Tutorial: Build your own website

Quico Spaen

IEOR GSG: Lunch & Learn Seminar

Feb 2, 2018

Why a website?

- It is your personal **brand**,
- A website **showcases** your work,
- It generates **attention** for your research,
- It looks **professional**,
- It may **attract the attention** of future collaborators, employers, etc,
- It can serve as your **blog** to disseminate your thoughts and ideas,
- It enables you to **easily share** information about yourself.

What is a good website?

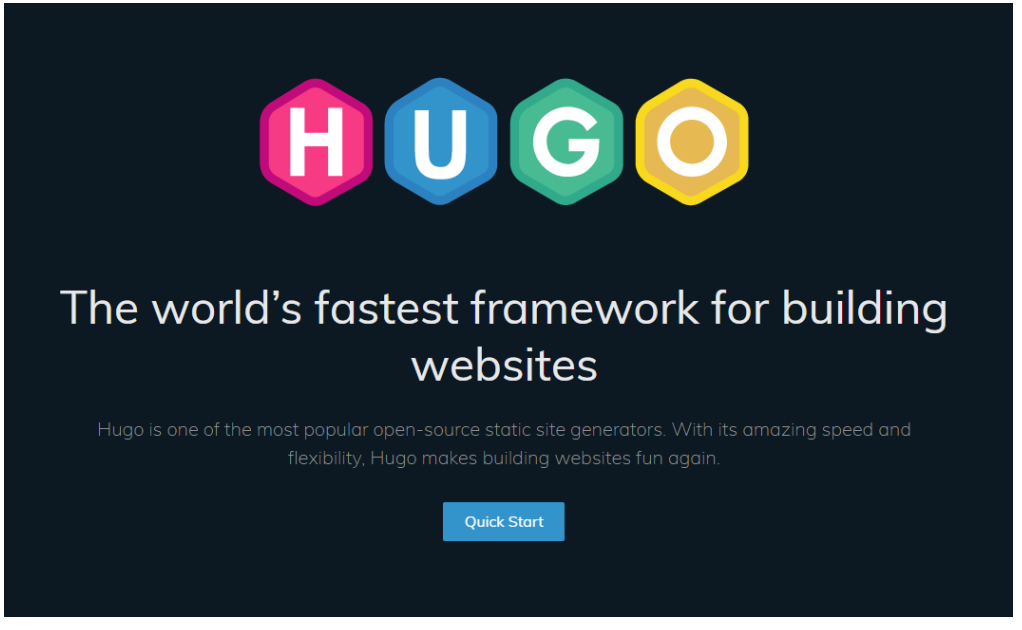
- It is **easy to find**,
- It looks **good**,
- It works well on **mobile devices**,
- It is **up-to-date**,
- It is **easy** to **setup**,
- It is **easy** to **maintain / update**.

Today's goal

Leave with a functioning personal website with:

- Your photo,
- 1 line bio,
- 1 uploaded publication

Enter Hugo


HUGO
The world's fastest framework for building websites
Hugo is one of the most popular open-source static site generators. With its amazing speed and flexibility, Hugo makes building websites fun again.
Quick Start

Why Hugo?

- Very fast
- Easy to edit with Markdown files
- Responsive design
- Support for themes & extensions
- Can be hosted for free on Github Pages

Academic Theme

- Made for academic personal websites
- Supports math syntax & code highlighting
- Built-in support for publication (formatting)
- Extendible with widgets
- Popular & actively maintained

Built-in widgets

- About/biography
- Recent & selected publications
- Recent news/blog posts
- Projects
- Talks
- Contact
- Hero (introduction banner)
- Featurette (show features)
- Custom widget (add your own content!)

Here we go!

Download & Install Git / Github Desktop

Comfortable with Git?

- Install Git - https://git-scm.com
- Want to learn Git? Checkout this tutorial.

Not so comfortable with Git?

- Install Github Desktop - https://desktop.github.com/.

Fork my Academic-Kickstart repo

- Log-in / create a Github account (free) if you don't have one.
- Go to https://github.com/quic0/academic-kickstart. Note: Use this version since I made some changes.
- Click Fork to create your own copy.
- For the forked repository, click on settings.
- Change repository name to `website`.
- Create a local copy of the website with the following commands (or use Github Desktop):

```
git clone CLONE_LINK_HERE my_website
cd my_website
git submodule update --init --recursive
```

Setting up repositories

We will use two repositories:

website (public)

Here we store all Hugo files. This is the repository you will edit. **We made this one already**.

$USER.github.io (public)

Here we store the compiled website. We will set it up for automatic updates. Replace `$USER` with your github username. **Make this public repository now**.

Download & Install Hugo

MacOS

- Download and install HomeBrew. It's a great tool.
- Install Hugo from the terminal with `brew install hugo`.

Windows

- Open PowerShell with administrative privilages (right-click, run as administrator).
- Install the Chocolately package manager with the following command: `Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString( 'https://chocolatey.org/install.ps1'))`.
- Install Hugo with `choco install hugo -confirm`.

Linux

- If you have the latest Ubuntu verison, use `snap install hugo` or use `apt-get install hugo`.
- Otherwise download Hugo binary from GitHub.
- Place the binary in your path.

Viewing your website locally

- View your new website by running the following command:

```
hugo server
```

- Now visit http://localhost:1313 and your new Academic powered website will appear.
- Your local version will update automatically with changes!

Structure of the website

A Hugo website typically has the following folder/files:

- `config.toml` - Configuration file for the website.
- `content/` - Folder for all website pages.
- `public/`- Folder with the compiled version of the website.
- `static/` - Folder for static objects.
    - Files go to `static/files/`
    - Images go to `static/img/`
- `themes/` - Folder for the website theme(s).

Editing the config file `config.toml`

- Change the baseurl to

  ```
  https://$USER.github.io/
  ```

  Replace `$USER` with your Github username.

- Edit your name and role on lines 53 & 54.
- Change line 58 to

  ```
  organizations = [{name = "UC Berkeley", url = "http://ieor.berkeley.edu/"}]
  ```

- Edit your contact information on lines 62-67.

Notice the changes to your website!

Commit the changes

Git

- Review changes with `git status`
- Add changes with `git add`
- Commit with `git commit`
- Push to GitHub with `git push`

Github Desktop

- Add summary
- Click commit to master
- Click Push to origin (GitHub)

Setup deploy script

We will use https://travis-ci.org/ for automatic deployment of our website.

First we need a GitHub personal token:

- Go to your GitHub account settings.
- Click on `Developer Settings` -> `Personal access tokens` -> `Generate new token` and enter your password.
- Under token subscription enter `Travis CI Website` and select scope `public_repo`. Click generate token.
- **Copy the token**. You will need it in the next step!

Setup deploy script II

- Go to https://travis-ci.org/ and sync Travis with your GitHub account.
- Click add repository (plus symbol on the left).
- Toggle the switch for the `website` repository.
- Click on the settings icon for `website`
- Under environment variables, add a variable with name `GITHUB_TOKEN` and as value the token you generated. Click add.

Setup deploy script III

- Edit your `.travis.yml` file by replacing `<USER>` with your GitHub username.
- Commit and push the change to GitHub!
- Now Travis will automatically publish your website to your `$USER.github.io` repository. **You can access your website via `https://$USER.github.io`**.

Structure of a webpage

The pages always have the folowing format.

```
+++
PREAMBLE - Configuration for the page
+++
BODY - Page content shown on website
```

Editing a webpage

- The website is written in Markdown. Markdown falls somewhere in between plain-text and LaTeX.
- Easy to edit yet powerful.
- Some simple Markdown commands
  - Heading with #, sub-headings with ##, and so forth.
  - *Italic* with *asterisks*.
  - **Bold** with **asterisks**.
  - Math mode with $ as in LaTeX.
  - For simple lists use:

```
- item 1
- item 2
```

Let's edit your bio

- Open the file `content/home/about.md`. This contains the about page.
- Replace your bio text with:

```
Hello, my name is ..... and I am a PhD student
at UC Berkeley, Department of Industrial Engineering
and Operations Research.
```

- Commit the changes to GitHub.

Add a publication

- From the command line, run `hugo new publication/NAME.md` where `NAME` is a short name for your publication. This creates a new file for your publication. **Don't forget the `.md` extension!**
- Edit title, authors, journal, and abstract.
- Add a link to the pre-print / pdf.
- Commit your changes and push to GitHub. See your website change!

Update your photo

Recall that images go to `static/img/`.

- Get an photo of yourself.
- Optional: Crop it to `1000 x 1000` pixels with https://imageresize.org/.
- Replace the file `static/img/portrait.jpg` with a photo of yourself.
- Commit and push!

Custom domain name

- Buy a domain name on Google Domain (or another registrar). About $10 a year.
- Link it to `htpps://$USER.github.io`
- Setup custom domain under `$USER.github.io` settings -> GitHub Pages.

Next steps!

Read the Academic docs for help with editing
https://sourcethemes.com/academic/docs/

Stuck?

Ask for help / read the docs

Documentation

- **Academic theme** - Docs: https://sourcethemes.com/academic/docs/
- **Hugo** - Quickstart: https://gohugo.io/getting-started/quick-start/
- **Hugo** - Docs: https://gohugo.io/documentation/
- **Hugo** - Themes: https://themes.gohugo.io/

Questions?

Slideshow created using Remark JS.