SendGrid

Parse Webhook Guide

Contents

2 Introduction

3 How Fancy Hands Routes Incoming Email

5 How iDoneThis Interactivity Increases Productivity

6 Sendgrid Developer Evangelist Projects

8 Parse Webhook Technical Overview

13 Parse Webhook: Interaction and Collection

Receive Emails with Incoming Parse Webhook

SendGrid makes it easy to send email. After all, "send" is right in the name. We also have an easy way to receive email directly in your application. You can use it to welcome feedback, interact with your customers or even collect data.

Make Email a Two Way Street

When it comes to our personal email accounts, we expect to both send and receive. If we get a message from a friend, that friend could easily be offended if we don't reply. Yet, with email sent from applications so many of us have no mechanism in place to accept a message back. In many cases it's about getting our message to our customer's inbox.

At SendGrid, we recommend not including "noreply" as your "from" address. Recipients should feel messages are coming from a human who values them as another human. Engaging the recipient by inviting a reply is the best way to achieve an air of friendliness.

Imagine how much more engaged a recipient would be if a reply was not only possible, but encouraged. This guide is about the opportunities in flipping the email direction – at least for some of your messages.

Common Use Cases

SOCIAL

Commenting is a staple of social application interaction. Email notifications of new comments is an excellent way to re-engage a user. Reply-able notifications reduce the friction and encourage more conversation.

SUPPORT

If you use software to provide customer service over email, you can use incoming email to create, continue and resolve help tickets. Create special reply addresses for each issue and allow the customer's email reply to go directly into your support application.

CONTENT MANAGEMENT

Accept photos or other types of content via email. It's certainly flashy to have a mobile app, but integrating content via email in addition means more ways to get images, videos or even the written word into your application.

ENDLESS POSSIBILITES

SendGrid developer evangelists have used incoming email to populate music playlists, put audience members on a map, play chess and identify nutritional content from packaging. The beauty of the Parse Webhook is your application can react to incoming email however you choose.

How Fancy Hands Routes Incoming Email to Assistants

Everyone gets overwhelmed with the many tasks that face them each day. Fancy Hands is a service that takes many of those off the shoulders of some very busy people. Fancy Hands assistants perform short, virtual tasks like scheduling a doctor's appointment, 15 minutes of data entry or waiting on hold with a credit card company.

50% of all Fancy Hands tasks are created via email, more than any other method.

To communicate with your assistant, you can use the website, its mobile app or simply email. That last option is extremely popular with Fancy Hands customers, with 50% creating new tasks via email. Fancy Hands uses the Parse Webhook to connect those emails into its backend for assistants.

Fancy Hands was founded by former New York Times web developer Ted Roden to scratch his own itch. "When my wife had our first baby, we were in the hospital, and while we were still in the recovery room I got a book deal," Roden said. "So I had a full time job, a book deal, and a brand new baby. If you asked me to do anything, I just couldn't get it done."

Roden built Fancy Hands so he could use it. The company has been a SendGrid customer since day one, though it has not always used the Parse Webhook. Originally, incoming email came via a feature of its cloud host. "We had a few problems with (the host). It couldn't handle large attachments gracefully at the time," Roden said. "Our only option was to use really long email addresses that didn't reflect our brand and were annoying to users."

How Fancy Hands Routes Incoming Email to Assistants

continued

Gathering attachments via the Parse Webhook is especially useful for Fancy Hands and its customers. Because SendGrid can accept up to 20 MB in every email, large files or many files can be easily attached to each task.

By using SendGrid and the Parse Webhook, Fancy Hands can focus on what it does best – helping its customers focus on what they do best.

> "We have all our code for the Parse Webhook in one small python file. It's pretty simple. I changed it about a month ago to handle a promotion we did, but normally, we don't have to touch it."
>
> **– Ted Roden**

How iDoneThis Interactivity Increases Productivity

When Rodrigo Guzman was in grad school, he would lose track of his physics research. Though he spent a lot of time working, the days would blend together. He wasn't really sure he was making forward progress. So he started outlining the important things he did each day on a calendar.

This problem of understanding one's own productivity exists outside of academia, too. Guzman founded iDoneThis to help others use his simple trick to track their own important things and share them with their team members. Every day iDoneThis sends an email out to its users that requests a reply with what each did that day. Since iDoneThis uses the Parse Webhook, recipients can use the entirety of the application from their email client.

Where the Parse Webhook really shines for iDoneThis is its team reporting tool. Organizations and companies can use iDoneThis to share their efforts. Communicating what everyone is working on becomes easier, but only if collaborators use it. Even though there is a web version of iDoneThis,70% of team members have never visited it.

Founded:
January, 2011

Sends:
750,000 emails/month

Receives:
300,000 replies/month

https://idonethis.com

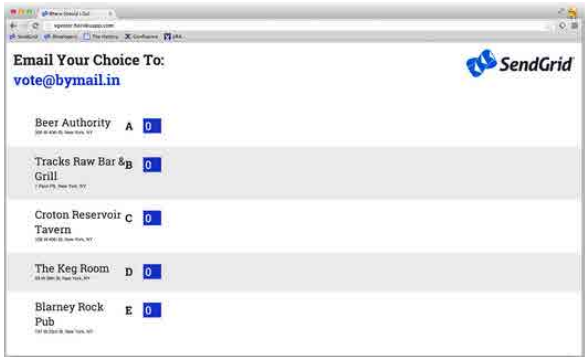
70% of team members use iDoneThis only via email sent to the Parse Webhook.

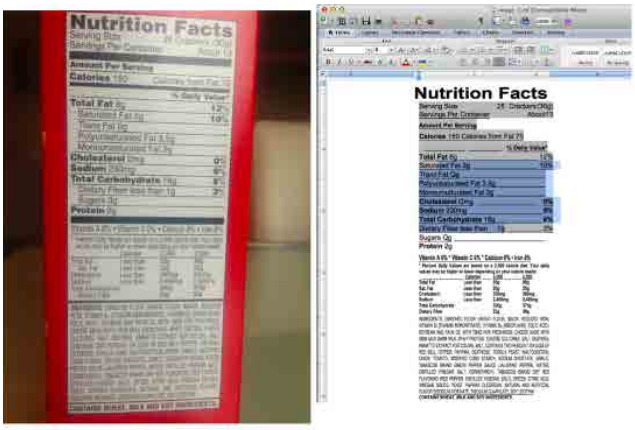
SendGrid Developer Evangelist Projects

Proactively Remove Unengaged Email Addresses

SendGrid has a team of developer evangelists who travel the world helping developers better use technology. The evangelists are all engineers themselves and often build projects that are fun or useful. This section shows a sampling of some of their projects that use the Parse Webhook.

Kunal Batra's first hack as a developer evangelist uses the Parse Webhook and OCR--optical character recognition. Snap a picture and send it to Kunal's special email address. The Parse Webhook accepts the email and posts it to his script, where the photo is sent to an OCR API for processing. Then Kunal's script emails a rich text attachment with the text.





Convert a Picture to Editable Text

Enable a Memorable 21st Birthday

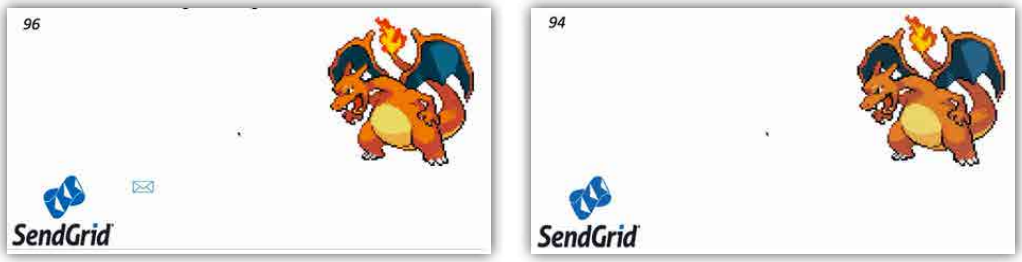
When Nick Quinlan joined SendGrid he wasn't even of legal age to drink in the United States. Since his first hackathon as an evangelist coincided with his 21st birthday, Nick built an app that would help the hackers choose his first drink.

As Nick demonstrated the SendGrid APIs, he invited the audience to email with one of five options in the subject line. The screen auto-populated as results came in. The crowd was impressed and, after the hackathon ended, Nick did have his first drink.

SendGrid Developer Evangelist Projects *continued*

Relive Childhood and Fight a Dragon

Yamil Asusta fondly remembered Pokemon from his childhood. So he built an app that allows audiences to fight a dragon, like he used to do playing the Pokemon card game.



Each new email that comes in is an attack on the dragon and the total hitpoints decrement by the level of the hit. Yamil's app then replies back to the emailer with an overview of the hit, while the entire audience watches the animation projected on the screen.

While these examples are mostly fun, hopefully they give you an idea of the many things possible with the Parse Webhook.

Parse Webhook Technical Overview

The Incoming Parse Webhook allows you to receive email as JSON from within your application. Don't fiddle with server-side configurations and cross your fingers they'll keep processing messages. Don't bother parsing headers to determine who an email is from or where the subject begins or ends. Just set up the webhook once and focus on your application.

How to Configure the Parse Webhook

Configuring the Parse Webhook requires very few technical changes. You'll need to make changes to your DNS settings and your SendGrid account.

① Add SendGrid as an MX record in your DNS settings.

In order for SendGrid to receive email on your behalf, you need to add an MX record to your DNS settings. Many times these settings will be within the company where you registered your domain name or a web hosting provider.

Add a new MX record for your domain or sub-domain and point it to:
mx.sendgrid.net

To use the Parse Webhook, SendGrid needs to receive all email for a particular domain or sub-domain. You don't want to use the same domain name where you already receive email. Instead use something like m.yourdomain.com or a special domain name just for your Parse features.

Parse Webhook Technical Overview *continued*

② Enable the Parse Webhook within the SendGrid control panel.

This is the easy part. Once incoming email is flowing through to us on your behalf, we just need to know where to redirect the email that is parsed.

Enter the following information to enable the Parse Webhook:
- Domain or sub-domain where users will send email
- Full URL (likely on your server) where the webhook should send email data
- Check whether you want SendGrid to perform a basic spam filter on incoming email

Once saved, you'll see the settings on your Parse Webhook page. Now you're ready to test it out.

Receive Email Via JSON

Emails arrive on most email servers as a whole mess of text. An un-parsed email includes headers that most recipients never see. Even the to, from, cc and subject are just plain text that software must separate out manually.

The Incoming Parse Webhook makes this process a whole lot easier. SendGrid does the work of parsing the email fields you expect, plus other useful data.

Field Label	Description
lheaders	The raw headers of the email.
text	Text body of email. If not set, email did not have a text body.
html	HTML body of email. If not set, email did not have an HTML body.

Parse Webhook Technical Overview *continued*

Field Label	Description
from	Email sender, as taken from the message headers.
to	Email recipient field, as taken from the message headers.
cc	Email cc field, as taken from the message headers.
subject	Email subject.
dkim	A JSON string containing the verification results of any dkim and domain keys signatures in the message.
SPF	The results of the Sender Policy Framework verification of the message sender and receiving IP address.
envelope	A JSON string containing the SMTP envelope. This will have two variables: to, which is an array of recipients, and from, which is the return path for the message.
charsets	A JSON string containing the character sets of the fields extracted from the message.
apma_score	Spam Assassin's rating for whether or not this is spam.
spam_report	Spam Assassin's spam report.
attachments	Number of attachments included in email.
attachmentX	These are file upload names, where N is the total number of attachments. For example, if the number of attachments is 0, there will be no attachment files. If the number of attachments is 3, parameters attachment1, attachment2, and attachment3 will have file uploads. TNEF files (winmail.dat) will be extracted and have any attachments posted.

Parse Webhook Technical Overview *continued*

Access Attachments

The possibilities are many just with the textual component of email. By providing access to the attachments in an email message, the Parse Webhook becomes even more powerful. Each message can be up to 20 MB in size, including the headers, message and attachments.

Attachments are posted to the webhook as multipart MIME data. Your application can access attachments the same way as file uploads from a form.

For example, the PHP code to save every attachment to the /tmp directory could look like this:

```
foreach ($_FILES as $key => $file)
{
    move_uploaded_file($file['tmp_name'], "/tmp/".$file['name']);
}
```

The JSON posted to the webhook also contains some meta-data about the attachments. In addition to the number of files, each attachment has its own collection with the filename and other details. For example, here's what the first attachment might look like:

```
"attachment1":{

"originalFilename":"filename.jpg", "path":"/tmp/2-w6psz.jpg",

"headers":{"content-disposition":"form-data; name=\"attachment1\"; filename=\"filename.
JPG\"","content-type":"image/jpeg"},

"size":80180,"name":"filename.jpg","type":"image/jpeg"}

Additional attachments would use attachment2, attachment3 and so on.
```

Parse Webhook: Interaction and Collection

Much like sending email through SendGrid, the Parse Webhook is a blank slate for you to add your creativity. The mundane details of separating out the sections of an email and routing an incoming message to your script are done for you. The rest requires some ingenuity.

The common use cases, case studies, developer evangelist examples and technical overview should give you somewhere to start. Also consider situations where you're interacting with your users or collecting data within your applications. When all the Parse Webhook examples are boiled down, those are the two most common ways it is used.

Frequent tasks can be made more productive if your users never have to leave their inbox. Make your email interactive and collect insights from users with a simple email.

Get Started with SendGrid

Learn More

Read Our Customer Success Stories

Sign Up

About SendGrid

SendGrid helps you focus on your business without the cost and complexity of owning and maintaining an email infrastructure. We manage all of the technical details, from scaling infrastructure, to ISP outreach and reputation monitoring, to whitelist services and real-time analytics. We offer world-class deliverability expertise to make sure your emails get delivered, handle ISP monitoring, DKIM, domain keys, SPF, feedback loops, whitelabeling, link customization, and more. To learn more, visit www.sendgrid.com.