

Mailgun

When building a website, for example a landing page for what is to come, one vital thing to do is capture the details of people that might be interested in what you are going to launch.

Most importantly, you want to capture their email address so you can communicate things to them. In this session we will learn how to capture a user's email address (and any other details for that matter) and programmatically send them an email to confirm that they have signed up to your awesome website's newsletter!

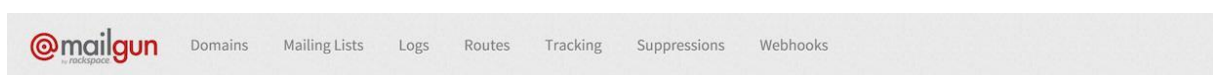
This will be an interactive session based on what we did in [Session 3](#). If you look at what we did during that session under the **GET & POST requests** section, you will see that we already learnt how to capture a user's name and email and print it out on our terminal. Now we are going to use that data to send an email.

We will be using Mailgun, an email infrastructure service - or in simpler terms, a service that enables you to programmatically send emails. We will be using a library called **Pony** to send emails via Mailgun's *SMTP* (a protocol to send and receive email, much like HTTP is a protocol to send/receive web pages) service.

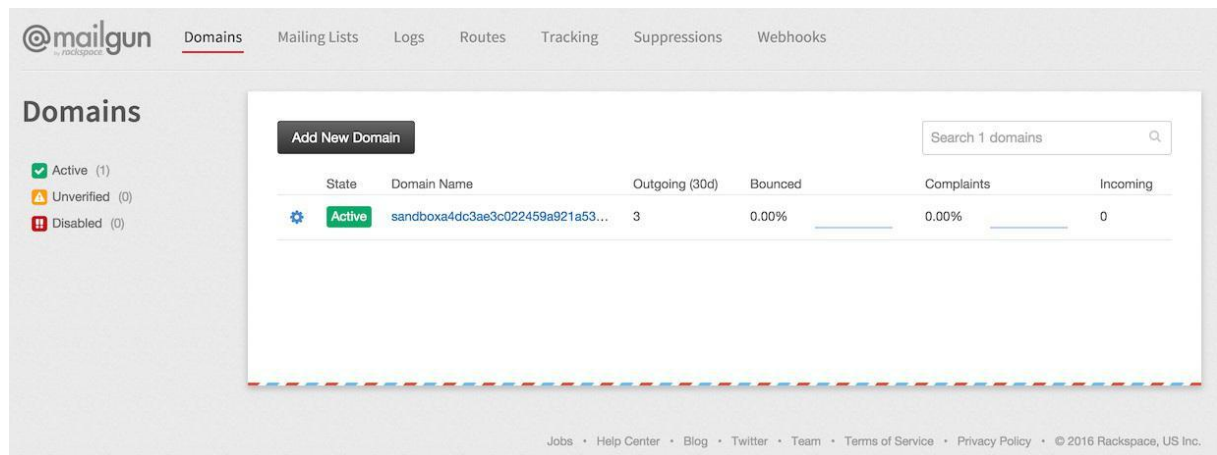
You should all already have a Mailgun account, as this was your homework from last session. If not, go ahead and create an account. You will need to confirm your email (by clicking on a link in an email Mailgun sends you) and enter a verification code sent to you via text - this is Mailgun's way to safeguard against spammers creating accounts programmatically to send spam email.

Using Mailgun

Once you log into your Mailgun account, click on the **Domains** link at the top of the page:



You should then see a page that looks like this:



By default, Mailgun gives us a sandbox domain, which can be used without a custom verified domain. This domain can send up to 300 emails/day, which should be more than enough!

Make sure that you have a green **Active** tag on your page - contact your instructor if you don't.

In order to programmatically send emails, we need to get our SMTP credentials, so go ahead and click on the link under the **Domain name** header that looks like **sandboxa4dc3ae3c022459a921a53...** (your domain name will be different). You should now see a page that looks like this:

@mailgun [Domains](#) [Mailing Lists](#) [Logs](#) [Routes](#) [Tracking](#) [Suppressions](#) [Webhooks](#)

[Domains](#) /sandboxa4dc3ae3c022459a921a5320a13af6f0.mailgun.org

sandboxa4dc3ae3c022459a921a5320a13af6f0.mailgun.org

Domain Information

| | |
|--------------------|--|
| State | Active |
| IP Address | 209.61.151.224 |
| SMTP Hostname | smtp.mailgun.org |
| Default SMTP Login | postmaster@sandboxa4dc3ae3c022459a921a5320a13af6f0.mailgun.org |
| Default Password | d5aa44fc23f9b484e2a52860558a16fde • Manage SMTP credentials |
| API Base URL | https://api.mailgun.net/v3/sandboxa4dc3ae3c022459a921a5320a13af6f0.mailgun.org |
| API Key | key-16a2355e31caf9c52dde1e3053231bca |

We will need the **Default SMTP Login** and **Default Password** in a bit - these are our credentials that we will use to access Mailgun's SMTP service - so make a note of them.

Pony

Firstly, we need to install the **Pony** gem - enter the following in your terminal:

```
gem install pony
```

(as with Sinatra, you may  need to add a **sudo** to the front of that command).

We now need to tell Pony about the SMTP server we want to use, and enter our SMTP credentials. Create a file called **email.rb** and enter the following code:

```
require 'pony'

Pony.options =
  { :via =>
    'smtp',
    :via_options
    => {

      :address =>
        'smtp.mailgun.org', :port
        => '587',
      :enable_starttls_auto =>
        true, :authentication
        => :plain,

      # This is the Default SMTP Login from earlier:

      :user_name => 'postmaster@your-sandbox-domain.mailgun.org',

      # This is your Default Password:

      :password => 'your password'

    }
  }
}
```

Next up, we want to actually send an email, so we need to add the following to our `email.rb` file:

```
message = {  
  
  :from => 'hello@example.com',  
  
  :to => 'Recipient Name Goes Here  
<some.email@example.com>', :subject =>  
  'Welcome!',  
  
  :body => 'Thanks for signing up to our awesome newsletter!'  
  
}  
  
Pony.mail(message)
```

We can now send an email from our terminal by running the file with Ruby:

```
ruby email.rb
```

This example is completely independent from a Sinatra app, so what you should do now is figure out how to apply the above code in the context of your Sinatra app that we created over the past few sessions.

Ensure that you are sending an email to a valid email address, set your own email address as the **from** and use the email and name submitted via the form that we were previously printing.

-
-

Task

Adapt your Sinatra code so that your contact form sends the user an email when they submit the form.

Read through the [Pony documentation](#) and figure out what other things you can do with Pony and Mailgun - can you send a HTML-formatted email? Can you add attachments? What about sending an email to multiple recipients (you might have to Google this one)?