WEASYPRINT

You Are My Only Hope!

What is WeasyPrint?

> WeasyPrint has been created and is maintained by Kozea community, a group of people designing, maintaining and developing free and open source software.
>
> You want to share some kind words with the core team? You would be interested in training courses about CSS for print? You would love new features but do not have the time or the development resources? Send us a message!
>
> WeasyPrint is free and open source software that can be easily plugged to your applications and websites.
>
> *https://weasyprint.org/about/*

Features

HTML/CSS Supported

Everyone knows enough HTML and CSS to generate simple report, no need to learn a whole new templating language

Productivity

Official samples are gorges and covers most layout use cases which make you more productive!

One click to debug

Weasyprint gives developer control to debug. Render/ Debug on Webpage shows you margins and paddings.

Flexibility

WeasyPrint can be used either as a standalone program or as a Python library

Documentation

When learning a new tool, the most challenging thing is hard to find how to use and customize. Weasyprint has very well documents.

Actively Maintained

It is actively maintained, WeasyPrint is evolving and you can get helped either from StackOverflow or GitHub issues.

Why I choose WeasyPrint?

Our story

A couple of months ago, I got a project and was told to build PDF reports based on given JSON data sources. Briefly, the report contains text messages, charts and tables. Before this project, I had more than 7 years back-end development background and 3 years web & mobile development background but still had no clue how to build beautiful reports in very limited time.

- 1. How to parse JSON sources and fetch required data?
- 2. What tech stack to choose for the project?
- 3. Should we build the report on client side or server side?

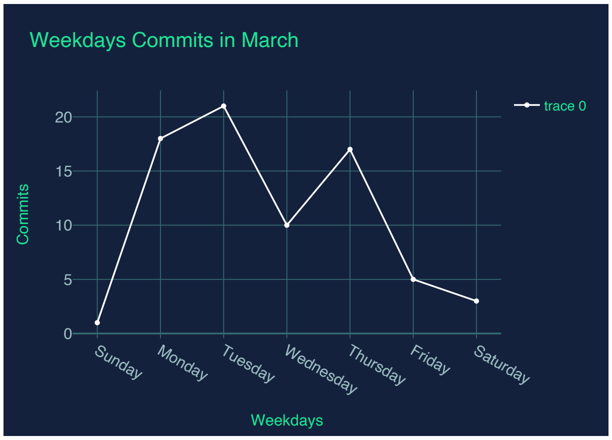
I don't want to make a rush decision even the deadline is quite urgent, I believed that the right tool will do the work fast and efficient. I spent a lot of time doing research on Google, Reddit, StackOverflow, GitHub. Finally found a couple potential tools like jsPDF, ReportLab and pyPDF. I have tried each of them in a week or two and built mockup report but I wasn't satisfied with the crude results.

The first tool I decided to not use was jsPDF. The first reason was the charts would be blurry when rendering PDF, my question can be found on SO: *Canvas charts is blurry when download as PDF* . Another reason was it was time consuming to parse the raw JSON source and it'd better to parse data and generate report on server side.

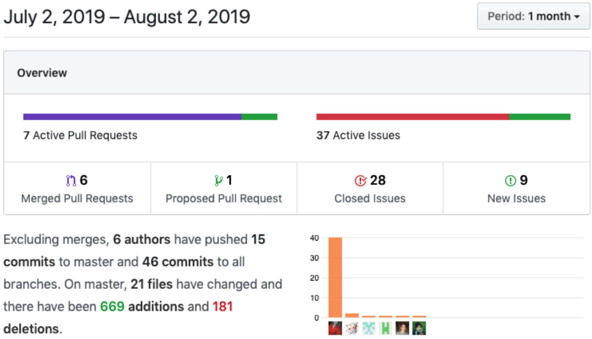
Other reasons to give up other tools were either we couldn't afford, we will build more than two thusand reports per year and the price would be crazy if it was over the limit; the learning curve< is another cause, learn a new templating language or rules to build layout was challenging and takes too much time and our sales team can't wait that long.

One month later after I did research and mockup, I was back to the beginning point. To be honest, I was a little bit frustrated because I still don't know what tool to use but the good side is I know what not to use for project under the circumstance.

Next week, I kept doing research and found *Creating PDF Reports with Pandas, Jinja and WeasyPrint* When I saw the samples from Weasyprint website, I was shocked and impressed by the gorges report. Weasyprint draw my attention and I started reading weasyprint docs and was happy that it uses HTML & CSS which I have enough knowledge. I can't wait to follow the samples to build a mockup in just half a day and showed it to my boss. I got a big WOW and I know we have found the right tool! Thank you Weasyprint!



When I choose one open source, the first thing I am checking is how active the project is. I have selected weekdays commit in March, 2019. You can see the project commits is frequent which means it is actively maintained.



July 2, 2019 – August 2, 2019

Overview

7 Active Pull Requests

37 Active Issues

6 Merged Pull Requests	1 Proposed Pull Request	28 Closed Issues	9 New Issues

Excluding merges, **6 authors** have pushed **15 commits** to master and **46 commits** to all branches. On master, **21 files** have changed and there have been **669 additions** and **181 deletions**.

The second thing I check is the numbers of open > issues and closed issues. You can see July, 2019 - August 2019 there were 28 issues had been closed, I believe this is a good sign and the project is reliable.

You may observed that the chart looks really good and has very high quality but the captured image is a little bit blurry. This is another thing I want to share: we use Plotly Python generate svg files and insert them into our report and it is clear even we zoom out to 200%.