Making an Academic Website

Elissa Berwick

elissa.berwick@mcgill.ca

McGill University

February 9, 2024

1

Agenda

1. Elements of an academic website

2. No-coding methods

3. `Quarto` for academic websites

   - Getting started with `Quarto`

   - Publishing with `GitHub Pages`

   - Advanced content and customization

Website elements

Landing page

Introduce and define yourself

- Provide current position and educational background
- Identify what field(s) you work in

James Gethyn Evans   =

Welcome

I am a scholar of modern and contemporary China with a
focus on China's foreign relations with the Global South. I
am currently a Ph.D. candidate at Harvard University,
where I am writing a dissertation on global networks of

https://jagevans.com/

Separate "About" page

- A separate "About" page can provide more detail

James Gethyn Evans   ≡

About Me

James Gethyn Evans

Single "Home" page

- Or a single "Home" page can have everything

About

Welcome to my site! I am an Associate Professor in the Department of Political Science

https://www.aaronerlich.com/

Research

Information on publications **AND** current projects

- Include summaries or keywords for papers

- Link to PDFs and/or add replication files

Andrew Heiss

- About
- CV
- Blog
- Research
- Teaching
- Talks
- Now
- Uses

-
-
-
-
-
-

https://www.andrewheiss.com/research/

Teaching

Describe your teaching experience

- Detail your approach to teaching

Kathleen A. Creel

About

Research

Teaching

Contact

CV

Pedagogical Approach	- Epistemic Virtues: Doing philosophy with others teaches charitable reconstruction of an opponent's views, respectful disagreement, argument formation, and equanimity in the face of uncertainty and doubt. - Ethical Reflection: As students develop their knowledge and craft, it is important that they also practice pursuing the good. - Synthetic worldview: Learning to develop a synthetic and action-guiding world view is one of the benefits of a philosophical education

https://kathleenacreel.com/teaching.html

Teaching

Describe your teaching experience

- Add teaching resources you've developed

[emi tanaka](#)

- [Home](#)
- [CV](#)
- [Research](#)
- [Teaching](#)
- [Talks](#)
- [Blog](#)
- [About](#)

-
-

Teaching

Teaching

https://emitanaka.org/teaching/

Extras

Make it interesting!

- Share projects

  Contents

  - [tidyhte](#)
  - [regweight](#)
  - [rdd](#)

  ## Software

  tidyhte

https://ddimmery.com/software

Extras

Make it interesting!

- Share a blog

[Beatriz Milz](#)

- About
    - English
    - Português
    - Español
- Posts
    - All posts
    - English
    - Português
    - Español
- Talks
    - All/Todas
    - English
    - Português
    - Español
- Resume
-
-
-

https://beamilz.com/posts/

Extras

Make it interesting!

- Add photos and hobbies

400 Bad Request

tr5U9ui3/ddFiUBPL @ Fri, 09 Feb 2024 20:35:52 GMT   SEC-43

https://www.tesaliarizzo.com/pic

No-coding methods

Pre-made templates

- Powerpoint-like experience
- Designed by professionals

Make any template yours with ease.

Whether you need a portfolio website, an online store, or a personal blog, you can use Squarespace's customizable and responsive website templates to get started.

https://www.squarespace.com/templates
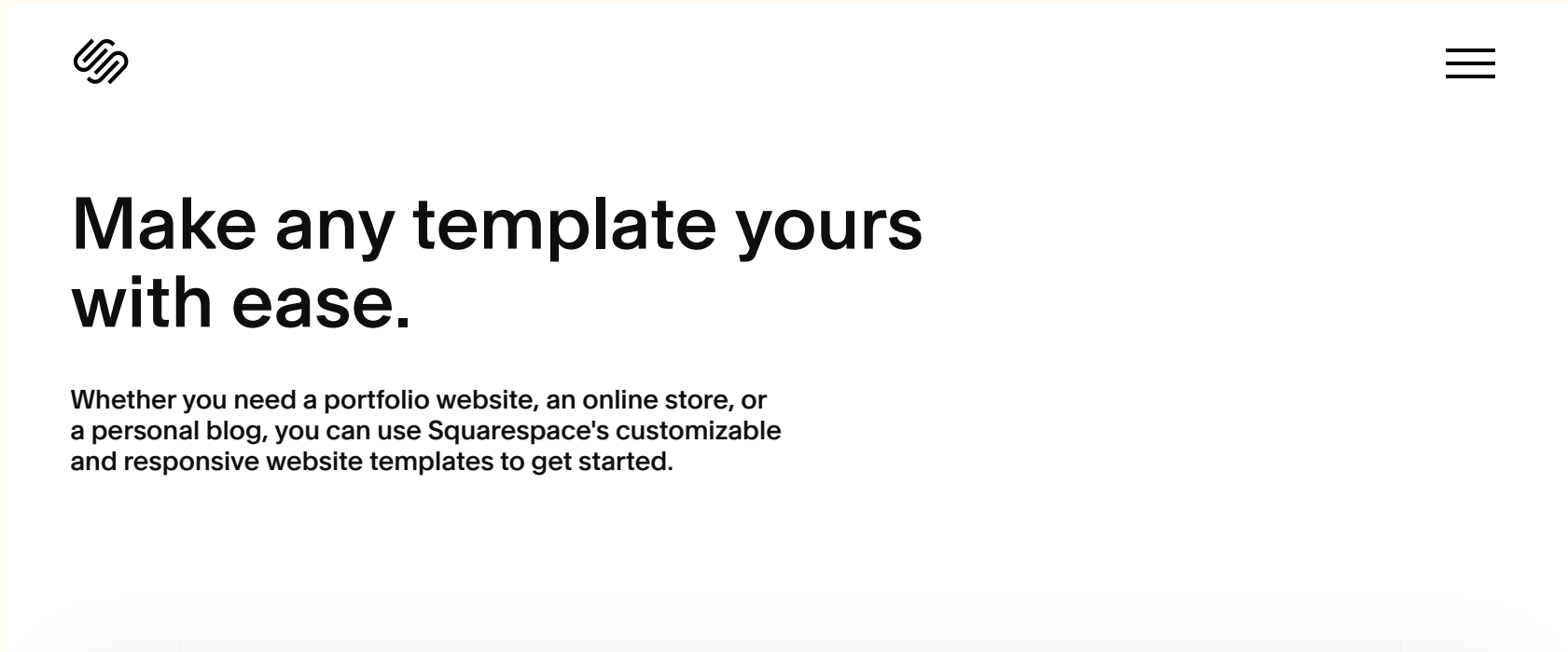
Template providers

- Pricing (CAD) for basic plan that includes domain name (i.e. `yourname.com`)

    - WordPress: from $5/month

    - Weebly: from $7/month

    - GoDaddy: from $12/month

    - Wix: from $15/month

    - Squarespace: $21/month

- Design quality and inclusions are proportional to price!

- WordPress and Weebly have free options without custom domain

- `GitHub` is free with domain name `username.github.io`... ... but requires coding

Custom domain names

- You can purchase a domain name directly from the last slide's website template providers

- OR you can purchase a domain separately from a domain registrar

  - Keeps your ownership of the domain independent of your current template

- Options include:

  - GoDaddy (largest)

  - Domain.com

  - Google Domains (easiest)

  - Cloudflare

Getting started with Quarto

Why use `Quarto`?

- Three building blocks of a website:

    - Basic structure built with `html`

    - Styling done with `css`

    - Dynamic content from `JavaScript` (not needed for a static site)

- **Static site generators**: convert easy-to-write `Markdown` to `html` for use in a static website

    - Examples: `Jekyll`, `Hugo`, `Gatsby`

- There are other `R` packages like `distill` and `blogdown` that let you do this too

    - `Quarto` is a new `Markdown` based way to generate static webite content

What is Quarto?

- RMarkdown beyond R (support for Python, Julia, Observable)
- Output types (via pandoc):
    - PDF (via LaTeX)
    - MS Word
    - Revealjs
    - PowerPoint
    - Beamer
    - Markdown
    - HTML

More information: https://www.apreshill.com/blog/2022-04-we-dont-talk-about-quarto/

Project setup

Create a new project in RStudio

- Select "new project" from upper right corner of GUI

- Select "new directory"

- Select "Quarto website"

- Give the directory a name

  - This will be the name of the `GitHub` repository as well

  - If you plan on hosting the page on `GitHub`, make sure you name the project/repository `username.github.io`

- Pick somewhere to save the entire project

More detailed directions: https://quarto.org/docs/websites/

What's in the project?

You will now have a folder with several (mostly blank) files:

- `_quarto.yml`: config file with common `yaml` information for *all* the pages in your site
  - Includes title, navbar contents, theme
- `styles.css`: where you can put design settings written in `css`
  - `css` ("cascading style sheets") are the main styling system for the entire internet
  - Holds custom design elements like colors and fonts
  - Note: you'll probably want to use `scss` (Sass) for customization instead!

What's in the project?

You will now have a folder with several (mostly blank) files:

- `index.qmd`: your site's landing page

  - This is where you'll put content to display when someone first arrives on your site

  - Don't change the name!

- `about.qmd`: an initial page on your site

  - You can copy and rename this demo page as needed

Editing `yaml`

- Change the site title to your name

```
 1  project:
 2    type: website
 3
 4  website:
 5    title: "My Name"
 6    navbar:
 7      left:
 8        - href: index.qmd
 9          text: Home
10        - about.qmd
11
12  format:
13    html:
14      theme: cosmo
15      css: styles.css
16      toc: true
```

- You can also move the navbar to the right or turn off the table of contents

Rendering

You already have the basics for a site!

- Click `Build > Render Website` to convert all of your quarto files into `html` output in the `docs` folder

    - Button is in the pane with Environment, History, Git, etc.

- A preview version should open up in RStudio

    - The site isn't live yet, this is just a local preview version!

    - To make it live we will deploy via `GitHub Pages` (to be continued…)

    - You can also use another service like Netlify

Themes: options

- You can change the appearance of your site to one of the pre-made "Bootswatch" themes (css someone else already wrote)
  - A random selection:
    - cosmo
    - lux
    - minty
    - slate
    - yeti
    - zephyr

All available themes: https://bootswatch.com/

Themes: trying it out

- Try setting `theme: zephyr` or another theme of your choice

```
 1  project:
 2    type: website
 3
 4  website:
 5    title: "My Name"
 6    navbar:
 7      left:
 8        - href: index.qmd
 9          text: Home
10        - about.qmd
11
12  format:
13    html:
14      theme: zephyr
15      css: styles.css
16      toc: true
```

- Render site to preview

Page content: Markdown

- You can use `Markdown` syntax to edit your landing page

- For example:

  - Use `#` for top-level headers, `##` for second-level headers etc.

  - Bullet points with `-`

  - Numbered lists with `1.` etc

  - Links: `[text](https://link.com)`

  - Images: `![Caption](project_directory/folder/image)`[1]

Markdown basics: https://quarto.org/docs/authoring/markdown-basics.html

1   Make sure any files are in the same directory as the project

Page content: Markdown

- Try adding an introduction in Markdown to your site and rendering it

```
 1   ## Lorem Ipsum
 2
 3   Lorem ipsum dolor sit amet, consectetur adipiscing elit.
 4
 5   - Sed at nulla lobortis, faucibus nisi et, tempor dui.
 6   - Fusce quis nulla sodales, ultricies arcu vitae, bibendum est.
 7
 8   Donec neque justo, vulputate vel velit ut, pulvinar suscipit nibh.
 9
10   1. Nullam risus erat, consectetur quis libero in, condimentum accumsan qu
11
12   Maecenas egestas pulvinar lobortis.
```

Page content: "About" pages

- For easier layout options, make pages into "About" pages
- To do this, you will add an `about` option to the `yaml` of the page you want to change (not `_yaml.qmd`!)
- You will need to pick one of 5 templates for your About page:
  - `jolla`
  - `trestles`
  - `solana`
  - `marquee`
  - `broadside`

More information: https://quarto.org/docs/websites/website-about.html

About pages: trying it out

- Try making `index.qmd` an "About" page using `template: jolla`

```
 1  ---
 2  title: "Finley Malloc"
 3  about:
 4    template: jolla
 5    image: profile.jpg
 6    links:
 7      - icon: github
 8        text: Github
 9        href: https://github.com
10      - icon: file-pdf
11        text: CV
12        href: my_cv.pdf
13  ---
```

About pages: trying it out

- Add a picture of yourself using `image: profile.jpg`

- You can also add links to social media, `GitHub` or files like your CV[1]

```
 1  ---
 2  title: "Finley Malloc"
 3  about:
 4    template: jolla
 5    image: profile.jpg
 6    links:
 7      - icon: github
 8        text: Github
 9        href: https://github.com
10      - icon: file-pdf
11        text: CV
12        href: my_cv.pdf
13  ---
```

Make sure any files (or their parent directories) are in the same directory as your project!

1 Available icons: https://icons.getbootstrap.com/

Make a new page

- You can make new pages by creating new `.qmd` files in the project directory

- Try making a new quarto file called `research.qmd`[1]

- You can add content using `Markdown` as before

    - **Tip**: you can also format *any* page (or part of a page) using one of the "About" templates

1. Or just rename the blank file `about.qmd`, it's not the same as your "About" page

Add it to the site

- Add your new page to the site by linking it to the navbar in the `_quarto.yaml` file

```
 1  project:
 2    type: website
 3    output-dir: docs
 4
 5  website:
 6    title: "My Name"
 7    navbar:
 8      left:
 9        - href: index.qmd
10          text: Home
11        - about.qmd
12        - research.qmd
13
14  format:
15    html:
16      theme: zephyr
17      css: styles.css
18      toc: true
```

Publishing with GitHub Pages

Project setup

In your RStudio project:

- Open the file `_quarto.yml` and set `output-dir: docs`

```
 1  project:
 2    type: website
 3    output-dir: docs
 4
 5  website:
 6    title: "My Name"
 7    navbar:
 8      left:
 9        - href: index.qmd
10          text: Home
11        - about.qmd
12        - research.qmd
```

- Add an empty text file to the project directory and name it `.nojekyll`

- Render site using `Build > Render Website`

GitHub setup (manual)

- Create a repository in your `GitHub` account that has the **same** name as your RStudio project

- From within the new repository, select `Add file > Upload files` and upload the entire *contents* of the project directory

- Every time you want to edit your site you will need to:

  1. `Build > Render Website`

  2. Upload **all** new files to `GitHub`

git and RStudio

- Haven't used `git` directly from RStudio before?
    - You will need to generate a PAT using
      `usethis::create_github_token()`
    - Store it in RStudio using `gitcreds::gitcreds_set()`

GitHub setup (`usethis`)

- Set up your project for git tracking by running `usethis::use_git()`

- Create a corresponding repository on `GitHub` by running `usethis::use_github()`

- Every time you want to edit your site you will need to:

  1. `Build > Render Website`

  2. Stage files by checking names in `Git` tab

  3. Commit files by clicking `Commit` and adding a message

  4. Push to the repository by clicking `Push`

GitHub Pages setup

In the repository page on `github.com`:

- Go to the "Settings" tab in the top menu bar

- Then select "Pages" from the left menu bar

- Set branch to `main` or `master`[1] and selected folder to `/docs` and save

- The url will change when the page is fully deployed

- Go to the "Actions" tab in the top menu bar to see the deployment status

1. Depending on your setup the main branch will either be called `main` or `master`; you can find out which one you're on by typing `git branch` in the Terminal

Advanced content and customization

Page content: pandoc

- Markdown is supposed to be easy to read

- For more complex formats, you can use pandoc syntax

- Try adding columns using a .columns div

```
 1  Lorem ipsum dolor sit amet, consectetur adipiscing elit.
 2
 3  ::: {.columns}
 4  ::: {.column width="50%"}
 5  Sed at nulla lobortis, faucibus nisi et, tempor dui.
 6  :::
 7
 8  ::: {.column width="50%"}
 9  Fusce quis nulla sodales, ultricies arcu vitae, bibendum est.
10  :::
11  :::
```

Custom styles

- Want to customize colors, fonts, and more? Use Sass!
  - **Tip**: Add customization on top of an existing theme by using multiple theme arguments
    - ex: `theme: [journal, styles.scss]`
- How:
  - Make a new file ending in `.scss` and add it to the theme setting in your `_yaml.qmd` file
  - Make two sections in your new file:
    - `/*-- scss:defaults --*/`
    - `/*-- scss:rules --*/`

Defaults

- Define variables like colors and fonts in `scss:defaults`

```scss
 1  /*-- scss:defaults --*/
 2
 3  // import a font from google
 4  @import url('https://fonts.googleapis.com/css2?family=Merriweather&displa
 5
 6  // set fonts for text
 7  $font-family-sans-serif: "Merriweather", Palatino, FreeSerif, serif !defa
 8
 9  // document colors
10  $body-bg: #181818;
11  $body-color: white;
12  $link-color: #75AADB;
13
14  // code blocks
15  $code-block-bg-alpha: -.8;
```

More on variables you can set: https://quarto.org/docs/output-formats/html-themes.html

Rules

- Use `scss:rules` to define additional css styles

```
1  /*-- scss:rules --*/
2
3  // highlight a word
4  .highlight {
5    color: #D33582;
6    font-weight: bold;
7  }
8
```

- To use this style, you would type:
  `[highlighted phrase]{.highlight}`

- Which would produce: highlighted phrase

More on using Scss customizations: https://quarto.org/docs/output-formats/html-themes-more.html

49

Resouces

- Guides to website setup:
  - https://quarto.org/docs/websites
  - https://ucsb-meds.github.io/creating-quarto-websites
  - https://ddimmery.com/posts/quarto-website
  - https://www.marvinschmitt.com/blog/website-tutorial-quarto
- Inspiration:
  - https://github.com/mcanouil/awesome-quarto
- Use a custom domain name with `GitHub Pages`:
  - https://hossainkhan.medium.com/using-custom-domain-for-github-pages-86b303d3918a