

Building your own research website

To advertise yourself

To feel cool

Dr. Antonio J. Porras-Valverde

Heising-Simons Postdoctoral Fellow at Yale University

Co-founder of Cenca Bridge



*Dr. Victor Calderon, Vandy alum
Senior Machine Learning Engineer at
5x5 Technologies Inc.*



*Dr. Natalie Hinkel, Past Vandy postdoc
Assistant Professor at Louisiana State
University
Adjunct Professor at UT San Antonio*

Credits

Event Discussion Topics

- Setting up GitHub
- Creating a repository for your website
- Downloading Brackets text editor
- Choosing a template from <https://html5up.net/>
- Populating the template with your information
- Peer review it for improvements!
- Publishing your website

Why have a website?

- Academic web presence:
 - A place to refer to people in talks/conferences, friends and family, random people you have not met

- Supplement a CV
 - You can include pictures, videos, longer explanations of your research, explain affiliations (or gaps), highlight achievements.
 - You can include personal (if you wish) information about hobbies, resources for people who ask you about where to find *xy*, and things you are passionate about aside from your research

The filling of your website

- What pages do you want to have?
Minimum: Splash page, Research, Contact
Additional: Break your research foci into separate pages, some place for outreach/media content, CV as a standalone page, hobbies
Be conscious of what you post online!
- Who is this page for?
My recommendation is to include background, motivation, and few details.
- Remember lots of pictures!
One picture (or plot) for every paragraph of text.

Second order consideration

- How often are you going to update your page?
Once a year? Once a month?
- Be cognizant of the information that you are making available to the public (or for web scrapers).
- Search Engine Optimization
Something to think about as you write text because this is how people will find you. Overview: <https://moz.com/beginners-guide-to-seo>
- Website address
Ultimately, a personal option. Keeping it short, simple, and findable helps!

Examples

<https://aporrasval.github.io/>

<http://emergence.asu.edu/>

<http://jedidahislerphd.com/>

<https://www.msoaresfurtado.com/>

<https://mdlreyes.github.io/>

<https://daxfeliz.github.io/>

<https://www.karanjani.com/>

<https://carlnotsagan.github.io/>

Setting up GitHub

- Open terminal, write “git clone <https://github.com/git/git>” if you don’t have git installed. To test if you have it or not, write “which git”*
- Go to <https://github.com/> to create an account if you don’t have one already*
- Create a repository (the home of your website).*
- Name the repository your [username.github.io](https://github.com/username.github.io) (this is the standard. You can change it in the future, but we won’t be covering how to, today).*
- Select “Public” to make it a public repository and add a Readme file*
- Create repository!*

Setting up Brackets

- Install Brackets for editing HTML5 code
- Compile and see what your changes look like!

Linking web repository to local computer: setting up your username

- *To set up your Git username for every repository on your computer, in your terminal, type: `git config --global user.name "astrotonyPorras"`*
- *To confirm that you have set the Git username correctly, type: `git config --global user.name`*

Linking web repository to local computer: setting up your email

- ❑ *To set up your Git username for every repository on your computer, in your terminal, type: `git config --global user.email "Your_Email"`*
- ❑ *To confirm that you have set the Git username correctly, type: `git config --global user.email`*
- ❑ *Under documents, create a folder named "Repositories"*

Authenticating with GitHub from Git

- *We will connect over HTTPS using git clone:*
- *Inside the folder “Repositories”, type: git clone
https_website_of_your_github_repo*
- *In your local repository, create a folder named “CV” and “Images” and add your CV (or any file) and an Image to those folders. Make sure they are not empty!*

Authenticating with GitHub from Git

- *Type: git status (to see the commits you need to make)*
- *Type: git add CV and git add Images so that we can push those changes to GitHub and see them*
- *Type git commit -m "Added new files for the website"*
- *Before pushing, we need to generate a personal access token:*
 - *Go to "settings", then "Developer settings". Click on "Personal access tokens". Click on generate new token. In the "Note", provide a descriptive name to your token. Give an expiration, and select the scopes you would like to grant this token. Then, generate token!*
- *git push origin main*

Choosing a website template

- Choosing a template from <https://html5up.net/>*
- Download the template and place it in your local computer repository under `Repositories/website_folder/`*
- git add all of the files!*

Time to get working!

- Don't forget to:*
- git add*
- git commit*
- Once everything is ready, git push*