your Twilio number!



**Raise your hand and ask for
help if your app isn't working!**

Table of Contents

1. What's Twilio?
2. MLH Memes App
3. Twilio Console
4. Test your App
-  5. Next Steps



Let's recap quickly...

- 1 Twilio is a PaaS platform which allows you add communication functionality to your projects.
- 2 The Twilio SMS API allows you to create all sorts of apps that use a texting API.
- 3 You can use the Twilio console to build the app of your dreams!

What did you learn today?

We created a fun quiz to test your knowledge and see what you learned from this workshop.

[**http://mlhlocal.host/quiz**](http://mlhlocal.host/quiz)



Where to go from here...

1

Learn about other Twilio APIs - Voice, Fax, and more!

2

Play TwilioQuest at **mlhlocal.host/twilioquest!**

3

Complete the additional practice problems.
These problems & way more are in your email!

Keep Learning: Practice Problems for Later

Extra Practice Problem 1:

Update the app to ask the participants how many results they want and return that many.

Extra Practice Problem 2:

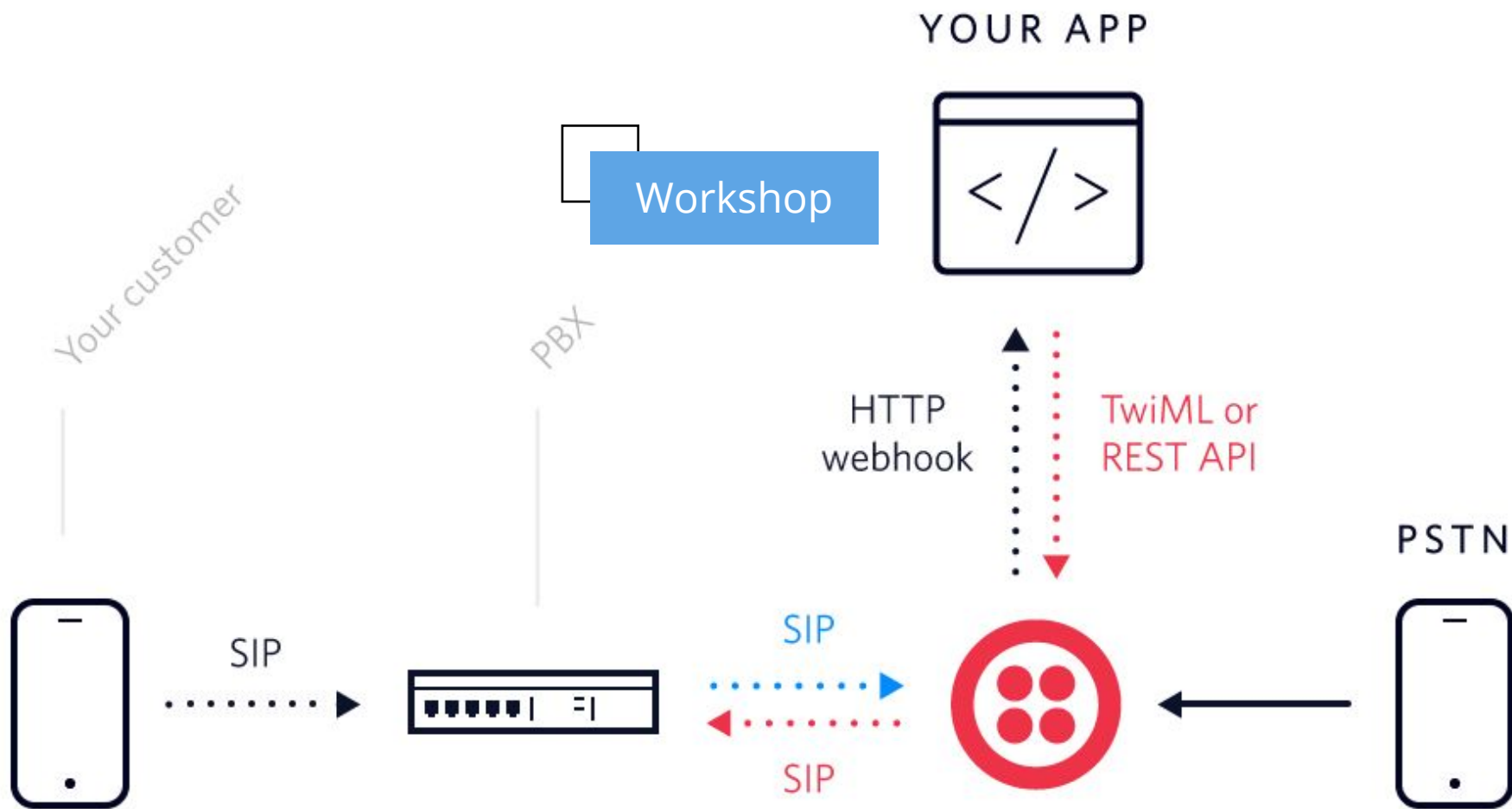
Update the app to ask participants if they want a GIF or a meme. Then, respond to their request accordingly!

Learning shouldn't stop when the workshop ends...

Check your email for access to:



- These workshop slides
- Practice problems to keep learning
- Deeper dives into key topics
- Instructions to join the community
- More opportunities from MLH!



Connect to Twilio over SIP