Objectives

To learn the basic strategies for designing an effective professional website. To develop an appropriate marketing strategy for the skills and the preferred career trajectory of oneself and to determine how to organize the professional website to present one’s abilities. To learn about a platform, called Netlify, that allows to automate modern web projects. To explore how to use the GitHub platform to store the website content and integrate it with Netlify. To create, build and deploy a static website using GitHub and Netlify. To start generating the traffic to your site and start analyzing the data that is created.

Reading Assignment

To do well on this assignment, you should first read the Netlify Blog. Next, you should learn about static site generators by reading The Reign of Static Site Generators. Finally, you should consider obtaining your own domain, for example using Google Domains.

Professional Website for Computer Scientists

Many online articles, such as Why Every Job Seeker Should Have a Personal Website discuss the benefits of professionals having their own websites and what it should include. A professional website can set you apart from other professionals in the field and can be utilized as a single resource for obtaining information about you. It is a resource that you can control! You should think of your website as an opening door to your professional portfolio. There are a number of items your website should contain, including:

- Authentic personality (think about the layout, colors, content of the text, etc.);
- An introduction (either a catchy, short description or a bio consisting of a short paragraph);
- Resume;
- Link to open source contributions and projects.

Therefore, your first task is to develop the concept for your website. You should decide on the theme of your website (how do you want others to perceive you once they land on your site), including the overall layout, color scheme, and the presentation of the textual content about yourself. Please discuss your ideas with your course colleagues and the course instructor.

GitHub repository

As you will use a GitHub repository to store your website files, please follow the steps below to set up and configure your GitHub account. You may skip any steps that you have already done in previous classes.
1. If you do not already have a GitHub account, then please go to the GitHub website (https://github.com/) and create one, making sure that you use your “allegheny.edu” email address so that you can join GitHub as a student at an accredited educational institution.

2. Now, please add a description of yourself and an appropriate professional photograph to your GitHub profile.

3. If you have never done so before, you should use the “ssh-keygen” program to create secure-shell keys that you can use to support your communication with GitHub. You will now need to type the “ssh-keygen” command in the terminal. Follow the prompts to create your keys and save them in the default directory. That is, you should press “Enter” after you are prompted to “Enter file in which to save the key ... :”.

4. Once you have created your ssh keys, you should look in the right corner for an account avatar with a down arrow. Click on this link and then select the “Settings” option. Now, scroll down until you find the “SSH and GPG keys” label on the left, click to create a “New SSH key”, and then upload your ssh key to GitHub. You can copy your SSH key to the clipboard by going to the terminal and typing “cat ~/.ssh/id_rsa.pub” command and then highlighting this output. When you are completing this step in your terminal window, please make sure that you only highlight the letters and numbers in your key—if you highlight any extra symbols or spaces then this step may not work correctly. Then, paste this into the GitHub text field in your web browser.

Creating a Static Website on Netlify

Static websites take the content of the website, usually stored in flat files instead of databases, apply it against layouts or templates and generate a structure of purely static files that are ready to be deployed and viewed by the public. Static site generators pre-build the website into static files for deployment, but allow developers to create a powerful, server-based website locally. For example, Obama’s presidential campaign in 2012 raised $250M through a Jekyll website and in 2013, Healthcare.gov switched to an approach using Jekyll as well. Static sites are powering large projects and are not just limited to blogging. There is also a strong open source community maintaining and pushing forward a wide range of engines with different flavours and features. In addition to Jekyll, static site generators such as Hugo and Next are also widely used.

We will use a platform called Netlify, which is a new generation of site and app host that is designed to give developers a place for storing files, deploying and developing apps or running a website all in one. You will store the website content in a GitHub repository and then link from Netlify to GitHub to deploy your site by clicking on “New Site from Git”, connecting to the GitHub, selecting the correct repository, and then identifying build options before finally deploying your site as outlined in the Guide on Deploying on Netlify.

Your second task is to get to know Netlify and available static generators by reading relevant Netlify Blogs and trying to follow the workflow by creating a GitHub repository with website content, linking to it from Netlify, and finally building and deploying your sample site. Feel free to explore Netlify Templates and Netlify Templates on GitHub as a starting point.

Once you are comfortable using the tools outlined above, your next task is to create your own website. At the minimum, your website should contain the following characteristics:

1. Have an inviting theme.
2. Land on an attention capturing introduction page.

3. Contain a link to your GitHub account.

Additionally, the GitHub repository containing your website files has to contain a well-documented README file that contains the project name and description and explains the tools used to create your website and provides instructions on how to install and to build it. Please make sure to give proper credits if relevant. Please read all of the relevant “GitHub Guides”, available at https://guides.github.com/, that explain how to use many of the features that GitHub provides. In particular, please make sure that you have read guides such as “Mastering Markdown” and “Documenting Your Projects on GitHub”; each of them will help you to understand how to use both GitHub and GitHub Classroom.

If you are just beginning to create your website it maybe less overwhelming to start with an existing template. For example, to use Hugo you can follow these steps:

- Find and clone a Netlify template repository for Hugo. Alternatively, you can follow the steps outlined in Quick start with Hugo and then Hosting on Netlify.

- Create your own repository with the contents of the cloned repository.

- Go to the Netlify’s website, choose “New site from Git”, click on “Github”, choose the newly created repository, verify the specified settings, and then click on “Deploy Site”.

- You will see that Netlify creates a random link ending with netlify.com. You can click on that link to see your published site.

- Now, go to “Site Settings” and change the site name to something more meaningful. In case you have your own domain name click on “Domain Settings” and add your custom domain.

- Finally, try to make a small change in your repository (for example, modify site/layouts/index.html). Then build the site locally using the npm command. When you are ready to commit the changes and deploy your updated site, type appropriate git commit and git push commands. If you return to Netlify’s website, you should see it building your site. Once the building is complete your changes should be published on your website.

- Now you can begin to further customize the template to make your site.

Analyzing Traffic of your Website

Once you have created a professional website you should link to it from your social media accounts. At the minimum, you have to add your website link in the “See contact Info” section of your LinkedIn profile.

Now you should add appropriate analytics to see where the traffic to your website is coming from, so that you can adjust your website design strategy. We will use Google Analytics tool, however, you are welcome to utilize another tool instead. First, visit https://analytics.google.com and get the tracking ID (“Admin” → “Tracking Info” → “Tracking Code”. Now go to your Netlify account, click on the “Deploys” tab, then “Deploy Settings”, find a subsection titled “Snippet injection”, finally copy and paste Google Analytics tracking .js code.
Over the next few weeks you are to observe the analytics data produced. In particular, pay close attention to the social media traffic report, which can be viewed by going to “Acquisition” and then “Social” (as a reference see Using Google Analytics to Measure Social Media Outcome).

Although it is optional, consider setting up analytics.js to measure social interactions. This allows you to analyze the user interaction with embedded social network buttons. You can follow instructions on About Social plugins and interactions and Social Interactions.

Preparation for the Peer Review

During our next class session you will be asked to perform peer review of each other’s websites. Please make sure that you have joined the course organization and that the GitHub repository for your website is created within the course organization. At the end of the semester you will be asked to transfer ownership from the organization to your personal account.

Deliverables and Evaluation

You are invited to submit a completed website that is properly deployed using Netlify by the due date stated in this assignment. The goal of the first week of the assignment is for you to get to know the tools and practice using them, in addition to developing conceptual ideas regarding your website. Before the beginning of the second week of the assignment you should have a general concept of the main page for your website, select a site generator you plan to use, and select a unique name for your page, optionally also purchasing a domain name. **Specifically, you are asked to submit the following materials:**

1. GitHub repository titled as the name of your website. This repository needs to be created inside the course organization Allegheny-Computer-Science-480-F2018. Finally, your website repository should contain:
   
   - Properly documented README.
   - All of the necessary files to build your website.

   The instructor will evaluate your website based on the timeliness of its submission (last commit before the due date will be graded), whether your website builds correctly, and whether it satisfies the characteristics of the website outlined in the sections above.