Mark Prucha

PDF Gem

There are 4 widely-used gems for converting files to PDF format. These are: PDFKit, wkhtmltopdf, Wicked PDF (which builds off of wkhtmltopdf), and Prawn. The first three are all direct HTML to PDF generators. This means that to create a PDF page, the programmer in his or her code must pass the HTML page path (view path) as an argument in one of the gem's functions. For example, using PDFKit, to create a PDF from an HTML file a programmer would write:

```
kit = PDFKit.new(File.new('/path/to/html'))
```

Example from: http://www.sitepoint.com/pdf-generation-rails/

After installing the gem, the programmer would write the above code in a controller class, and when called, the pdf document would be created on the server. The pdf document could then be linked to in subsequent code so the client could access it.

The main problem that I found with the direct HTML to PDF generator gems is the programmer cannot format how much content is displayed on a PDF page. So for example, if a table is being used on an HTML page to show data, when that page is converted to a pdf page, the table may run on to another page in the pdf document, causing a messy document. The programmer would have no control over this to fix it, unless he or she changes the actual HTML document.

Therefore, for what we're doing, I think it's best we use the Prawn gem. The Prawn gem lets you create the PDF document from scratch, using domain-specific language developed for Prawn. While this may require a little more work from us, it gives us more control in what's rendered and how it's rendered on the PDF document. I got Prawn running on my computer, and it was a relatively easy process. The language to create the document is also easy to understand.

So, for example, to create a Hello World PDF document you install the gem Prawn, put a line of configuration code in the `config/initializers/mime_types.rb` file, and you're ready to start

rendering pdf documents. Then in a controller method (I created a 'hello' controller and defined index), I wrote:
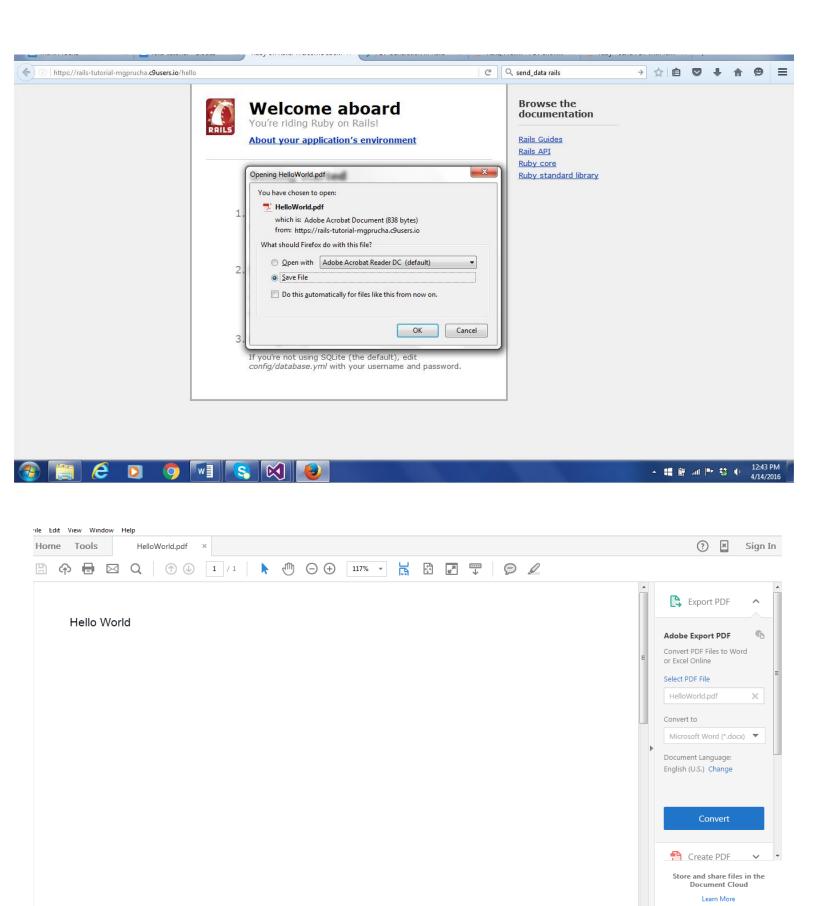
```
def index
  pdf = Prawn::Document.new
  pdf.text "Hello World"
  send_data pdf.render, :filename => "HelloWorld.pdf", :type =>
       "application/pdf"
end
```

The variable 'pdf' holds a new PDF document at the line: `pdf = Prawn::Document.new`. The 'text' method lets us put whatever data we want in the document. For this example, I just put in a string, "Hello World". Here you can also put in any and all data held by your Ruby variables using normal Ruby code to access them. Finally, the last line sends the newly created document to the client using send_data. It also renders the document through the function pdf.render. The other parameters are "filename" to name the file and "type" to tell the renderer what type of document you want.

I should note that render was used in this example. There is also a method called render_file which creates and stores the file on the server. We probably don't want this for our project so a simple 'render' is the way to go.

Other Prawn methods include: "pdf.table" which takes a 2-D array as an argument and renders its contents in a table format, and pdf.move_up / pdf.move_down which let you navigate to where in the document you want to write to. There are also methods to change fonts and sizes, draw shapes, add images etc.

There is a manual for Prawn available at: http://prawnpdf.org/manual.pdf

So when we run the server, and append "/hello" (from the example above) to the url, we get:

https://rails-tutorial-mgprucha.c9users.io/hello

send_data rails

Welcome aboard
You're riding Ruby on Rails!

About your application's environment

Browse the
documentation

Rails Guides
Rails API
Ruby core
Ruby standard library

Opening HelloWorld.pdf

You have chosen to open:

HelloWorld.pdf

which is: Adobe Acrobat Document (838 bytes)
from: https://rails-tutorial-mgprucha.c9users.io

What should Firefox do with this file?

Open with   Adobe Acrobat Reader DC (default)

Save File

Do this automatically for files like this from now on.

OK   Cancel

If you're not using SQLite (the default), edit
config/database.yml with your username and password.

12:43 PM
4/14/2016

File   Edit   View   Window   Help

Home   Tools   HelloWorld.pdf   ×

Sign In

1   / 1   117%

Hello World

Export PDF

Adobe Export PDF

Convert PDF Files to Word
or Excel Online

Select PDF File

HelloWorld.pdf   ×

Convert to

Microsoft Word (*.docx)

Document Language:
English (U.S.) Change

Convert

Create PDF

Store and share files in the
Document Cloud

Learn More

12:46 PM
4/14/2016