Project Websites

Lecture 15

Outline

- Project Website: purpose, content, and style
- Creating a static website

Purpose of your project website

Website serves as an introduction to your project for

- attendees at the Capstone Presentation Event
- the person at Company X that you messaged on LinkedIn
- a person browsing the internet after a Google Search in your project area
- an expert in the field vaguely interested in your results

A website/blog for your project is an effective way to hook people to read a little further and get to know your work in depth

- *nobody* dives into code without significant motivation!

Content of your project website

First you must ask:

- *Who is your target audience?*

Are you giving a project introduction to a general audience or expert reader?

Examples for:

- A general audience
- Experts in the field

Always include *something* for the general audience (for Presentation Event)

Content of your website

- Always include an introduction
  - What is the motivating question/problem being addressed?
  - Why should the reader care?
  - How are you approaching the problem? Why is it reasonable?
- Describe how you approached the problem
  - A description of your methods describes how your approach is effective and/or novel.
  - How in-depth the explanation depends on the audience!
- End with the results and impact!

You can juggle different target audiences by hiding "in-depth" technical details behind links/dropdowns.

Style of your website

- Keep it simple!
- Pictures and plots should tell your whole story
- An informal narrative is engaging and effective
  - Taking pieces of your report wholesale is ok, as well
- Refrain from complicated layouts and too many pages
  - Readers like to scroll, not click.
- Look back at your DSC 106!

Creating a Website

Use templates/frameworks, not builders

You *must* create your website in a way that lets you own your code.

- Owning your website code allows you to:
  - version your code and reuse it easily
  - move your website to a different hosting service easily
- You *may* use templates and/or frameworks that generate HTML
  - E.g. Jekyll creates HTML from Markdown (more on this later)

Note:

- All-in-one website builders/hosts are **NOT** allowed
- E.g. you forever lose your UCSD Google Pages after graduation

Static vs Dynamic

- Static Websites run all computation in the client (web browser)
  - HTML, CSS (styling), Javascript (interactivity)
  - Very cheap or free (e.g. GitHub Pages)
- Dynamic Websites require server-side computation
  - Client sends a request to (your) server, which responds with output to client
  - You must pay for the server-side compute (more traffic = more expensive)
  - Websites that need databases or an ML model execution will be dynamic
- Try to stick to static websites (unless it's a larger part of your project)
  - Simple, cheap, portable…
  - You would be surprised at the interactive things that are possible with static pages!

See instructor and/or TA if you think you need a dynamic page

Static websites using GitHub Pages

GitHub offers free hosting of static websites:

- One personal website (<username>.github.io) per user
- Project websites (<username>.github.io/<project-page>)

Your website can consist of:

- html/js/css created by you (e.g. supply an index.html file)
- markdown files that GitHub Pages uses to build HTML, using Jekyll

Either method is fine, and allows you to own your website indefinitely.

Creating a Project Page (from scratch)

To create a project page on GitHub Pages:

1. Create a repository named <project>
2. Add/Commit/Push an `index.html` file
3. Go to Repository Settings => GitHub Page => Select Source

Your website will be available at https://<username>.github.io/<project>

- See GitHub Pages Tutorial (select: "project page"; "from scratch")

Creating a Project Page (using Jekyll)

To create a project page on GitHub Pages:

1. Create a repository named <project>
2. Go to Repository Settings => GitHub Page => Select Source
   a. Also choose a Theme here
3. Your README.md will become your website
   a. Delete/change the `_config.yml` file to delete/change the theme

Your website will be available at https://<username>.github.io/<project>

- See GitHub Pages Tutorial (select: "project page"; "using themes")

Using Jekyll to convert markdown to HTML

- Jekyll is a framework that converts markdown *content* into HTML
  - Useful for multi-page, complicated sites
- This content is converted using templates and metadata:
  - `_config.yml` contains information about the website and relevant files
  - `_layouts` contains template HTML files
  - The top of each rendered markdown file contains front matter about the page
- Examples (what is the url for each of the GitHub Pages websites?):
  - Malware Domain: https://github.com/afraenkel/capstone-malware-domain
  - Capstone project page: https://github.com/DSC-Capstone/projects-2019-2020
  - Capstone main page: https://github.com/DSC-Capstone/dsc-capstone.github.io
  - Examples of company pages: https://github.com/collections/github-pages-examples

Tutorials for creating GitHub Pages (and Jekyll)

- [GitHub Pages Guide (documentation)](#)
- [Jonathon McGlone](#) thorough, from the start, tutorial with demo code
- [Jekyll Documentation](#)
  - Installation / CLI explains how to set-up your site locally
  - Content / Site-Structure explains how to structures your site (e.g. for use on GitHub)
- [Introduction to Static Site Generators](#)
- More examples of [visualization tools](#) by Keng-Chi Chang

Grading Project Website

With the default grading choice (report=primary; website=secondary), the rubric is described below.

- A = "Well thought out *visual* presentation for a specified target audience; includes some content for general audience; code for website on GitHub.
- B = "Converted project report to directly website without much change (e.g. markdown/png); code for website on GitHub"
- C = "Has some content about project, though incomplete; may have used a website builder/hosting service"
- F = "Content is incomplete; code for website not on GitHub"