

CS305 HIS - REPORT

Steve Jones

TABLE OF CONTENTS

[1 - INTRODUCTION](#)

[2 - RESUME](#)

[2.1 - 10 Commandments of the Software Engineering Resume](#)

[2.2 - Resume Resources](#)

[3 - PERSONAL WEBSITE](#)

[Getting Started](#)

[3.1 - Buy a Domain Name](#)

[3.2 - Hosting Your Site](#)

[3.3 - How to Point a Domain on Google Domains to GitHub Pages](#)

[3.3.1 - CNAME File Point to Domain](#)

[3.3.2 - Configure Google Domains](#)

[3.3.3 - Help & Resources](#)

[3.4 - Design & Development](#)

[3.4.1 - Bootstrap](#)

[3.5 - Website Content](#)

[3.5.1 - Home Page](#)

[3.5.2 - Projects](#)

[3.5.3 - Portfolio](#)

[3.5.4 - Resume](#)

[3.5.5 - Contact](#)

[4 - RESOURCES](#)

[Questions?](#)

1 - INTRODUCTION

Computer science is an exciting and unique field because it is constantly changing. Tech companies have been revolutionizing traditional business strategies and one example of this is the hiring process. Companies such as Google and Apple as well as smaller tech startups have changed the way they hire for technical positions. As someone who is passionate about the tech industry and the hiring process I researched this topic. This report presents my research in a way that I can share with fellow computer science undergraduates to help them navigate this process.

Many software engineers have a personal website where they showcase themselves. This is an ideal place to send recruiters to in order to display your technical abilities and provide them with any relevant information about yourself. Generally on these sites you will provide information about yourself, contact information, a resume, and a portfolio. This report describes my discoveries of best practices and optimal strategies to build and display a personal website to enhance the owner of the website's chances in the hiring process.

2 - RESUME

2.1 - 10 Commandments of the Software Engineering Resume

The hiring process for software engineering positions can be very competitive, especially when applying to Silicon Valley tech giants such as Google, Facebook, Apple. Here are 10 tips to help your resume get you that interview.

- 1) **One Page Resume** - If you are applying to an entry level or internship position KEEP YOUR RESUME TO ONE PAGE. Unless you have 10+ years of industry experience or years of research publications you should stick to one page. Forcing your resume onto one page will force you to only include the most important pieces, and also cut down parts to be more concise. The person reviewing your resume will likely spend less than 30 seconds looking at it. Make sure the most important parts of your resume stand out.

> WORK EXPERIENCE

TimeTrade
Summer 2015

SOFTWARE ENGINEER

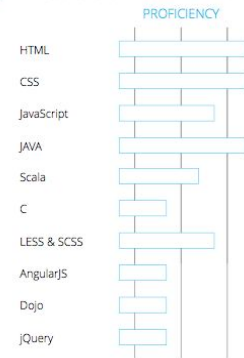
- Designed & developed TimeTrade's new Click To Check In web application
- Researched, designed, & implemented UX improvements to 4 of TimeTrade's apps


- 2) **Short Bullets** - If you have text that is more than a handful of words long, reviewers are much less likely to read it. Separate your descriptions into bullets and then further trim down those bullets to use the least amount of words possible.
- 3) **Resume Template** - Unless you have extensive design experience, don't design your resume from scratch. There are an abundance of great resume templates on the web. However, be careful, don't go overboard in terms of design. Avoid templates that include lots of color and graphics.
- 4) **Accomplishments** - Don't talk about your responsibilities at a company, rather what you accomplished. Show how you impacted that company.
- 5) **Quantify Accomplishments** - Using numbers to quantify your accomplishments makes them more understandable and impressive. Example "Managed software engineers to build mobile app" → "Managed team of 12 software engineers to build an iOS app downloaded by 24,000 users"

PROJECT MANAGER

- Managed team of 10 undergrads to design & build web app for client
- Lead 30+ meetings to create requirement spec, high & low level design docs, and test plan

- 6) **Languages / Technologies** - Many resume reviewers will go straight to this section to see if you have experience with the technologies/languages their company uses. Be careful: if you list a programming language on your resume, you should be confident that you can answer a coding question about it.
- 7) **Include Projects** - Simply put: show that you can actually code. Unlike most professions, in software engineering a GPA cannot accurately reflect your dev skills. Getting good grades doesn't necessarily show you can develop software. It's possible to graduate from college with a computer science degree still not be a skilled developer.

> LANGUAGES & FRAMEWORKS**> PROJECTS**

 Honors Thesis

UX RESEARCH OF IOS VS. ANDROID DESIGN GUIDELINES

- Design and develop iOS & Android app demonstrating respective design guidelines
- Comparison analysis of user testing for each application

- 8) **GitHub** - If you are applying for a software engineering position you should have a GitHub account. If you are very active on GitHub that is a huge plus, but at minimum have an account with at least a couple repositories. This shows you can actually write code and many companies use Git and will expect you to be proficient with it.

**GITHUB**

GitHub.com/Steve-Jones

- 9) **Show Your Passion** - What have you done outside of school or your job? Demonstrate you're passionate enough about your field to do work outside of the classroom or the office. Include stuff you've done in your free time such as projects, open source code you've contributed to, a link to your tech blog, organization's you're a member of, etc.

> MEMBERSHIPS

UMass UX Club (Founder & Chair)
 UMass Association for Computing Machinery
 Hackers of UMass
 UMass iOS Developers
 Colleges Against Cancer
 UMass Entrepreneurship Club
 UX Professionals Association Boston

- 10) **Modify Resume For the Position You Apply To** - If you're applying to a web development position make sure you emphasize your web development skills. Put those languages at the top of your list. Read through the job position posting and modify your resume to target what their looking for.

2.2 - Resume Resources

Feel free to check out my resume: <http://stevejones.io/SteveJones-Resume.pdf>

The best resume resource I have discovered is the Hackathon Hackers “[HH Websites and Resumes](#)” Facebook group. I highly recommend joining this group if you are interested in improving your resume. Demographically, the group consists of mostly software engineers and other computer science related professionals and students. There is also a solid presence of design and UX members.



Members often post resumes to receive constructive feedback. The group is a very positive and helpful community and if you have a question or want feedback on a resume you will likely be very please with the response you get. This also means the group is a great resource for looking at example resumes. Looking at these resumes can give you inspiration and ideas for how to improve your own resume.

In general, Hackathon Hackers is an incredible resource for anyone in a software engineering, product design, product management, or any related field. There is a main Hackathon Hackers group, but there are also nearly 100 Hackathon Hackers (HH) subgroups focused on a more specific topic such as the [HH Websites and Resumes](#) subgroup. Check out the links below:

Hackathon Hackers Facebook Group - <https://www.facebook.com/groups/1487708811477672/>

List of HH Subgroups - <https://github.com/HackathonHackers/groups>

Subgroups Relevant to this:

HH Websites and Resumes - <https://www.facebook.com/groups/1487708811477672/>

HH Internships - <https://www.facebook.com/groups/HHinternships/>

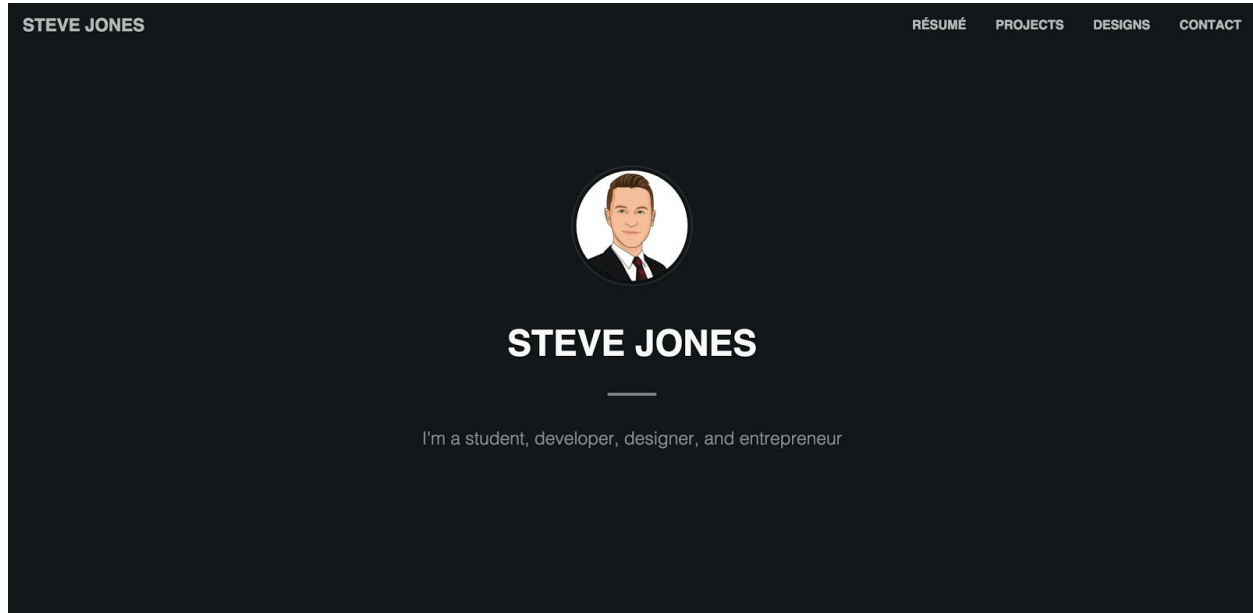
HH Coding Interview Prep - <https://www.facebook.com/groups/722062947858000/>

HH Interview Experiences - <https://www.facebook.com/groups/interviewexper/>

HH Job Listings - <https://www.facebook.com/groups/254805221385780/>

HH: Share Your Projects - <https://www.facebook.com/groups/778907548826382/>

3 - PERSONAL WEBSITE



Having a personal website can help you land a job or internship in many ways. A website can serve as a central location for someone to find any information about you and links to accounts that are helpful for companies to see if you're a good fit for a position.

Having a website shows that you are passionate enough to spend time to design and develop a website. The site itself is a great opportunity to demonstrate your skills. It also gives you a space to showcase projects you've done and portfolio items you have. If you have written articles or blog posts related to your industry including these on your site will show your passion of the industry. Even if you have a Twitter account where you post about tech related topics, this can make a good impression on employers.

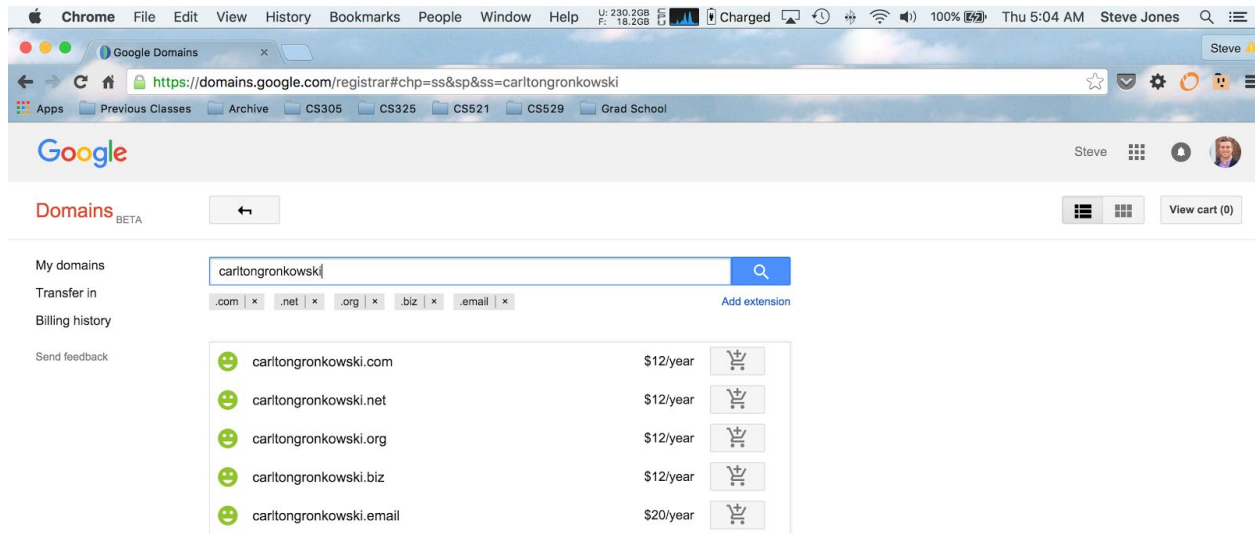
Below I will describe strategies I recommend to create a personal website for yourself.

Getting Started

3.1 - Buy a Domain Name

Having your own domain name is a great way to make your site look more professional. If you must use a third party domain such as `steve-jones.github.io` or `steve-jones.wordpress.com` that's okay, the important part is getting a website up.

Purchase a domain name from Google Domains, Hover, GoDaddy, or many others for ~\$10/year

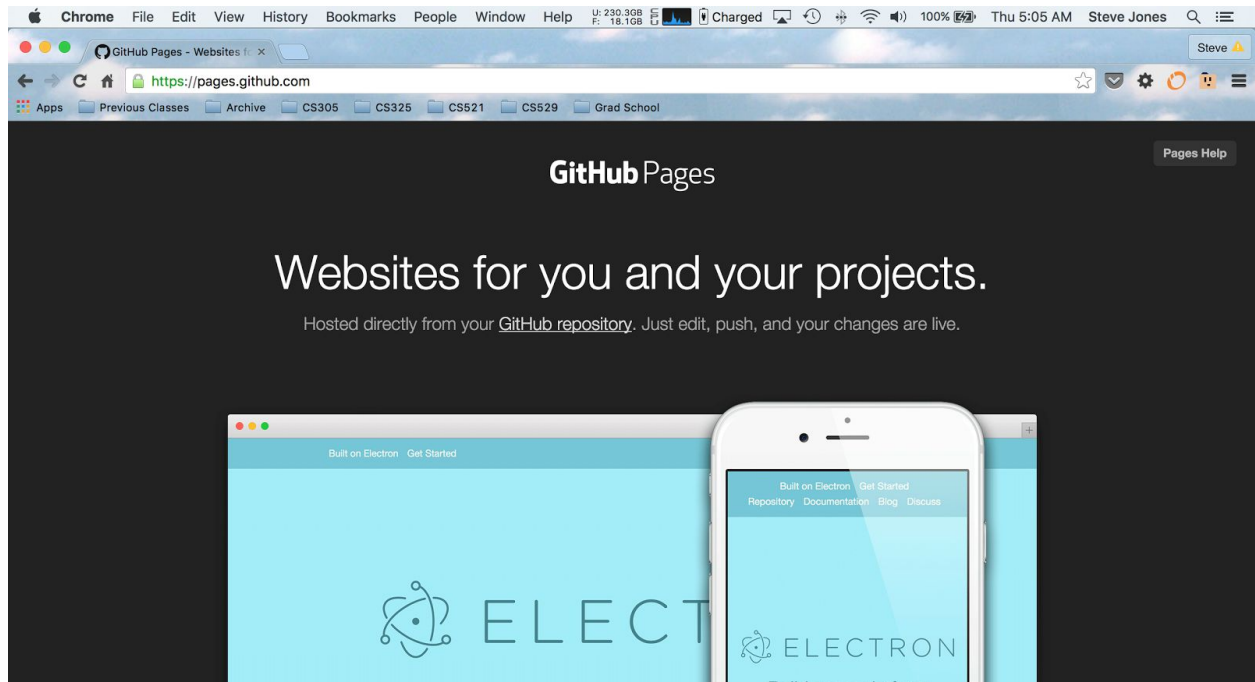


I personally use Google Domains because they make it simple and have great customer service

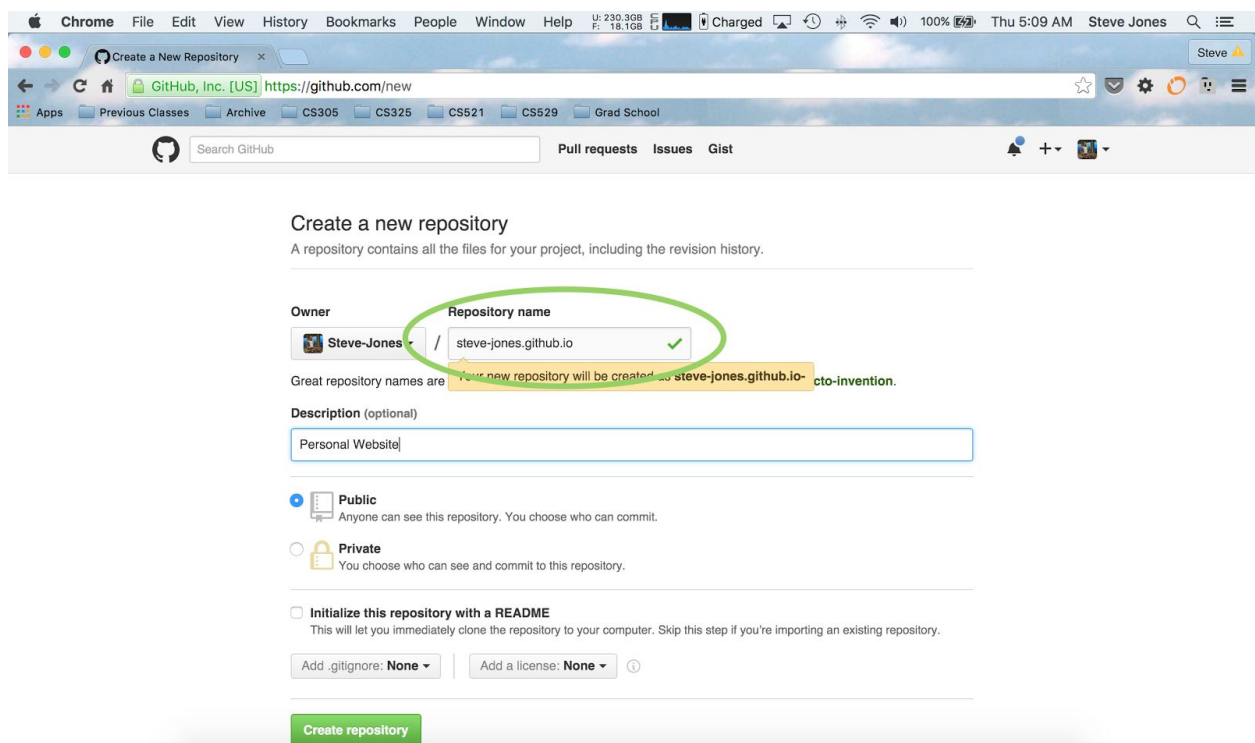
If possible try to get a .com name with your name (`stevejones.com`, `steve-jones.com`, etc.). If you have a common name this will likely not be an option. Check out other [TLDs](#) such as `.me` or `.io` (`.io` is a popular TLD in the tech industry as an abbreviation for input/output). There are about 1000 TLDs, so if you cannot find a common TLD check out some more unique ones. For example, because Steve Jones is such a common name I used the domain `stevenjon.es` as a [domain hack](#) for years because including the `.es` it spells my full name. `.es` is the TLD for Spain (España).

3.2 - Hosting Your Site

One option is to use a web hosting account such as `siteground.com`. This is what most companies would do for their website. However I use GitHub Pages to host my site and it is a much simpler solution, and it's free!



If you are applying for a software engineering position you should already have a GitHub account. Using GitHub pages to host your site is simple because it is hosted directly from your GitHub repository. Just edit, push, and your changes are live. You get one site per GitHub account for free.



Check out this link to get your GitHub Pages site set up: <https://pages.github.com/>

Once you complete that simple setup, your site should be live at <http://yourusername.github.io>

3.3 - How to Point a Domain on Google Domains to GitHub Pages

Once you have your GitHub Pages site set up (<http://yourusername.github.io>)

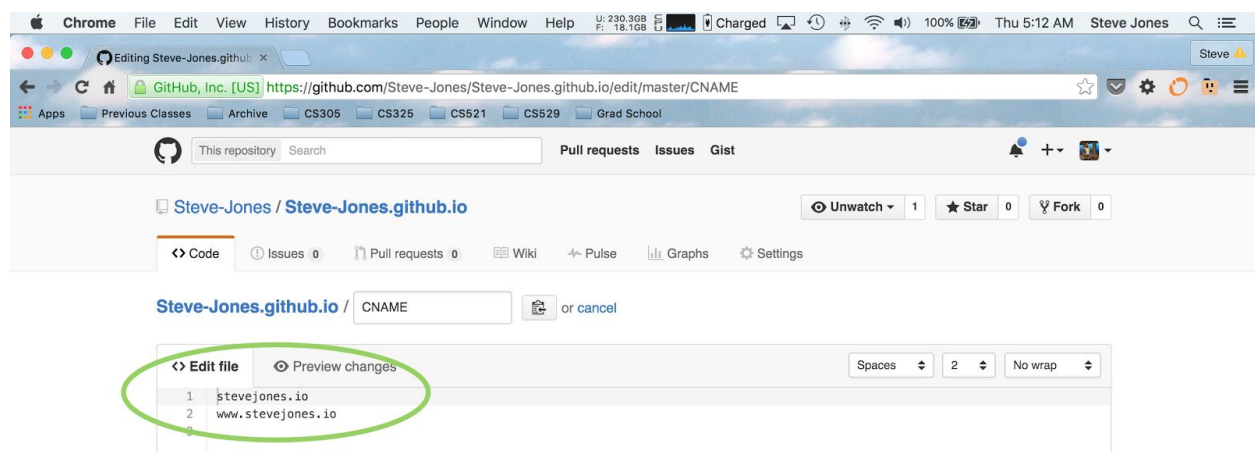
Go to the repository of your GitHub Pages site

3.3.1 - CNAME File Point to Domain

In this repo create a CNAME file in the root directory. Have the file include only these two lines:

`yourdomain.com`

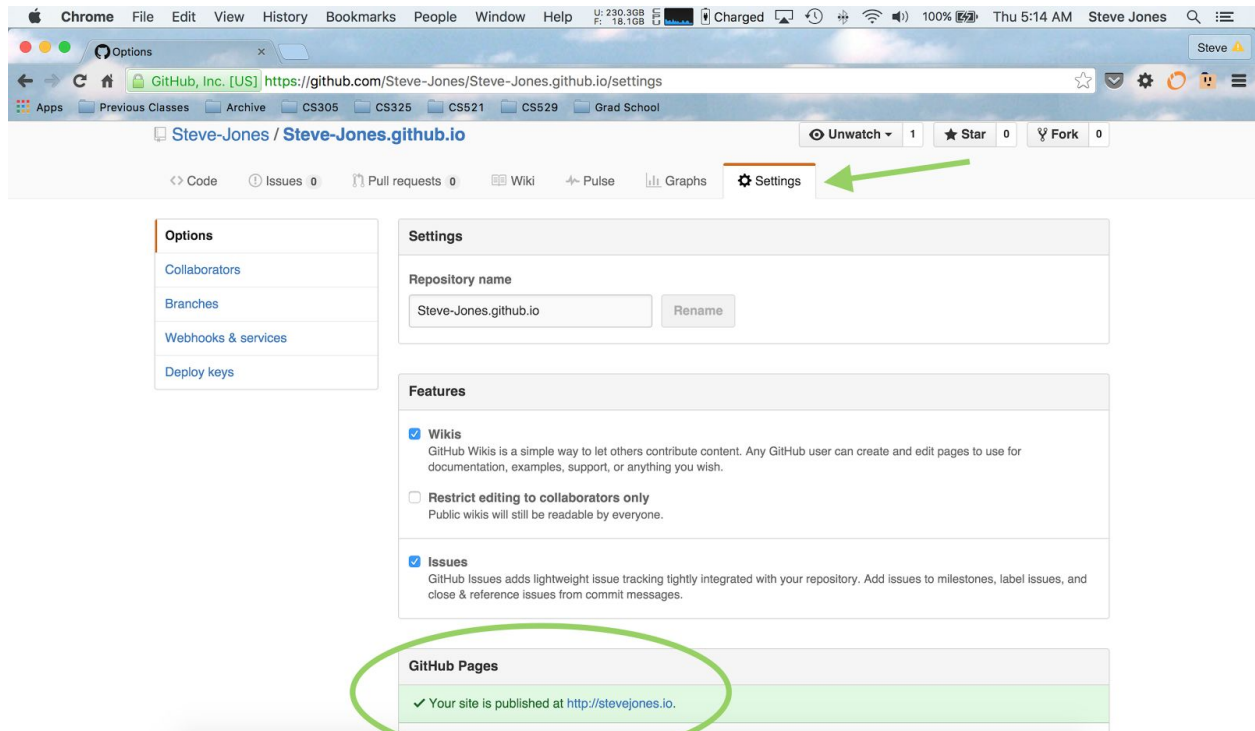
www.yourdomain.com



This CNAME file will now tell GitHub to redirect requests to `yourusername.github.io` → `yourdomain.com`

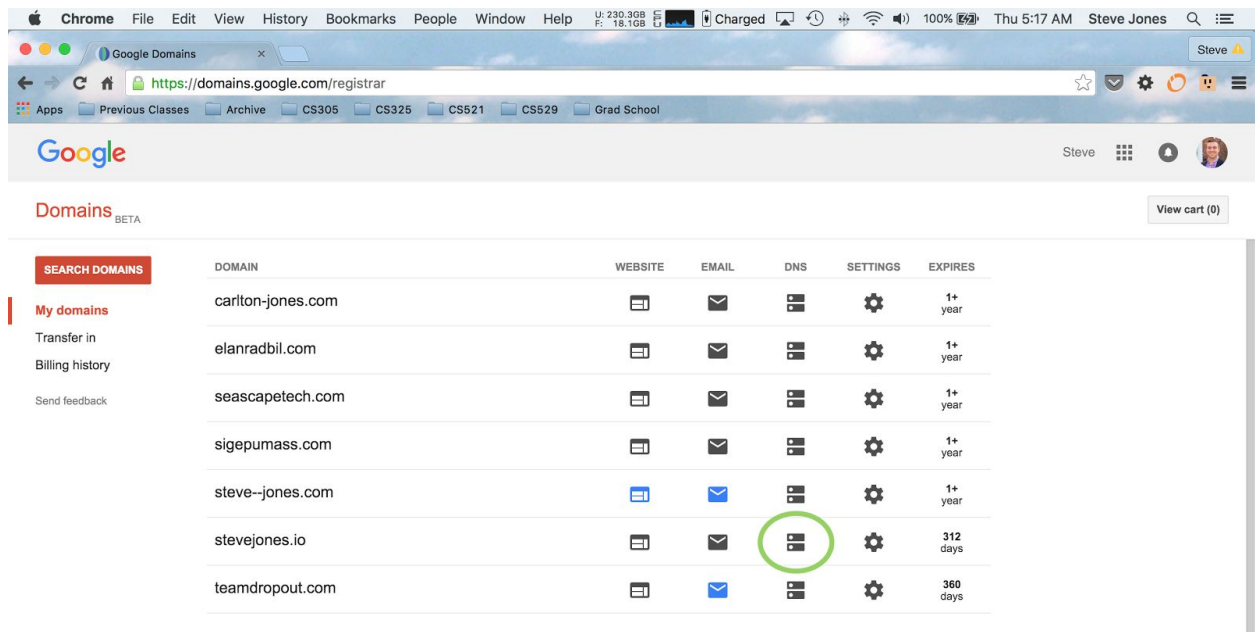
Include both your domain name with and without the `www`. this ensures whether a user includes the `www` or not they will still be directed to your site. If you would like the name of the domain in their URL field to include the `www`. put that line on first in the CNAME file. If you prefer to not have the `www` display in the URL field use the line without it first in the CNAME file.

Check now that if you navigate to `yourusername.github.io` it should redirect you to `yourdomain.com`. You can also check in settings of your repo. On the repo settings page under the GitHub Pages heading the first line should say '✓ Your site is published at <http://yourdomain.com>'

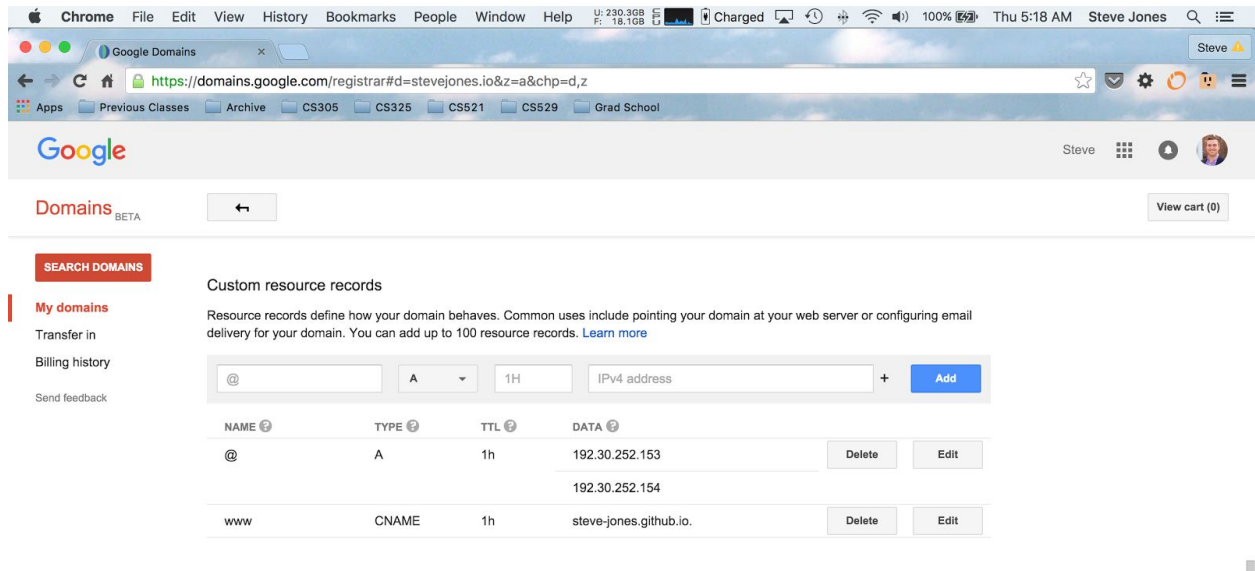


3.3.2 - Configure Google Domains

Go to <https://domains.google.com/registrar> and click the DNS icon to the right of your domain name.



At the bottom of the page under the 'Custom resource records' header you must add 3 custom resource records



Add two records with Name: @ and Type: A

Under Data have these 2 record point to the GitHub IP Addresses 192.30.252.153 and 192.30.252.154

Add one record with Name: www and Type: CNAME

Point this record to your `yourusername.github.io` URL

At this point all you have to do is wait for the DNS records to propagate (this can take 24 to 48 hours)

3.3.3 - Help & Resources

If you're having trouble check out GitHub's help documentation or shoot me an email: steve@stevenjon.es

[GitHub Pages Basics](#)

[Adding a CNAME file to your repository](#)

[Setting up a custom domain with GitHub Pages](#)

[Tips for configuring a CNAME record with your DNS provider](#)

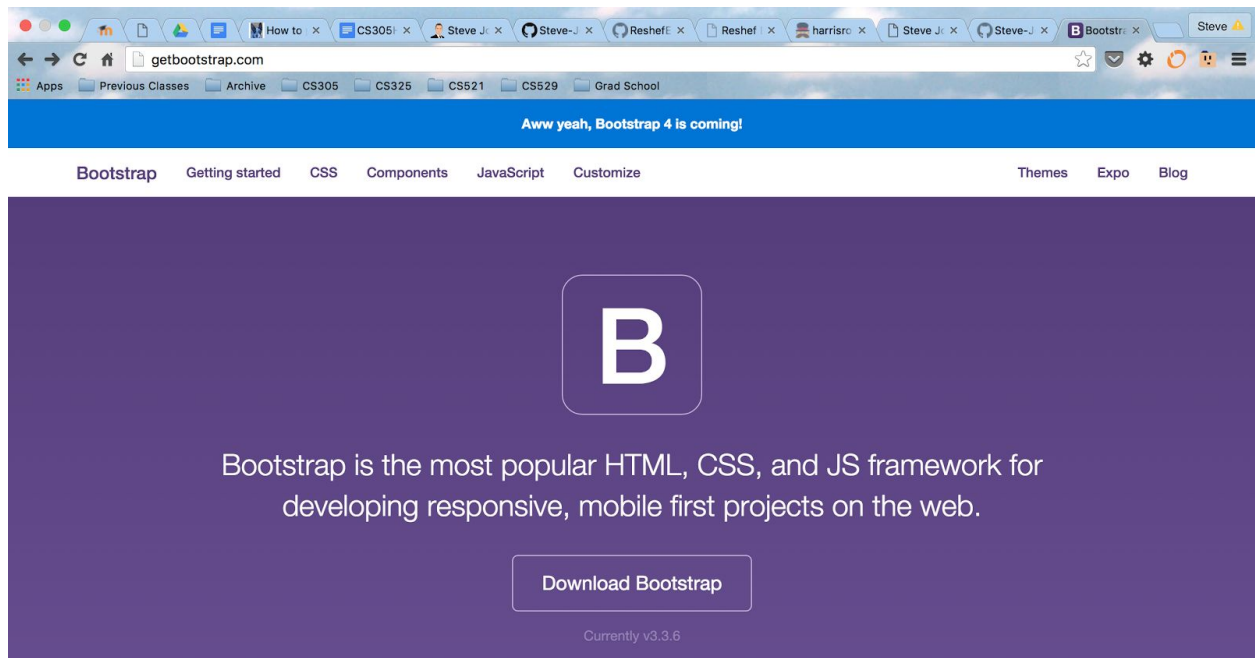
3.4 - Design & Development

There are limitless options for how you want to design and develop your website. This is an opportunity for you to learn and demonstrate your proficiency with a new technology or framework. If you are not confident with your design and development skills you can always use a template or a third party service. Below are a few that I recommend if you choose that route:

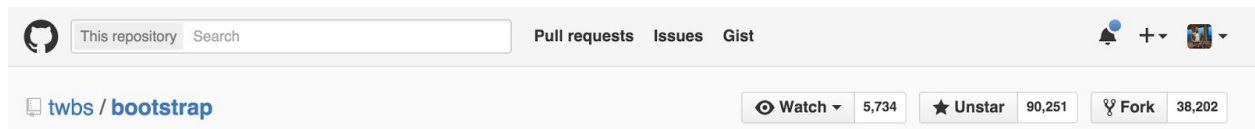
- ❖ [GitHub Pages Themes](#)
- ❖ [WordPress](#)
- ❖ [Squarespace](#)
- ❖ [ThemeForest HTML Templates](#)

3.4.1 - Bootstrap

For my personal website I chose to go with [Bootstrap](#). Bootstrap is a very popular free and open-source front-end framework for creating websites and web applications.



Bootstrap contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. It aims to ease the development of dynamic websites and web applications.



Bootstrap is incredibly popular, it is the most-starred project on GitHub, with 90,000+ stars 38,000+ forks.

You can take advantage of the components and resources Bootstrap provides while still having the freedom to design and develop your site without limitations.

I highly recommend checking out the [Hackathon Hackers directory of personal websites](#). There you will find a list of 350+ personal websites. Take a look at these websites to inspire your design. You can also post your site or any questions you have in the [HH Websites and Resumes](#) Facebook group to get feedback.



3.5 - Website Content

My recommendations for what content to include on your site

My Personal Website - <http://stevejones.io>

3.5.1 - Home Page

Keep it simple - include your name, one sentence to describe the most important thing you want to communicate with a visitor of your site. Also include links to the other content of your site.

3.5.2 - Projects

If possible list your projects and include thumbnail images. You can have each project link to a page on your site, or you can have the link open a new tab with your project on GitHub or another third party site.

3.5.3 - Portfolio

If you have design experience consider including a portfolio section. If you already have an established portfolio on a site like Behance or Dribbble, look into seeing how you can embed that portfolio into your website.

3.5.4 - Resume

Include a link to your resume on your homepage. Have that link direct to a PDF version of your resume. This way employers can easily download it. I've found this advantageous over including a resume on your site as HTML. Having a PDF in your root directory allows you to link to it using `yourdomain.com/resume.pdf`. Having it in PDF form also means employers will see it in the common form they are used to reviewing resumes.

3.5.5 - Contact

Don't forget to include your email! Employers are sometimes impressed if you have a `yourname@yourdomain.com` email address. When reviewing my resume I've often had recruiters appreciate my `steve@stevenjon.es` email address.

Include links (preferably with icons) to your online accounts such as LinkedIn, GitHub, your blog, portfolio, Twitter, etc.

4 - RESOURCES

List of all the resources I've compiled in this report that may be useful to someone who is in the hiring process for a software engineering position:

My Personal Website - <http://stevejones.io>

GitHub Repo for My Personal Website - <https://github.com/Steve-Jones/Steve-Jones.github.io>

Hackathon Hackers Facebook Group - <https://www.facebook.com/groups/1487708811477672/>

Full List of HH Subgroups - <https://github.com/HackathonHackers/groups>

Relevant HH Subgroups:

HH Websites and Resumes - <https://www.facebook.com/groups/1487708811477672/>

HH Internships - <https://www.facebook.com/groups/HHinternships/>

HH Coding Interview Prep - <https://www.facebook.com/groups/722062947858000/>

HH Interview Experiences - <https://www.facebook.com/groups/interviewexper/>

HH Job Listings - <https://www.facebook.com/groups/254805221385780/>

HH: Share Your Projects - <https://www.facebook.com/groups/778907548826382/>

Questions?

Shoot me an email, I'm happy to help: steve@stevenjon.es