

SendGrid is a cloud-based email delivery service that assists businesses with email delivery. And that probably is the most boring definition ever, considering the powerhouse of tools that SendGrid provides.

With SendGrid you don't just get to send emails, you get to monitor link clicks, un-subscription rates, geographical tracking and a lot more. In this tutorial I'll walk you through building your own email sending service on Rails with SendGrid.

The first thought that probably pops up into your mind is why SendGrid? Believe me, you'll figure out for yourself by the end of this tutorial.

Let's first a Rails project via command line. I'll be using Rails 5 here.

```
rails new sendemail
```

(How *creative of a name* is sendemail though?)

We would like for users to sign up and then maintain a personal profile of all the emails they send. So, we'll be adding the [Devise](#) gem here. I won't be covering how to add Gems and customize Devise in this tutorial, but here's a few [sources that cover](#) it.

We'll then create a scaffold called `email`. The model attributes will be the `receiver` field, `subject` field and `body` field. Everything necessary to create an email! Oh, we'll also be adding a `user_id` mapping column, which will be used to query emails belonging to an user.

```
rails g scaffold email user_id:integer:index receiver:string subject:string body:text
```

Here's where the fun starts.

To send emails from your Rails application, you need to,

1. Add email provider credentials to your development.rb (production.rb if in production environment). I'll be using Gmail here.

ProTip: If you've enabled two factor authentication on your account, you need to create an [app password](#) to bypass it.

It'll look something like this,

```
config.action_mailer.smtp_settings = {  
  :address => 'smtp.gmail.com',  
  :port => 587,  
  :user_name => 'email@gmail.com',  
  :password => 'frsghrjdyquftlsh',  
}
```

```
:authentication => :plain,  
:enable_starttls_auto => true  
}
```

1. Add this to your `app/mailers/application_mailer.rb`. This will send the email.

```
def send_it(email)  
  @email = email  
  
  mail(  
    from: email.user.email,  
    to: email.receiver,  
    subject: email.subject  
  )  
end
```

_Notice that I have specified the from address as well. Also, I have NOT provided a default *from* in *ApplicationMailer*.

You'll also have to create a `app/views/application_mailer/send_it.html.erb` and add this to it,

```
<%= @email.body %>
```

This prints the body of the email into what is being sent out.

1. I've also added the method to trigger email sending to our model. Here's how it looks,

```
class Email < ApplicationRecord  
  belongs_to :user  
  validates_format_of :receiver, with: /\A(?:[^\s]+)@((?:[-a-z0-9]+\.)+[a-z]{2,})\z/i  
  
  after_create :deliver  
  after_update :deliver  
  
  def deliver  
    ApplicationMailer.send_it(self).deliver_now  
  end  
end
```

I've created simple `after_create` and `after_update` callbacks that send the email.

Let's test and see if this works.

I'll create an user by signing up with `swaathi @ skcript` email address. And then send an email to `swaathikakarla @ gmail`.

New Email

Receiver

swaathikakarla@gmail.com

Subject

SendEmail Test

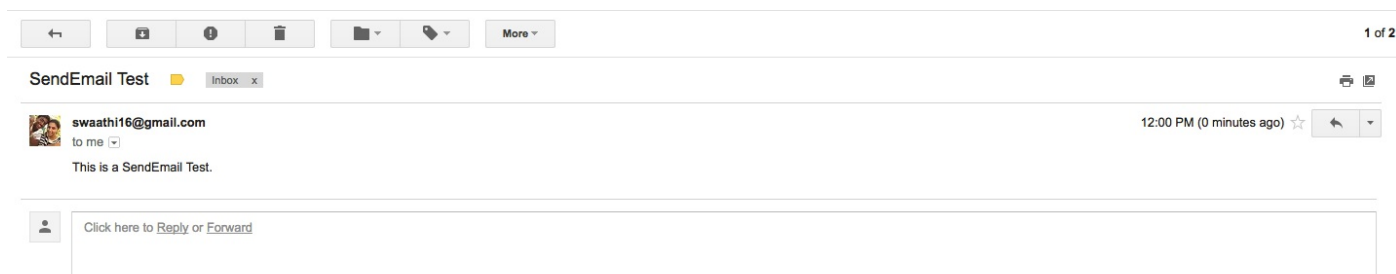
Body

This is a SendEmail Test.

Create Email

[Back](#)

Let's see what happens.



Hey! It got sent from `swaathi16 @ gmail` . But, the `from` section was set to `swaathi @ skcript` . Why did this happen?

It's because of the credentials in `development.rb` . We added Gmail credentials belonging to `swaathi16 @ gmail` . So, no matter what you over-ride it'll be sent from the email that's tied to the credentials.

This works perfect when you don't want someone masking your identity. But often sucks when you just want to send an email from another address, with no way of tracking incoming emails. But, with SendGrid you can send emails as other addresses - but it's not as scary as it sounds - it'll be appended with an 'via Domain' text. And, any replies to the email will automatically be send to the address itself.

Let's see how that works!

1. Sing up on SendGrid Headover to the signup page of [SendGrid](#), and to test things out select the

FreeTrial. This allows you to send 40,000 emails per day for 30 days.



Trial 40K \$0.00 / mo ▾

1. Head over to the Whitelabels section After that visit the Settings section in the sidebar and then click on the [Whitelabels link](#) from the dropdown.
2. Add a Domain Click on the 'Add Domain' button and fill in the form. You need to do this so that users who receive an email that your app sends will be shown the location of origin. So, even if an user on your app has xxx@yyy.com, it'll get sent from this email, but will also include a [text that says 'via DomainName'](<https://support.google.com/mail/answer/1311182>).

You'll see "via" and a website name next to the sender's name if the domain it was sent from doesn't match the domain in the "From:" address. For example, you got an email from john.smith@gmail.com, but it could've been sent through a social networking site and not Gmail.

Add New Domain Whitelabel

Adding a whitelabel allows you to send through your own custom domains instead of SendGrid's default settings. This will mask the header information of your emails with your data--not ours--and will improve your email deliverability.

Subdomain Domain

☒ DEFAULT WHITELABEL

Designating this whitelabel as your default will mask click and open-tracking links to this domain rather than a SendGrid domain. You can only have one default whitelabel at a time. Selecting this option will update any existing default whitelabels belonging to you.

AUTOMATED SECURITY

OFF ON

Allow SendGrid to automate your DKIM security and management.

Automating security allows us to redirect ISPs to SendGrid to check DNS records that follow strict security protocols and are custom to your account.

Due to a character limit on TXT records, we are only able to create a custom SPF record for users with up to 11 IP addresses. This will not affect deliverability.

Cancel

Save

Feedback

In the form you'll have to enter a subdomain and a domain you'd like to send emails through. I suggest you to create a new subdomain, so that you don't run into any weird conflicts. In fact, it shouldn't even exist at this stage.

1. Add to your CNAMEs registry Once you add a domain you'll be taken to a page that looks like this,

← emails.swaathi.com Domain

Remove Default Status

VALID	DEFAULT	SUBDOMAIN	DOMAIN	USER ASSIGNED
✗	✓	emails	swaathi.com	swaathi (Your Account)

DNS Record

information for sending your email with your whitelabel. Copy the following DNS Records into your hosting provider. If you have questions please refer to our documentation.

Validate Record

VALID	TYPE	HOST	DATA
✗	CNAME	emails.swaathi.com	u5857606.wf107.sendgrid.net
✗	CNAME	s1_domainkey.swaathi.com	s1.domainkey.u5857606.wf107.sendgrid.net
✗	CNAME	s2_domainkey.swaathi.com	s2.domainkey.u5857606.wf107.sendgrid.net

DELETE WHITELABEL

Deleting your whitelabel will restore SendGrid as sender.

Delete

Feedback

All you have to do is navigate to your hosting (like GoDaddy) or CDN (like Cloudflare) provider, whichever manages your domain. You can then take the three subdomains SendGrid gives you and map it to your website.

It should take about a minute or more to reflect the subdomain changes on our DNS.

Then headback to your SendGrid page and click on the 'Validate Record' button, and it should look like this now,

← emails.swaathi.com Domain

Remove Default Status

VALID	DEFAULT	SUBDOMAIN	DOMAIN	USER ASSIGNED
✓	✓	emails	swaathi.com	swaathi (Your Account)

DNS Record

information for sending your email with your whitelabel. Copy the following DNS Records into your hosting provider. If you have questions please refer to our documentation.

Validate Record

VALID	TYPE	HOST	DATA
✓	CNAME	emails.swaathi.com	u5857606.wf107.sendgrid.net
✓	CNAME	s1_domainkey.swaathi.com	s1.domainkey.u5857606.wf107.sendgrid.net
✓	CNAME	s2_domainkey.swaathi.com	s2.domainkey.u5857606.wf107.sendgrid.net

DELETE WHITELABEL

Deleting your whitelabel will restore SendGrid as sender.

Delete

Feedback

3 tick marks! Yay!

If not, wait for awhile for your DNS to get updated and try again.

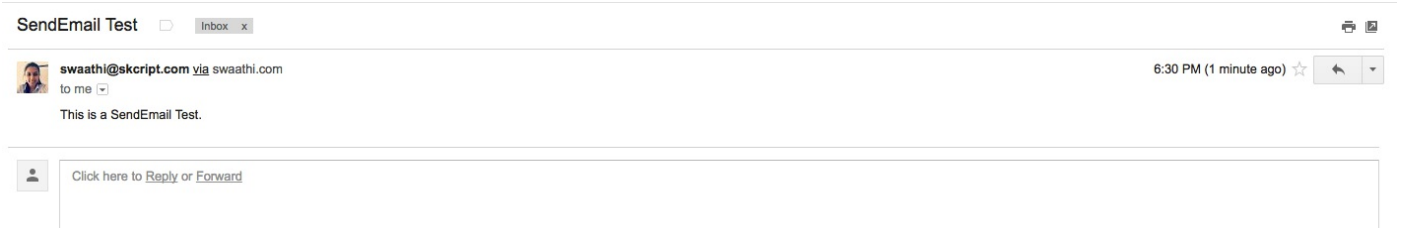
1. Add SendGrid credentials to your app We're almost there. All you have to do now is add SendGrid credentials to your `development.rb` (or `production.rb`). It will look something like this,

```
config.action_mailer.smtp_settings = {
  :user_name => 'your-sendgrid-username',
```

```
:password => 'your-sendgrid-password!',  
:domain => 'your-sendgrid-domain',  
:address => 'smtp.sendgrid.net',  
:port => 587,  
:authentication => :plain,  
:enable_starttls_auto => true  
}
```

And that's it! 5 simple steps.

Let's try it out on the app and see how it looks now.



See the via? That's it!