Deploying a web app

We've written web apps

- Seen them on "localhost"

How to get them where other people can see?

Static-vs-dynamic

You need a **host** for your content

A big factor is what you need to host

- static HTML/JS/CSS
    - `dist/` contents after `npm run build`
- OR a server that generates dynamic content
    - Directly or via web services

Just static is easier/cheaper

- Limited functionality
- Great for one-way information flow

A few options

There are countless hosting providers

- Here are a few free/cheap options
    - I've used or heard recommended
    - Not an endorsement
    - Not exhaustive list
- Good to show off
    - Anything with high traffic/high reliability should pay for stronger options

Github pages

https://pages.github.com/

- Static only
- Either a site for repo, or a site for you
- Change settings to publish a `github.io` site

Confusing: Some options we DON'T want:

- We aren't building a site from `.md` files using Jekyll
- `gh-pages` CAN be used, not required
- Docs confuse by talking about both

Creating a GitHub Pages site for a repo

- Will publish to `https://{USER}.github.io/{REPO}/`
    - Or configure a domain you have registered
    - Registering a domain separate process ($)

Deploying Manually

- Configure the Github Pages in Repo settings
  - Branch
  - Directory
- Configure `vite.config.js`
  - set `base` to match Github path
    - Ex: `/YOUR_REPO_NAME_HERE/`
  - set build path to match setting choice

```
// one option in vite.config.js
build: {
  outDir: 'docs', // Matches Github Pages setting
}
```

  - MAY DESTROY EXISTING CONTENT in that dir!
- `npm run build`, then add, commit, and push files

Deploying with gh-pages

- Configure Github Pages in Repo settings
  - `gh-pages` branch
- Follow **https://vitejs.dev/guide/static-deploy.html#github-pages**
  - install `gh-pages`
- `npm run deploy` when you want to publish the site

Both configurations can take a few minutes

- Refresh the Github Pages page in Repo Settings
  - Watch "last deployed"

Additional Notes

- Can use custom domain name
  - You must have one registered ($)
  - **https://docs.github.com/articles/using-a-custom-domain-with-github-pages/**
- Can have one overall subdomain for you
  - Create public repo `{username}.github.io`
    - Ex: `swiftone.github.io`
  - Will be `https://{username}.github.io/`
    - No required path
    - repo `/index.html` is `https://{username}.github.io/index.html`

Github Pages does not always serve index.html

- Use hash-based deeplinking for a SPA, OR
- Use multiple html files, not a SPA, OR
- Use a hacky trick so the 404 page returns SPA html

Vercel offers hosting for various frameworks

- Has Vite template, but uses Vite w/Vue, not React
    - Google for unofficial tutorials for Vite/React
- Vercel is notable for Next.js (server-side React)
- Works by having a separate github repo
    - Push to that repo to deploy to site

Fly.io has more free options

https://fly.io/docs/speedrun/

- Free tier for a lot of options
- Static requires running a server
  - Which you have to provide
  - But the docs have a demo
- I had some errors/bumps

Render offers free STATIC hosting, $ for more

https://render.com/

- I've not used them
- Wanted more access to my repo than I liked

https://create-react-app.dev/docs/deployment#render

Amazon Web Services (AWS)

Notice: Most of AWS isn't always free

- And can wrack up bills if you exceed use
- BE CAUTIOUS

You have many options with Amazon

- S3 can host static websites
- Amplify can host a single page app
- Lightsail

```
https://docs.aws.amazon.com/AmazonS3/latest/
userguide/WebsiteHosting.html
```

Cloudflare Pages came highly recommended

https://pages.cloudflare.com/

- I haven't had time to confirm yet
- Looks like a similar offering

Railway.app another host

- I use this with server + static files
    - Like what I show in INFO 6250
- May not be great choice for JUST static files
- Was a "Free" tier, now "Hobby"
    - Now requires credit card :(
    - "Free" for < $5/month use
        - Can lock at $10/mo
            - Good if you go "viral"

Summary - Hosting

- **Hosting providers**
    - Places to put your web content as a site
    - Offer their domain by default
    - You have to register a domain separately ($)
        - Then configure host to use domain
- **Static files** (HTML/JS/CSS) vs server is a big factor
    - "Serverless" = Static files calling web services
        - Web services on other server than host
- Free and Paid options exist

Summary - Options

- So many hosting options
- Some examples:
  - Github pages
  - Vercel
  - Render
  - Fly
  - Railway.app
  - AWS S3
  - AWS Amplify
  - AWS Lightsail
  - Cloudflare Pages