

Deploy Front End



Apps



Deploying donation-svelte
to Netlify & Vercel services

Cross-Origin Resource Sharing (CORS)

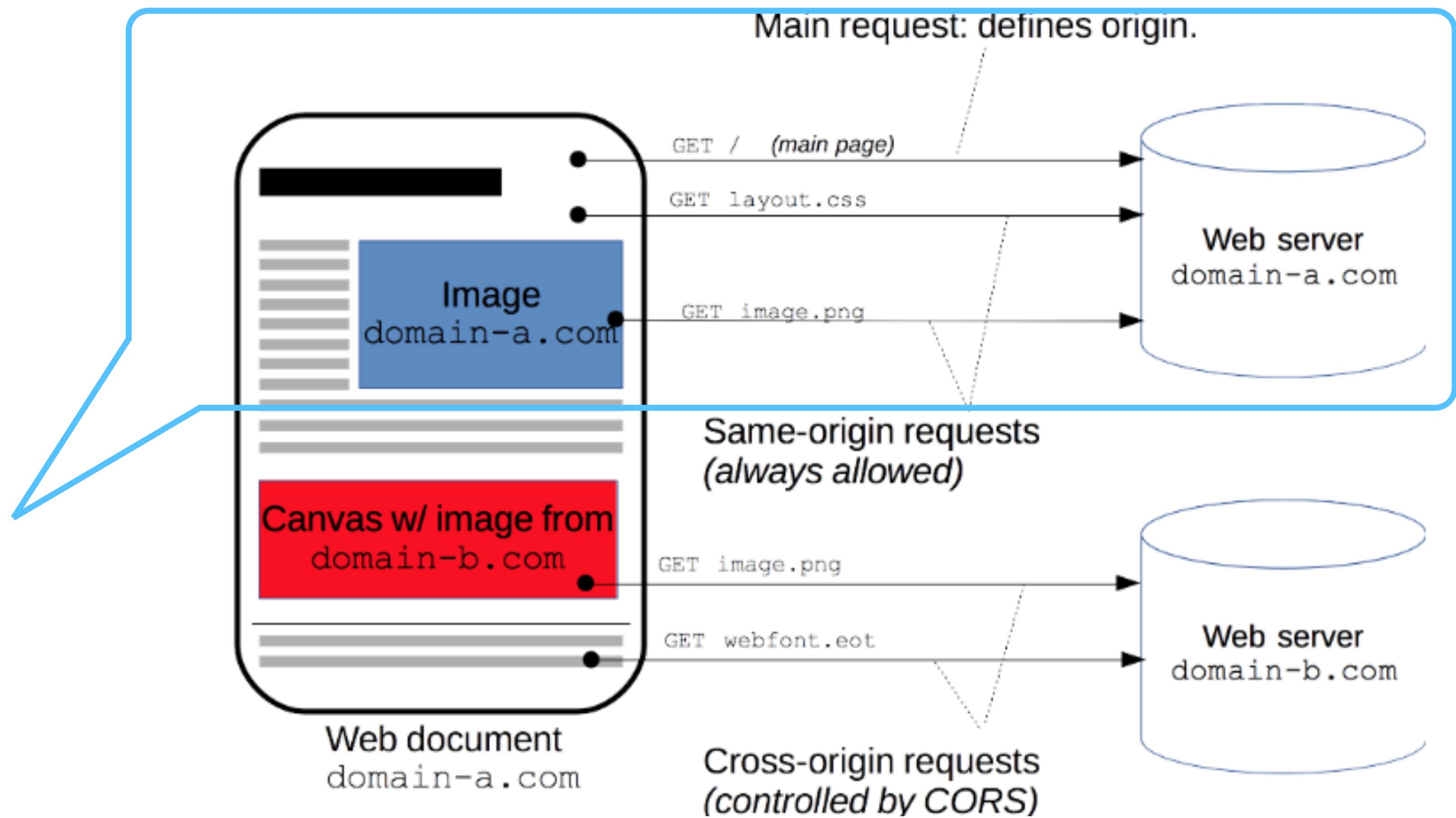
For security reasons, browsers restrict cross-origin HTTP requests initiated from scripts.

The CORS mechanism supports secure cross-origin requests and data transfers between browsers and servers.

<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

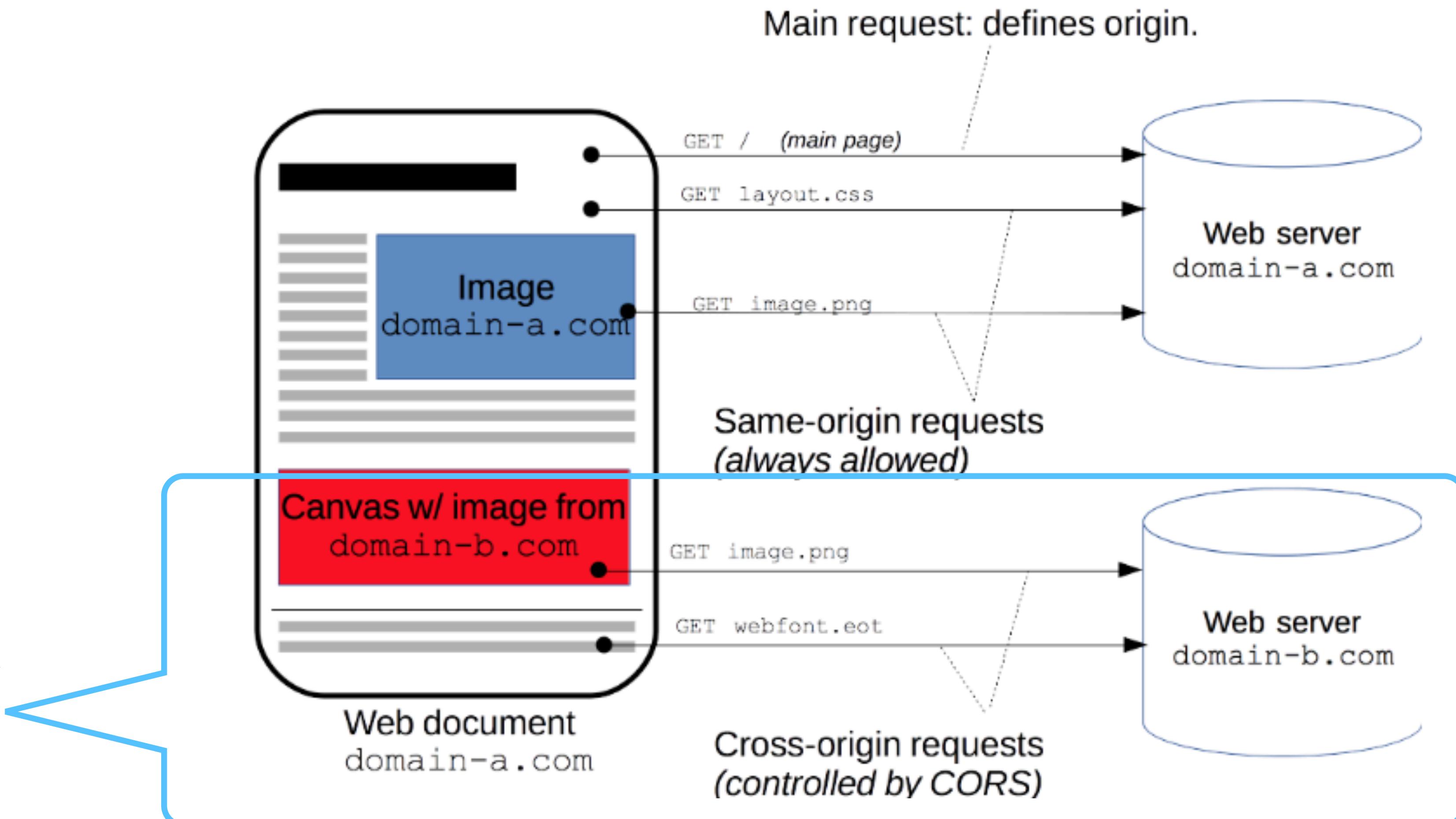
Browser Security: Same Origin Policy

- By default, browsers applications can make request to the domain from which they were loaded. This is called the Same-Origin Policy.
- So if an app is loaded from domain-a.com, then it can make requests to domain-a.com after it has been loaded



Browser Security: CORS: Cross-Origin Resource Sharing

- If app is making a request to a different domain, it must be explicitly enabled by the **server**
- This is controlled by headers included in the server.
- Our server side framework will have a facility for enabling this.



CORS in Hapi

server.ts

```
const server = Hapi.server({  
  port: process.env.PORT || 4000,  
  routes: { cors: true },  
});  
...
```

- When cors enabled: the server will automatically include headers like:
 - Access-Control-Allow-Origin
 - Access-Control-Allow-Methods
 - Access-Control-Allow-Headers

- When cors: true is enabled, server will:
 - Accept requests from different origins (domains)
 - Include the necessary CORS headers in responses
 - Allow browsers to make cross-origin requests to API
 - This is particularly important for Donation API because:
 - The frontend application might be hosted on a different domain than the Hapi app
 - Mobile apps or other services might need to access the API
 - Third-party applications might need to integrate with donation

Bundlers

- Module Bundler: take all of javascript files, bundle & compress.
- This monolithic file is then added to your index.html file as a script tag.
- When the browser loads your page, it only has to make one additional HTTP request.



PARCEL
Blazing fast, zero configuration web application bundler



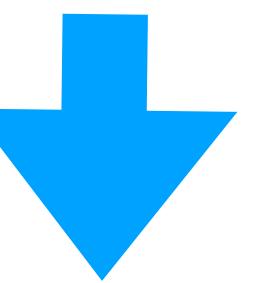
webpack
MODULE BUNDLER



Vite

```
✓ DONATION-SVELTE-06-CHARTS
  > node_modules
  < src
    < lib
      > services
      > types
      > ui
        TS index.ts
        TS runes.svelte.ts
    < routes
      > charts
      > donate
      > login
      > logout
      > report
      > signup
        S +layout.svelte
        S +page.svelte
      TS app.d.ts
      F app.html
    > static
      .gitignore
      .npmrc
      .prettierignore
      .prettierrc
      O eslint.config.js
      N package-lock.json
      N package.json
      M README.md
      S svelte.config.js
      TS tsconfig.json
      V vite.config.ts
```

npm run build



```
< .svelte-kit
  > generated
  > output
  > types
  TS ambient.d.ts
  TS non-ambient.d.ts
  TS tsconfig.json
```

- Code Optimization:
 - Minifies JavaScript, CSS, and HTML
 - Removes development-only code
 - Optimizes assets and dependencies
- Bundling:
 - Combines all your source files into optimized bundles
 - Handles code splitting for better performance
 - Processes and optimizes static assets (images, fonts, etc.)
- SvelteKit-Specific Processing:
 - Builds both client-side and server-side code
 - Creates server endpoints and API routes
 - Generates static pages where possible
 - Sets up proper routing for the production environment⁷

Modern web apps shipped faster

An intuitive Git-based workflow and powerful serverless platform to build, deploy, and collaborate on web apps

Deploy to Netlify
using Github

Create a new site

From zero to hero, three easy steps to get your site on Netlify.

1. Connect to Git provider 2. Pick a repository 3. Site settings, and deploy!

Site settings for wit-hdip-comp-sci-2020/donation-svelte

Get more control over how Netlify builds and deploys your site with these settings.

Owner: Eamonn de Leastar's team

Branch to deploy: master

Basic build settings

If you're using a static site generator or build tool, we'll need these settings to build your site.

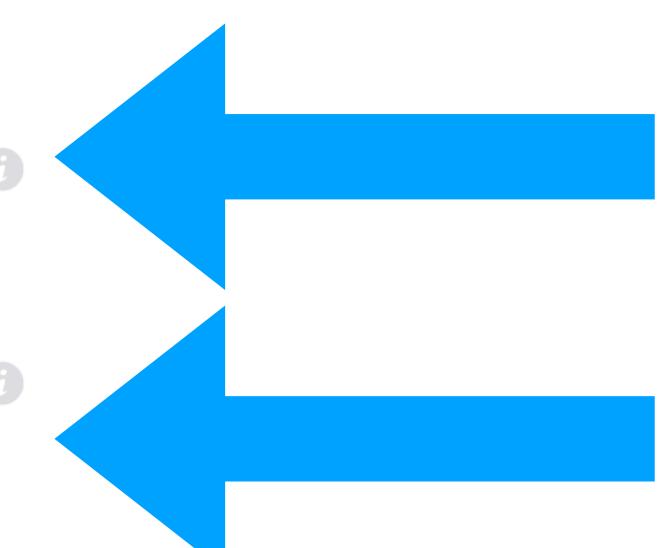
[Learn more in the docs ↗](#)

Build command: npm run build

Publish directory: dist

Show advanced

Deploy site



- Deploy directly from github
- Driven from Netlify web dashboard
- Specify:
 - Build command
 - Build folder

Develop. Preview. Ship.

Deploy to Vercel

Import Project

Importing `donation-svelte` from master into  Eamonn de Leastar

PROJECT NAME: donation-svelte

FRAMEWORK PRESET:  Svelte

ROOT DIRECTORY: ./ Edit

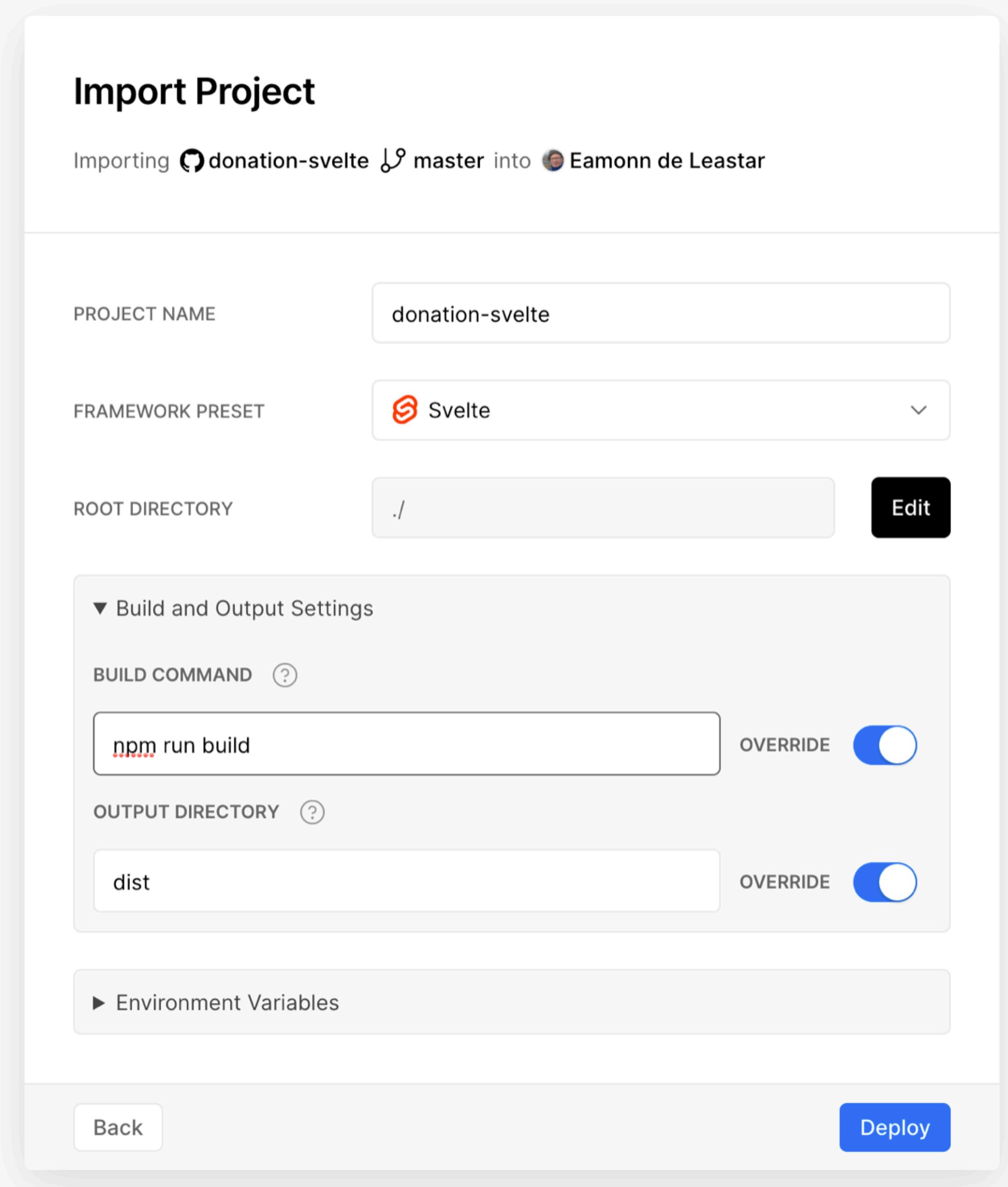
▼ Build and Output Settings

BUILD COMMAND: `npm run build` OVERRIDE

OUTPUT DIRECTORY: `dist` OVERRIDE

► Environment Variables

Back Deploy



- Deploy directly from github
- Driven from Vercel web dashboard
- Specify:
 - Build command
 - Build folder

Client Side Rendering (CSR)

- Client-side rendering (CSR) is the generation of the page contents in the web browser using JavaScript.
- Server-side rendering (SSR) is the generation of the page contents on the server.
- SSR is significantly superior for SEO (Search Engine optimisation)



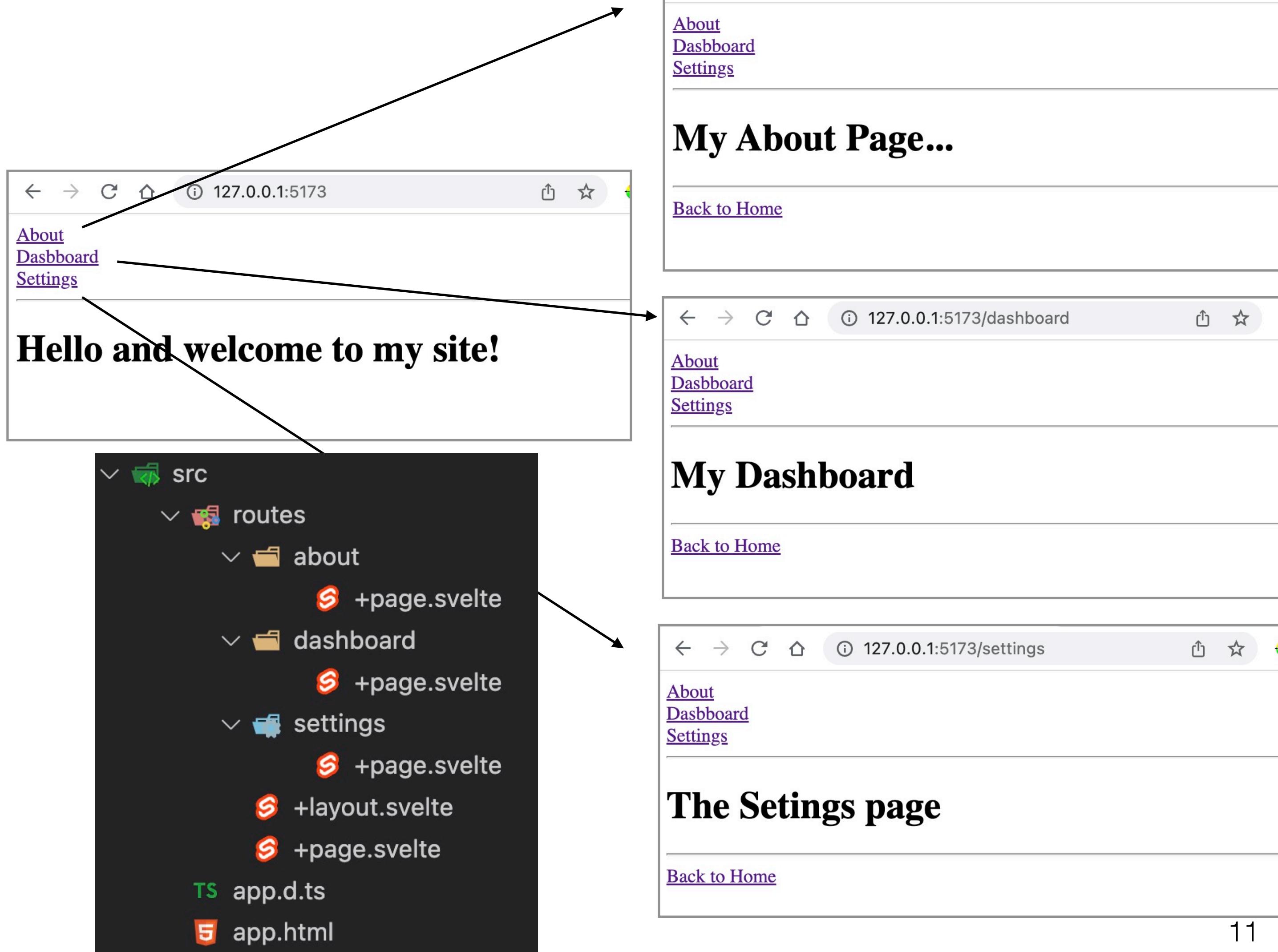
Three separate browser windows are shown, each displaying a different page:

- The top window shows the "About" page with the title "My About Page..." and a "Back to Home" link.
- The middle window shows the "Dashboard" page with the title "My Dashboard" and a "Back to Home" link.
- The bottom window shows the "Settings" page with the title "The Setings page" and a "Back to Home" link.

Server Side Rendering (SSR)

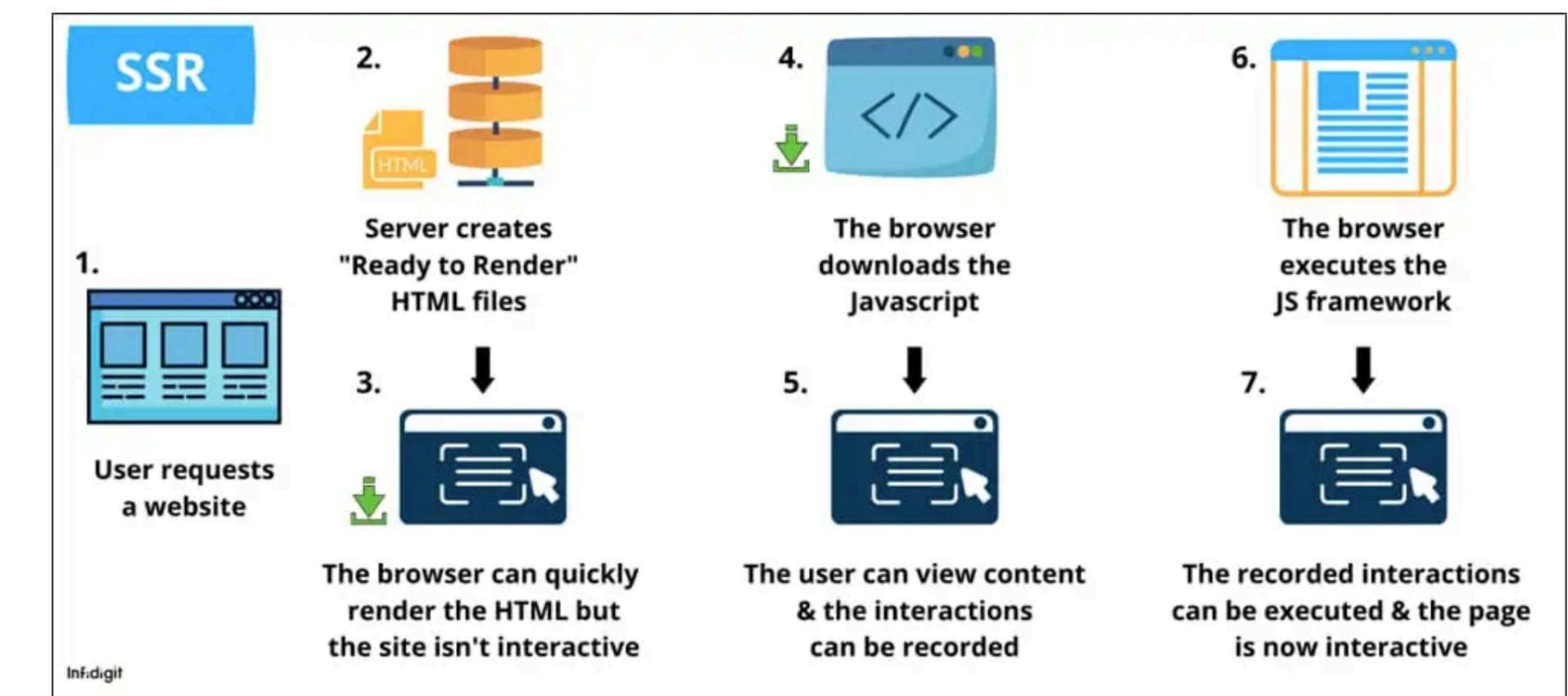
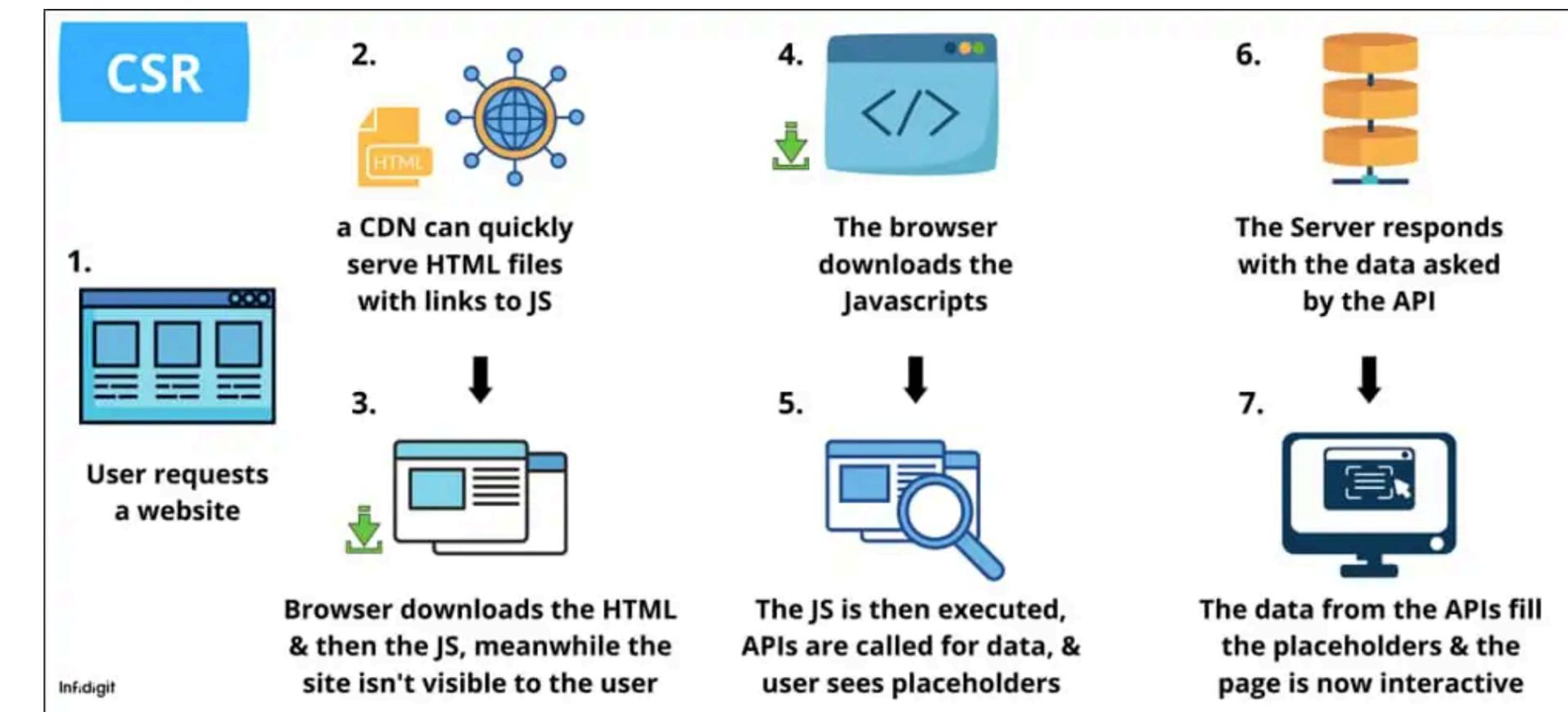
- By default, when you navigate to a new page (by clicking on a link or using the browser's forward or back buttons), SvelteKit will intercept the attempted navigation and handle it.
- SvelteKit will then update the displayed contents on the client by rendering the component for the new page
- This process of updating the page on the client in response to attempted navigation is called client-side routing.

Client Side Routing



SvelteKit Rendering Model

- By default, pages are also server-side (SSR) rendered when you first visit a page.
- When you navigate back to a page that has already been visited, SvelteKit will use client side rendering (CSR)
- This will improve the perceived performance of the app (it will feel like an SPA), but retain potential SEO benefits of an MPA model
- This behaviour can be customised -
 - Full CSR (SPA like)
 - Full SSR (MPA like)
 - Per page behaviour CSR/SSR



SvelteKit Deployment

- Sveltekit requires an ‘Adapter’ in order to deploy to specific platforms:
- You select and install the adapter for the platform your are deploying to
- This adapter will automatically handle all SSR/CSR configuration

- `@sveltejs/adapter-cloudflare` for Cloudflare Pages
- `@sveltejs/adapter-cloudflare-workers` for Cloudflare Workers
- `@sveltejs/adapter-netlify` for Netlify
- `@sveltejs/adapter-node` for Node servers
- `@sveltejs/adapter-static` for static site generation (SSG)
- `@sveltejs/adapter-vercel` for Vercel

- Install the Netlify Adapter

```
npm install -D @sveltejs/adapter-netlify
```

- Include this adapter in the svelte config
- Include build instructions for netlify

```
// import adapter from "@sveltejs/adapter-auto";
import adapter from '@sveltejs/adapter-netlify';
import { vitePreprocess } from "@sveltejs/vite-plugin-svelte";
/** @type {import('@sveltejs/kit').Config} */
const config = {
    // Consult https://svelte.dev/docs/kit/integrations
    // for more information about preprocessors
    preprocess: vitePreprocess(),

    kit: {
        // adapter-auto only supports some environments, see https://svel
        // If your environment is not supported, or you settled on a spec
        // See https://svelte.dev/docs/kit/adapters for more information
        adapter: adapter()
    }
};

export default config;
```

```
[build]
command = "npm run build"
publish = "build"
```

SvelteKit Netlify Deployment

Import into Netlify Dashboard

Sites

Open Source Current usage period (Mar 27 to Apr 26)

Bandwidth 16 GB/1 TB Build minutes 78/25,000 Concurrent builds 0/3 Team members 6

Add new site

Import an existing project Start from a template Deploy manually

Search sites

Owned by This team Created by Anyone Access Any Edit filters

full-stack-1-2023 Owned by Tutors team Deployed from GitHub Published on Apr 14 (a day ago)

tunepal-prototpe-site Owned by Tutors team Manual deploys Published on Apr 14 (a day ago)

wit-hdip-comp-sci-2024-full-stack-1 Owned by Tutors team Deployed from GitHub Published on Apr 12 (3 days ago)

Eamonn de Leastar edeleastar@wit.ie

Let's deploy your project with...

Review configuration for donation-svelte-07-maps

Deploy as edeleastar on Tutors team team from master branch using `npm run build` command and publishing to `build`

Team: Tutors team

Site name: donation-svelte-maps

Git repository: donation-svelte-07-maps

Deploying: master

Build command: `npm run build`

Published to: build

Build settings

Specify how Netlify will build your site.

Learn more in the docs ↗

Branch to deploy: master

Base directory:

The directory where Netlify installs dependencies and runs your build command.

Build command: `npm run build`

Examples: `jekyll build`, `gulp build`, `make all`

Publish directory: build

- Will be automatically configured

- Deploy!

app.netlify.com/sites/donation-svelte-maps/deploy/67fe903943220109864c0d3b

Sites / donation-svelte-maps

Site overview Site configuration

Deploys

- Preview Servers
- Logs
- Metrics
- Domain management
- Forms
- Blobs

Published deploy for donation-svelte-maps

Netlify's robots are busy building and deploying your site to our CDN.

Today at 5:58 PM Production: master@HEAD

Functions

Cancel deploy Deploy settings

Fancy a game while you wait?

We're on a mission to find the cutest emojis.

Play game

Deploy log

Step	Status
Initializing	Complete
Building	Complete
Deploying	Complete
Cleanup	Complete
Post-processing	In progress

```
196 5:59:12 PM: Post processing - redirect rules
197 5:59:12 PM: Post processing done
198 5:59:12 PM: Section completed: postprocessing
199 5:59:13 PM: Site is live ✨
200 5:59:12 PM: Skipping form detection
201 5:59:12 PM: Post processing - header rules
```

Donation

News Support

Donate Report Charts Maps Logout [Homer Simpson]

Make a Donation

Donations to Date

Please Donate

Enter Amount: 033

Select Payment Method: paypal direct

Select Candidate: Simpson,Maggie

Lat: 54.160858 Lng: -7.15242

Donate

Thanks! You donated 33 to Maggie Simpson

Donations to Date

Donor	Amount	Method	Candidate	Donor
Simpson,Lisa	40	paypal	Simpson,Bart	
Simpson,Maggie	90	direct	Simpson,Marge	
Flanders,Ned	430	paypal	Flanders,Ned	Simpson,Homer
	33	direct	Simpson,Maggie	Simpson,Homer

Please Donate

Amount	Method	Candidate	Donor
40	paypal	Simpson,Lisa	Simpson,Bart
90	direct	Simpson,Lisa	Simpson,Marge
430	paypal	Flanders,Ned	Simpson,Homer
33	direct	Simpson,Maggie	Simpson,Homer

Deploy Front End



Apps



Deploying donation-svelte
to Netlify & Vercel services