# Hugo website builder workshop!

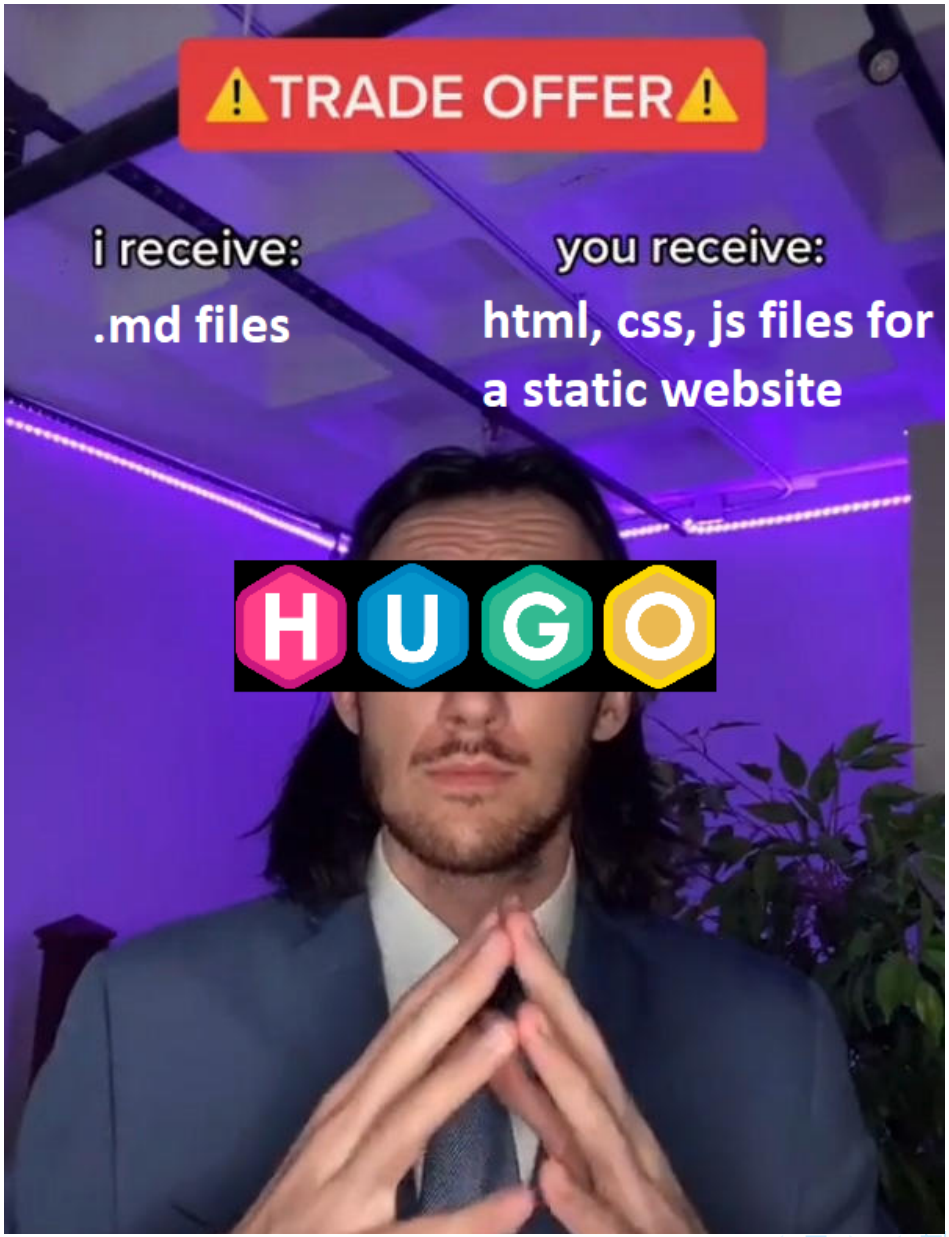
## Make a portfolio website!

Nathaniel Budijono

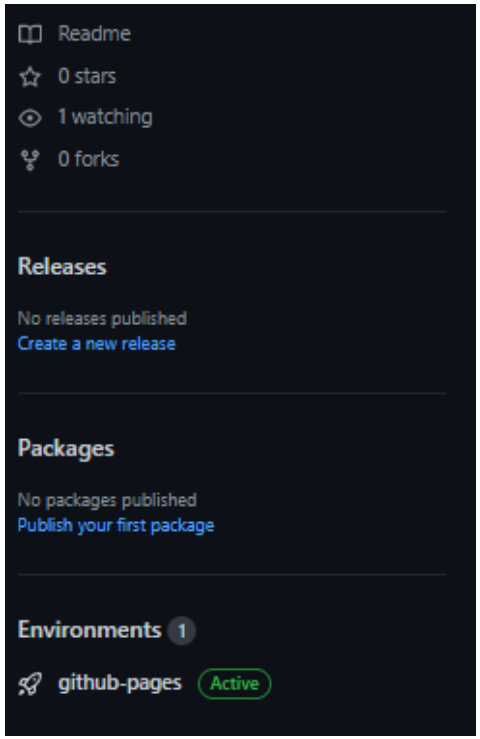
UMN ADC

April 26, 2022

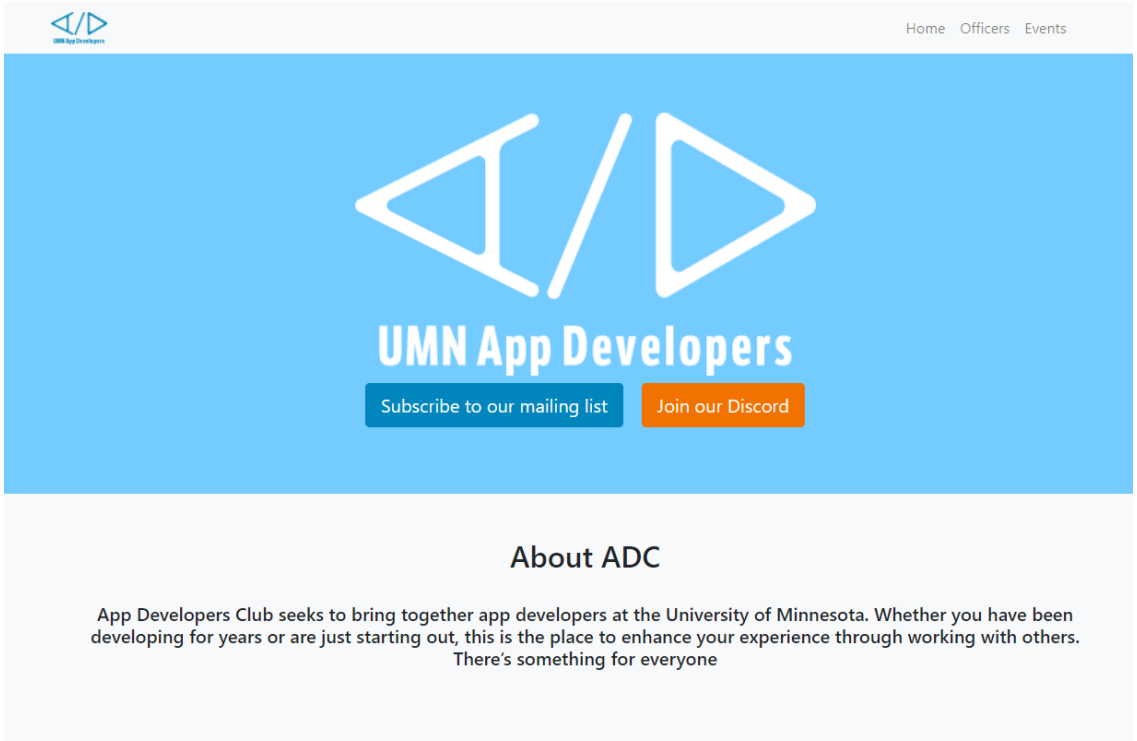
# What is Hugo?


⚠TRADE OFFER⚠
i receive:
.md files
you receive:
html, css, js files for
a static website
HUGO

# What is GitHub Pages?



# Hugo/GH Pages powers the ADC website




https://adcumn.org

# Setup

1. Create a GitHub account

# Setup

1. Create a GitHub account
2. Install git

# Setup

1. Create a GitHub account
2. Install git
3. Install Hugo

# Setup

1. Create a GitHub account
2. Install git
3. Install Hugo
4. Pick a theme

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.
- Running `hugo new site <your-site-name>` makes a new directory called `<your-sitename>` with subdirectories called `content`, `static`, and `themes`.

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.
- Running `hugo new site <your-site-name>` makes a new directory called `<your-sitename>` with subdirectories called `content`, `static`, and `themes`.
- Clone your theme's GitHub repo into `themes`.

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.
- Running `hugo new site <your-site-name>` makes a new directory called `<your-sitename>` with subdirectories called `content`, `static`, and `themes`.
- Clone your theme's GitHub repo into `themes`.
- `hugo new <title>.md` creates a new page `<title>.md` in the `content` folder and autogenerates the front matter.

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.
- Running `hugo new site <your-site-name>` makes a new directory called `<your-sitename>` with subdirectories called `content`, `static`, and `themes`.
- Clone your theme's GitHub repo into `themes`.
- `hugo new <title>.md` creates a new page `<title>.md` in the `content` folder and autogenerates the front matter.
- See the Markdown cheat sheet.

# Theme-agnostic tips

- Read your theme's documentation! It may also have an `exampleSite` directory in its GitHub repo. See its `config.toml`.
- Running `hugo new site <your-site-name>` makes a new directory called `<your-sitename>` with subdirectories called `content`, `static`, and `themes`.
- Clone your theme's GitHub repo into `themes`.
- `hugo new <title>.md` creates a new page `<title>.md` in the `content` folder and autogenerates the front matter.
- See the Markdown cheat sheet.
- Run the website locally with `hugo server` and view at `localhost:1313`.

# Creating a Toha site

## Toha theme

# Creating a Toha site

## Toha theme

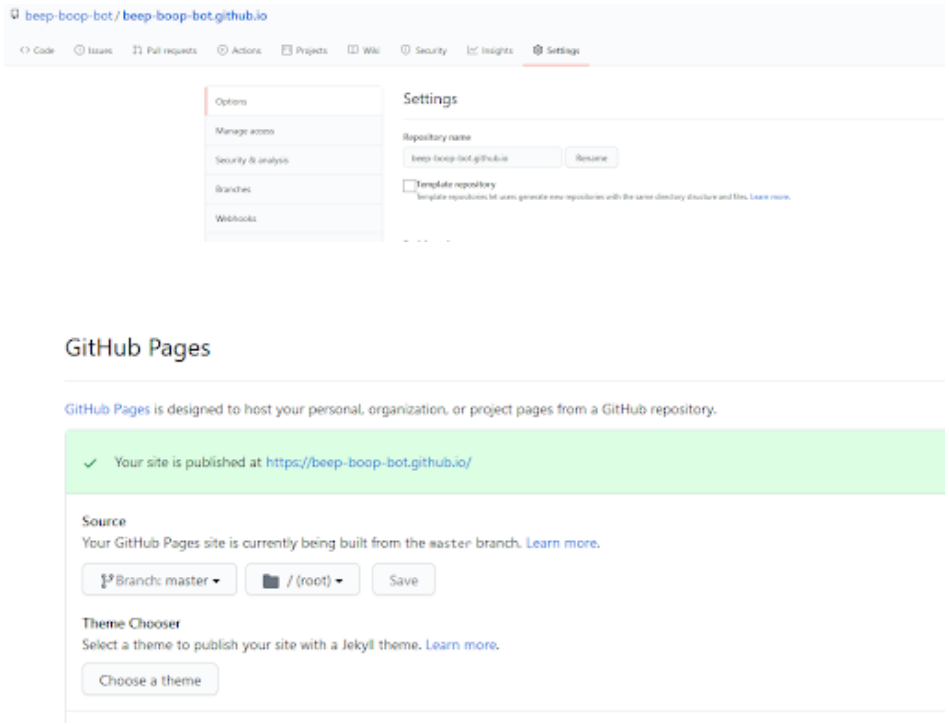
1. Clone the starter code: `git clone --recurse-submodules https://github.com/hugo-toha/hugo-toha.github.io`

# Creating a Toha site

## Toha theme

1. Clone the starter code: `git clone --recurse-submodules https://github.com/hugo-toha/hugo-toha.github.io`
2. Run the website locally

# Deploying the site (first time)




```
git submodule add -b main
https://github.com/<your-username>/<your-username>.github.io.g
public

hugo
```

# Deploying the site (each update)

1. Log changes to git: `git add .`

# Deploying the site (each update)

1. Log changes to git: `git add .`
2. Commit changes: `git commit -m "a descriptive message"`

# Deploying the site (each update)

1. Log changes to git: `git add .`
2. Commit changes: `git commit -m "a descriptive message"`
3. Push changes to GitHub: `git push -u origin master`