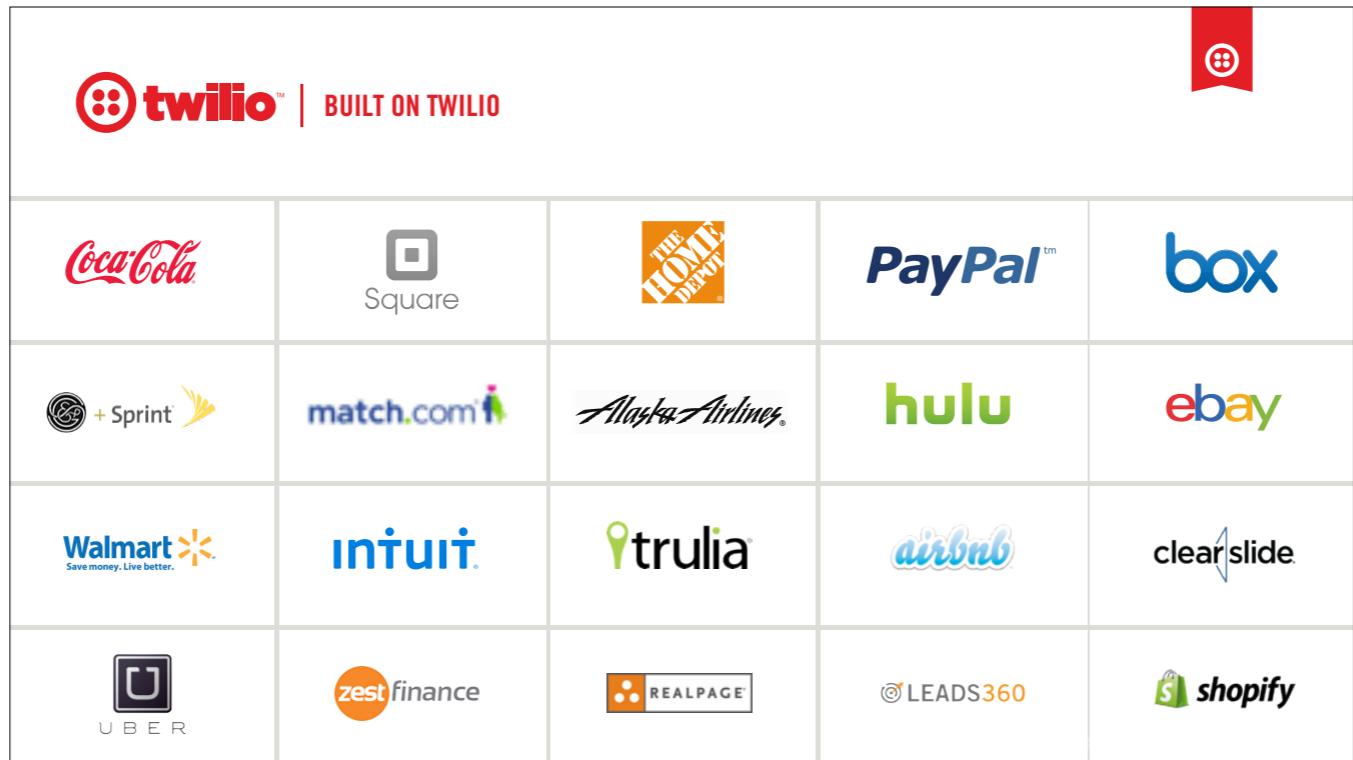


TECHNICALLY, WRITING IS EASY



@jreyesdesign
jreyes@twilio.com
github.com/jarodreyes



•Companies of all sizes around the globe are taking advantage of Twilio to build smarter, more agile communications solutions.



HOW THE PLATFORM WORKS



- The platform involves three components:

- An end-user calling or SMS'ing phone numbers you purchase from Twilio
- Twilio receiving the call and making a web request to your server for instructions on how to proceed
- Your web application responding with instructions on how to proceed with the call



RESPONSIBILITIES

So what do I actually do? Who am I?
Front-end Developer, designer, hacker. New York to San Diego.
First technical content producer at Twilio.
A lot has changed over the last year, just like the market.

The screenshot shows a web browser displaying the Twilio REST API documentation for the 'Messages' resource. The page has a header with links to 'Quickstart', 'HowTo's', 'Helper Libraries', 'API docs', and 'Security'. On the right side, there is a sidebar with navigation links for 'Quickstart Tutorials', 'HowTo's and Examples', 'Helper Libraries', 'TwiML Reference', 'REST API Reference' (version 2018-04-01), 'Overview', 'Request Formats', 'Response Formats', 'Test Credentials', and a list of 'REST Resources' including Accounts, Subaccounts, AvailablePhoneNumbers, OutgoingCallers, IncomingPhoneNumbers, Applications, ConnectApps, AuthorizedConnectApps, and Calls.

REST API: Messages

A [Message instance](#) resource represents an inbound or outbound message. When you send an SMS or MMS message via the REST API, use the `<Message>` verb in TwiML, or someone sends a message to one of your Twilio numbers Twilio creates a Message instance resource.

The [Messages list](#) resource represents the set of messages sent from and received by an account.

If you are looking for documentation on the deprecated SMS/Messages resource, that documentation is available [here](#).

Message Instance Resource

This resource represents an individual SMS or MMS message.

Resource URI

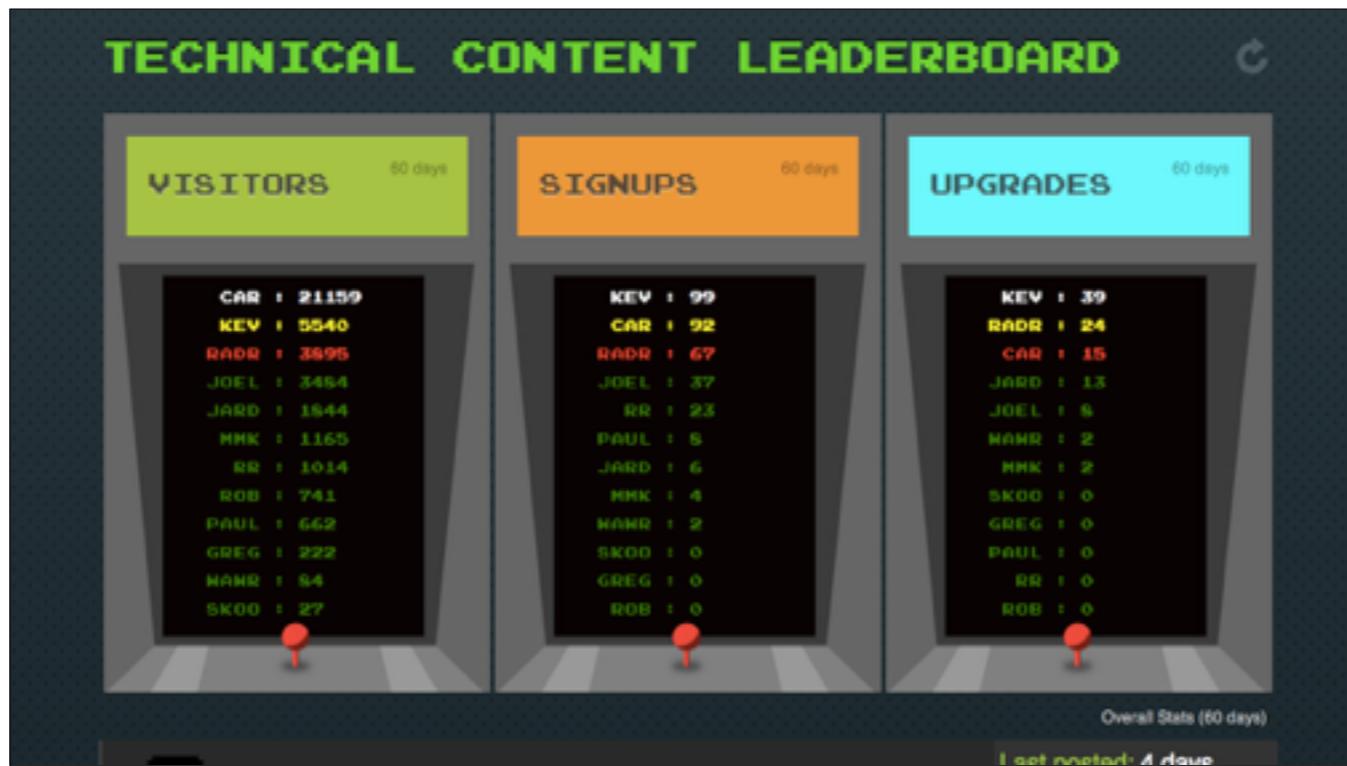
```
/2018-04-01/Accounts/{AccountSid}/Messages/{MessageSid}
```

Resource Properties

30% Writing documentation.

```
// routes/voice.js
8  var input = request.body.RecordingUrl || request.body.Digits;
9
10 // helper to append a new "Say" verb with alice voice
11 function say(text) {
12     twiml.say(text, { voice: 'alice'});
13 }
14
15 // respond with the current TwiML content
16 function respond() {
17     response.type('text/xml');
18     response.send(twiml.toString());
19 }
20
21 // Find an in-progress survey if one exists, otherwise create one
22 SurveyResponse.advanceSurvey({
23     phone: phone,
24     input: input,
25     survey: survey
26 }, function(err, surveyResponse, questionIndex) {
27     var question = survey[questionIndex];
28
29     if (err || !surveyResponse) {
30         say('Terribly sorry, but an error has occurred. Goodbye');
31         return respond();
32     }
33
34     // If question is null, we're done!
35     if (!question) {
36         say('Thank you for taking this survey. Goodbye!');
37         return respond();
38     }
39
40 })
41
42
43
44
```

30% Coding – Usually 2–3 weeks straight.



40% my time doing admin stuff.

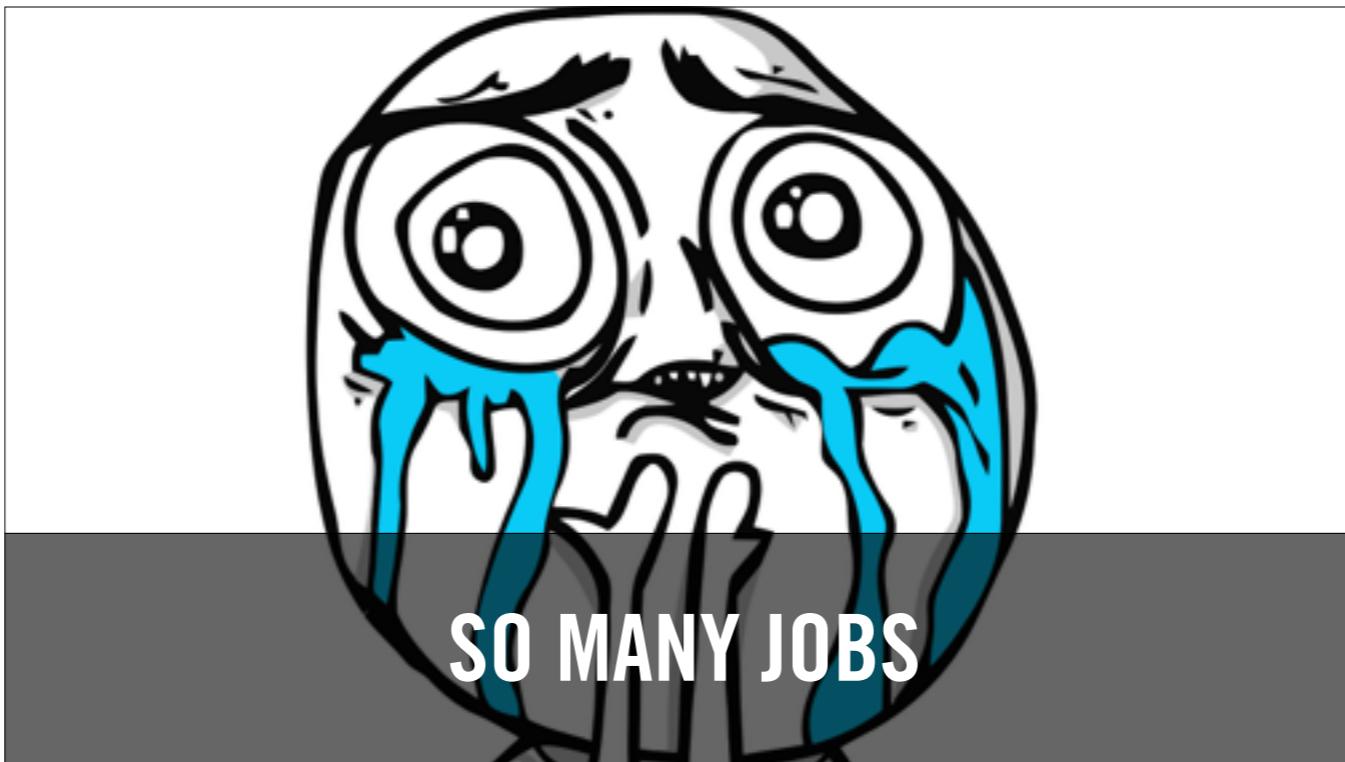
- Keeping the content pipeline full. Writing tutorials.
- Metrics, user reporting.
- User testing
- Managing external resources

Cultural shift- writing needs to get better.



TECHNICAL WRITING

How many have written for a technical audience before?
Whiteboard – What are some examples of Technical Writing?



Obviously this is a lot of different content.

One Note: Your job = easier if you contribute to building a better product.

Your job is multi-faceted.

You are translator.

You are a teacher.

You have design influence.

You have voice influence.

My goal is to equip you with some tools in order to make the best decisions when writing for a technical audience.



Before you make any decision you simply need to ask yourself 4 questions.

NARRATIVE

Always start with Narrative. What is the story here? But we'll skip this for now because before you can write a successful narrative you need to know two things.

OBJECTIVE

What is the purpose of this ?

What should the reader take away?

This should be clear to you and the reader.

Tip: look past the obvious goal “get readers” or “get signups”.

Look for genuine appeal.

AUDIENCE

Who is the audience?
Where are they?
What is their relationship with the material?
Limitations? Writing documentation for a ship captain?

DELIVERY

Knowing the Audience, what is the best route for delivery?
Book? Documentation? Report? Blog Post? Video? Diagram? Infographic?

NARRATIVE

What is Narrative? Structure..... STORY..... and hopefully WONDER.

Evolved to learn from stories.

Early civilization taught basic skills through story.

Because our brain thinks in Narrative. Organizes information this way.

Our brain lights up different when information is through story, areas that process memories light up which is why campfire nights stay with us.

Narrative pre-dates writing. Storytelling predates writing, with the earliest forms of storytelling usually oral combined with gestures and expressions. In addition to being part of religious ritual, rock art may[original research?] have served as a form of storytelling for many ancient cultures. The Australian aboriginal people painted symbols from stories on cave walls as a means of helping the storyteller remember the story. The story was then told using a combination of oral narrative, music, rock art, and dance, which bring understanding and meaning of human existence through remembrance and enactment of stories.[1] People have used the carved trunks of living trees and ephemeral media (such as sand and leaves) to record stories in pictures or with writing. Complex forms of tattooing may also represent stories, with information about genealogy, affiliation, and social status.

On-boarding Instructions for Viewsaurus

```
Open up your computer.  
Open up terminal.  
Type `npm install -g viewsaurus`  
cd ../your_github_project  
type `saurus new`  
type `saurus author`  
go to "localhost:8080"  
Edit config.json  
Edit index.jade  
git checkout gh-pages && git commit -m  
"viewsaurus content" && git push -u origin  
master  
All done!
```

Want to take a guess at the objective of this document?

Audience: npm install?

What should be in config.json?

What is the objective?

How would you improve this?

Given the objective "How to write an interactive tutorial for any github repo using Viewsaurus"

OBJECTIVE EXERCISE

Dissect the game.

One person comes up to the desk, reads the rules.

The group then must play the game for 10 minutes.

After the round they must write down the rules to the game and pass to the next group.

5 MIN to read rules, fill in the rest.

10 MIN to play.

5 MIN to revise rules.

10 MIN for group chat.

15 MIN break.

Crazy Eights

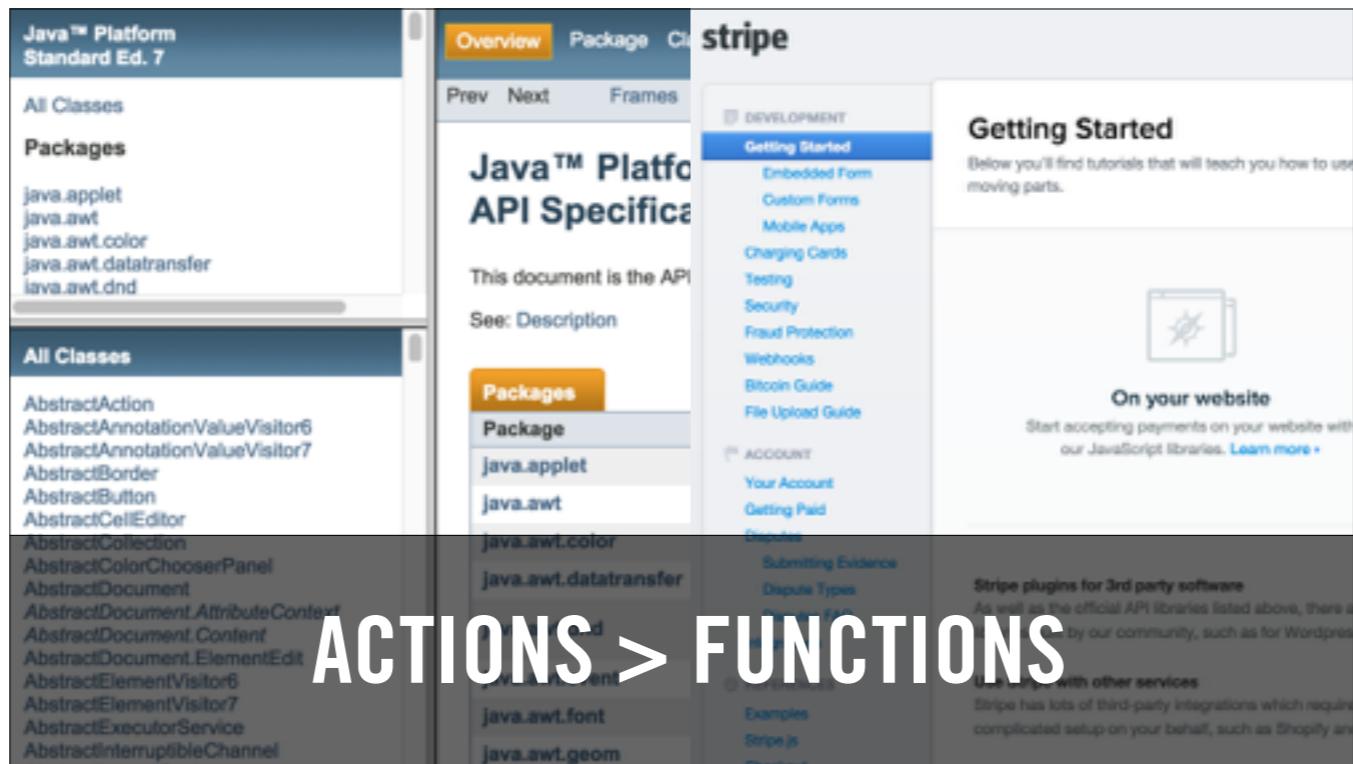


Be clear (whittle down your thoughts)

Be accurate (Possibly, sometimes, it may be, it's possible that)

Be humble (talk at the same level)

SPLICE hyperbole



The image shows two side-by-side API documentation examples. On the left is the Java™ Platform Standard Ed. 7 API documentation, featuring a sidebar with 'All Classes' and 'Packages' sections, and a main content area listing various Java classes. On the right is the Stripe API documentation, which has a more user-friendly, website-like interface with sections for 'Getting Started', 'On your website', and 'Stripe plugins for 3rd party software'. The word 'stripe' is written vertically between the two screenshots.

ACTIONS > FUNCTIONS

Write about actions, not functions
We call this behavior-driven documentation.

The image displays two web pages side-by-side. The left page is from Khan Academy, titled 'Computer programming' under the 'COMPUTING' category. It features a green sidebar with options like 'New Program (JS + Processing.js)', 'New Webpage', 'New SQL script', and 'Browse Creations'. The main content area contains text about programming drawings, animations, and games using JavaScript & Processing.js, or creating webpages with HTML & CSS. The right page is from Goodreads, titled 'Api Documentation'. It includes a 'Getting Started' section with instructions on how to access data using a developer key, and a code example in a terminal window showing 'curl "https://www.goodreads.com/search.xml?key=YOUR_KEY&q=insert27+Games"'.

Track your developers progress.
Insert tokens into examples.
Lookup API calls.

KNOW YOUR DEVELOPER



COMMON MISTAKES

Here are some common mistakes and how to avoid them.



Know your topic

Keep topics in the subject

Hurricane Sandy caused the outage last week with our server.

The server went offline due to hurricane Sandy.



Stress

Stress new information at the end of the sentence.

I built the Garduino, a
garden monitor, in an effort
to better take care of my
garden.

In order to keep my garden
healthy I built a garden
monitor called Garduino.

Stress new information at the end of the sentence.

I built the Garduino, a garden monitor, in an effort to better take care of my garden.

In order to keep my garden healthy I built a garden monitor called Garduino.



Nominalization

DON'T USE THEM!

“As a pre-requisite of deployment, the certification of the accuracy of a developer's code must be made upon completion. Certification is verified by two members of the same team in which the developer is assigned.”

“Before pushing your code, two other team members must certify that it is technically accurate.”

Nominalizations confuse the issue, as do any abstract nouns. If you find you are using them in your prose, removing them will almost always clean up your prose.

Deployment is here...

Initialization...



TOOLS

Gist - code samples

Cisco Style Guide

Apple Style Guide

Google Docs (for reviewing)



Cadence & Slang: <http://cadence.cc/>

Joseph M. Williams: Style toward clarity and grace.

Style Guides – Chicago element of style as well.

All of these can be found in the github address given on your printed packet.

CONTENT SPEC



TITLE

Objective: What is the purpose?

Audience: Who is your audience?

Delivery: Publishing platform, promotion sites.

Outline:

- Outline your narrative here
- Section headers
- Call out dependencies (ie; need diagram here)

Waterfall Style:

- Planning: 20%
 - Write your content spec, prepare your narrative
- Writing: 50%
- Review: 20%
- Promotion: 10%

RELUCTANT I.

5 Minutes, tell me about your first day in college.

Q&A

Writing assignment? Brainstorm.

Documentation: how to make it better? Brainstorm.

Hardest part of writing technically.



JAROD REYES

@jreyesdesign
jreyes@twilio.com
github.com/jarodreyes