

**rvg.me**

# Adding Bootstrap from a CDN to your Rails application

In a [previous post](#), I talked about manually adding Bootstrap 3 files to a new Rails 4 project. What about if we made upgrading easier by loading the files from a CDN instead? Then we could choose between using the latest version of Bootstrap 3, an earlier release or even Bootstrap 2 by just changing a string in a helper. Sound good?

Let's get started by creating a project:

```
rails new bootstrap-cdn
```

Next, created a new file called `layout_helper.rb` in `app/helpers/` and populate it with the contents below. I've chosen to use the latest version of Bootstrap at the time of writing which was `3.0.3`.

```
module LayoutHelper

  def bootstrap_version
    # Choose the Bootstrap version you would like to use from the
    CDN
    "3.0.3"
  end

end
```

Now let's edit `application.html.erb` in `app/views/layouts`. In the head, we want to include the CSS and just before the closing body tag, we add the JavaScript.

```
<!DOCTYPE html>
<html>
<head>
  <title>Loading Bootstrap from a CDN</title>
  <link href="//netdna.bootstrapcdn.com/bootstrap/<%=
bootstrap_version %>/css/bootstrap.min.css" rel="stylesheet">
  <%= stylesheet_link_tag      "application", media: "all", "data-
turbolinks-track" => true %>
  <%= javascript_include_tag "application", "data-turbolinks-
track" => true %>
  <%= csrf_meta_tags %>
</head>
<body>
  <div class="container">
    <%= yield %>
  </div>
  <script src="//netdna.bootstrapcdn.com/bootstrap/<%=
bootstrap_version %>/js/bootstrap.min.js"></script>
</body>
</html>
```

That's all it takes to get up and running quickly without a gem. It also happens to be the simplest upgrade method that I can think of for down the road. For those of you looking to expand the application slightly for testing purposes, then read on.

Open your Gemfile and add [high\\_voltage](#).

```
gem 'high_voltage', '~> 2.1.0'
```

Edit your `routes.rb` found in `config/`.

```
Rails.application.routes.draw do

  get 'pages/home' => 'high_voltage/pages#show', id: 'home'

end
```

For High Voltage to function correctly, we need to create an initializer called `high_voltage.rb` in the `config/initializers/` directory and populate it with:

```
# config/initializers/high_voltage.rb
HighVoltage.configure do |config|
  config.home_page = 'home'
end
```

In your Terminal, create the home page and directory that we referenced above.

```
$ mkdir app/views/pages
$ touch app/views/pages/home.html.erb
```

Now let's get to some fun testing. Open the file `home.html.erb` in `app/views/pages`. We'll create a number of buttons to make sure the Glyphicons are loaded along with a Javascript tooltip. Here's some code to get you started:

```
<h2>Some Glyphicons</h2>

<%= button_tag(type: 'button', class: 'btn btn-default btn-lg' )
do %>
  Star <%= content_tag(:span, '', class: 'glyphicon glyphicon-
star') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-default' ) do %>
  Star <%= content_tag(:span, '', class: 'glyphicon glyphicon-
```

```
      Star <%= content_tag(:span, '', class: 'glyphicon glyphicon-
star') %>
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-default btn-sm' )
do %>
      Star <%= content_tag(:span, '', class: 'glyphicon glyphicon-
star') %>
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-default btn-xs' )
do %>
      Star <%= content_tag(:span, '', class: 'glyphicon glyphicon-
star') %>
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-primary btn-lg' )
do %>
      <%= content_tag(:span, '', class: 'glyphicon glyphicon-user')
%> User
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-primary' ) do %>
      <%= content_tag(:span, '', class: 'glyphicon glyphicon-user')
%> User
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-primary btn-sm' )
do %>
      <%= content_tag(:span, '', class: 'glyphicon glyphicon-user')
%> User
    <% end %>

    <%= button_tag(type: 'button', class: 'btn btn-primary btn-xs' )
do %>
      <%= content_tag(:span, '', class: 'glyphicon glyphicon-user')
%> User
    <% end %>
```

```

    <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-user' )
  %> User
<% end %>

<br><br>

<%= button_tag(type: 'button', class: 'btn btn-default btn-lg' )
do %>
  <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-default' ) do %>
  <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-default btn-sm' )
do %>
  <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-default btn-xs' )
do %>
  <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-primary btn-lg' )
do %>
  <%= content_tag(:span, ' ', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

```

```

<%= button_tag(type: 'button', class: 'btn btn-primary' ) do %>

```

```
<%= button_tag(type: 'button', class: 'btn btn-primary' ) do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-primary btn-sm' )
do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-primary btn-xs' )
do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-warning btn-lg' )
do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-warning' ) do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-warning btn-sm' )
do %>
  <%= content_tag(:span, '', class: 'glyphicon glyphicon-align-
justify') %>
<% end %>

<%= button_tag(type: 'button', class: 'btn btn-warning btn-xs' )
do %>
```

```
<%= content_tag(:span, '', class: 'glyphicon glyphicon-align-justify') %>
<% end %>

<h2>A Javascript Tooltip</h2>

<%= button_tag('Tooltip on the right', type: 'button', class: 'btn btn-default', id: 'tooltip', data: {toggle: 'tooltip', placement: 'right'}, :data-original-title => 'Tooltip on the right' )%>
```

One last to do, be sure to edit `application.js` in `app/assets/javascripts` so that our tooltip works.

```
$(document).ready(function(){
  $("#tooltip").tooltip();
});
```

That's it and don't forget to run `bundle install` at the Terminal. We added a gem to the Gemfile. Enjoy!

A copy of this application is on Github and you can [download it here](#).

If you have any suggestions or feedback, please let me know in the comments, via [email](#) or on [Twitter](#).

Share this:



Related

[Adding a Bootstrap 3 layout to a Rails 4 project](#)

November 15, 2013

In "bootstrap"

### [Using Bootstrap 3 with Rails 4](#)

November 8, 2013

In "bootstrap"

### [Rails Composer](#)

April 9, 2014

In "ruby-on-rails"



Richard / January 8, 2014 / bootstrap, ruby-on-rails



**2 Comments** **rvg.me****1** **Login** ▾ **Recommend**  **Share****Sort by Best** ▾

Join the discussion

**Stephen Giles** 2 years ago

Would I need to remove

//= require jquery

or anything else from my application.js?

I'm having an issue with the dropdowns and I'm thinking I may be loading jQuery twice?

^ | ▾  **Reply**  **Share** **Richard van Geilswyk** Mod  **Stephen Giles** 2 years ago

No you shouldn't need to remove that line.

If you think you are loading jQuery twice, could it be possible that you added jQuery to your project? Take a look in your application.html.erb view (path: app > views > layouts > application.html.erb)? Look either in the head or just before the body close tag for some links to javascript. In the tutorial above, I'm only adding links to Bootstrap CSS and JS files.

Did the sample project on GitHub work with your dropdown? I just cloned the GitHub repository locally to test. Once downloaded, I opened the Gemfile and removed the rails version. Ran Bundle Install then Bundle Update. I also needed to do a onetime Spring update (bundle exec spring binstub --all) to get the sample project to load. Then I opened app > views > pages > home.html.erb and pasted some Dropdown code (<http://getbootstrap.com/com...> from the folks at Bootstrap and it worked for me. I didn't need to edit the application.js or application.html.erb files.

Good luck!

^ | ▾  **Reply**  **Share** **ALSO ON RVG.ME****Using Bootstrap 3 with Rails 4**

43 comments • 3 years ago •

**First Post**

1 comment • 3 years ago •