



Progress on rncode

<https://rncode.herokuapp.com/>

General Approach

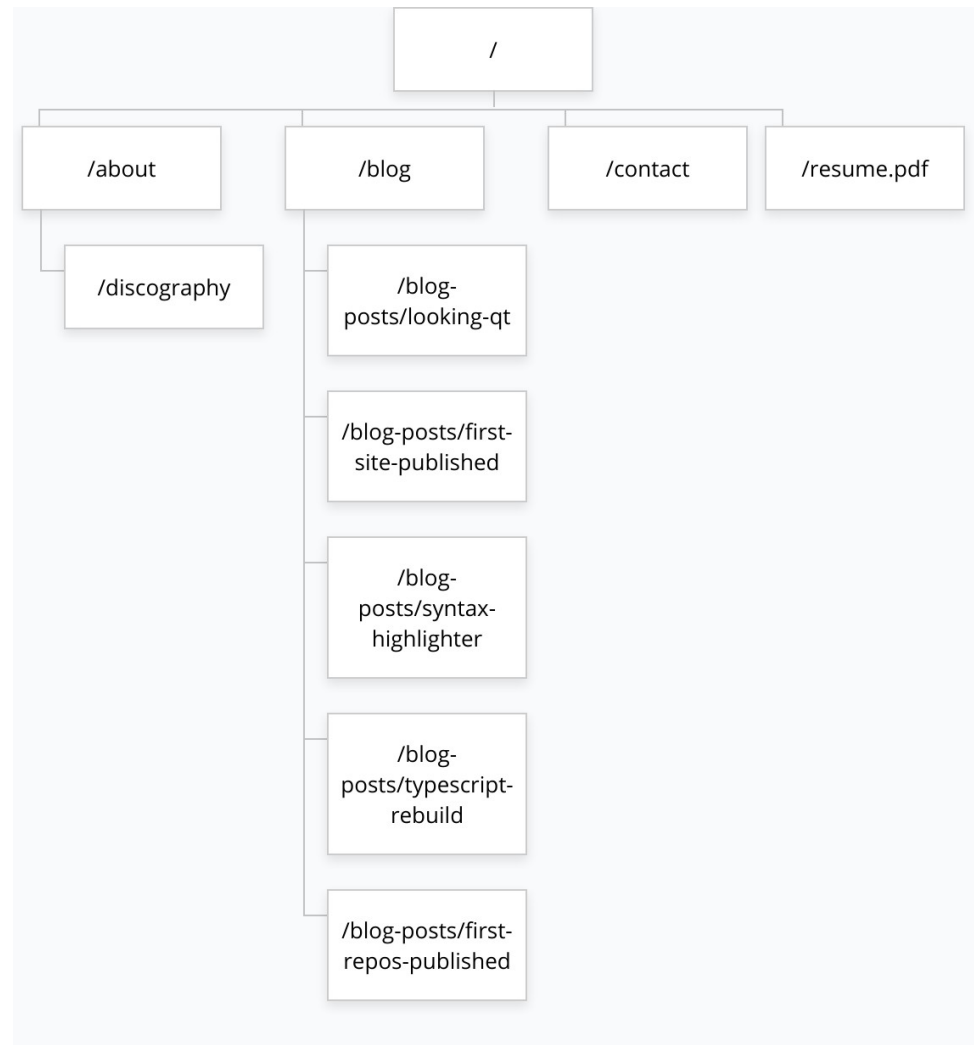


- Focusing on mobile-first
- Minimal engaging design for primarily textual content
- Some 'sprinkles' (animations + JavaScript)
- Not going overboard with features – just focusing on implementing those that are necessary at this stage as well as I can (to demonstrate my skills)

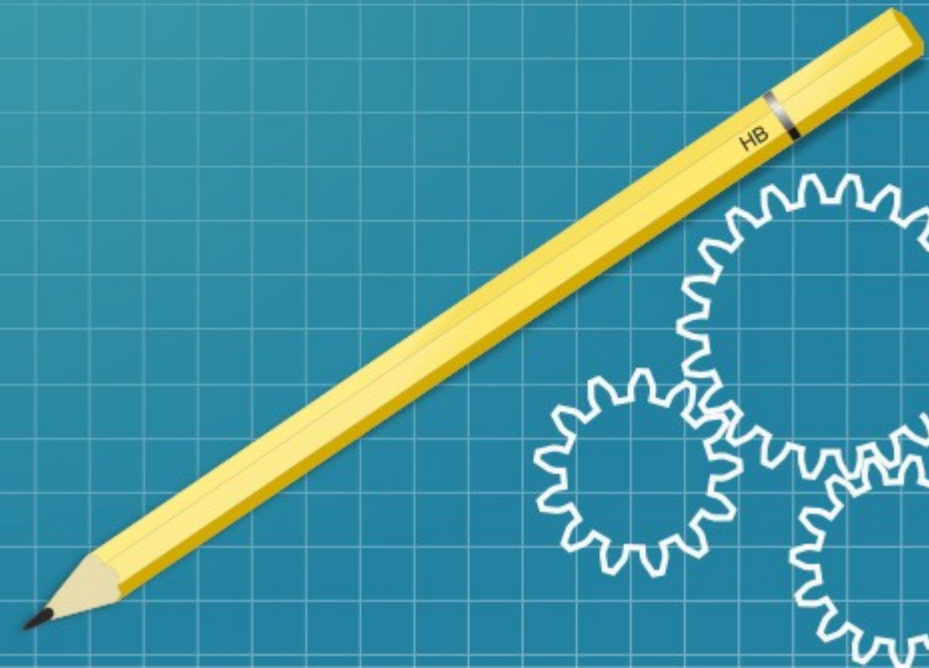
Site Map



- Simple and logically organised site structure
- Easy navigation – links to the home page, /about, /blog, and /contact will be present in the header and footer navs on all pages
- Standard URLs
- Each blog post will contain a link to /blog (the blog index) at the end of its content



Layout Overview

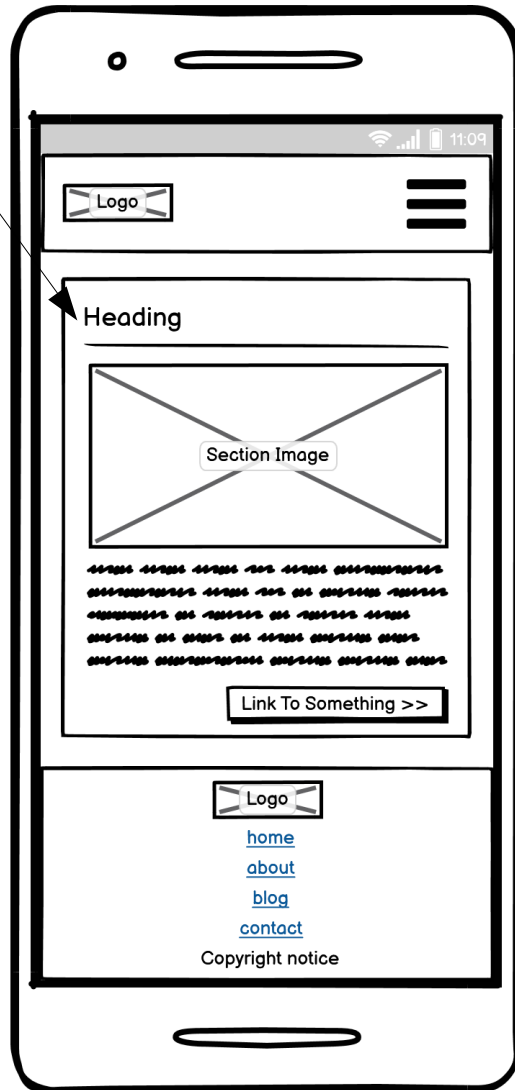


Mobile

All sections across site have
The same styling

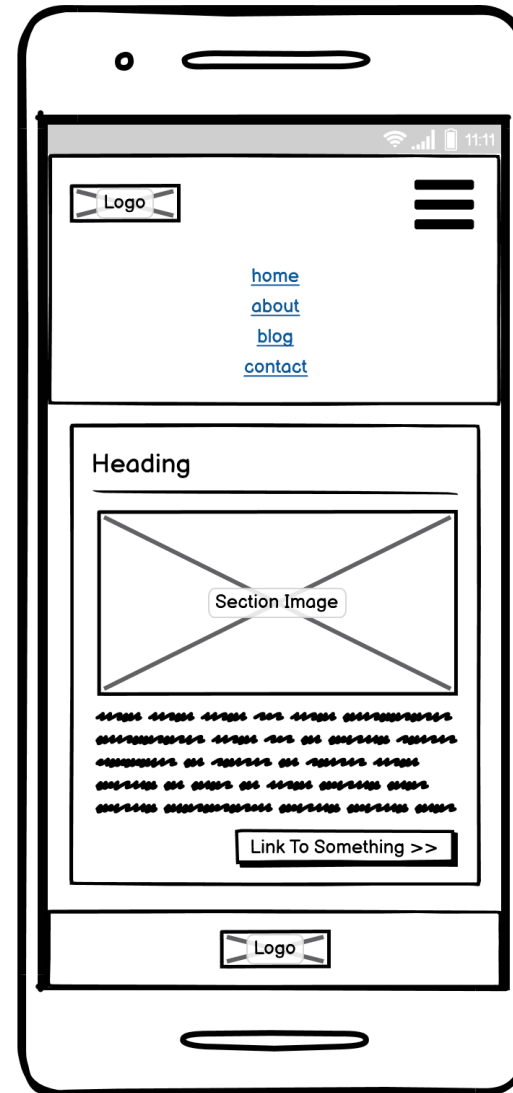
(Menu closed)

Footer is
pushed to
page bottom
(‘body’ is flex
and ‘main’ has
‘flex-grow: 1;’)



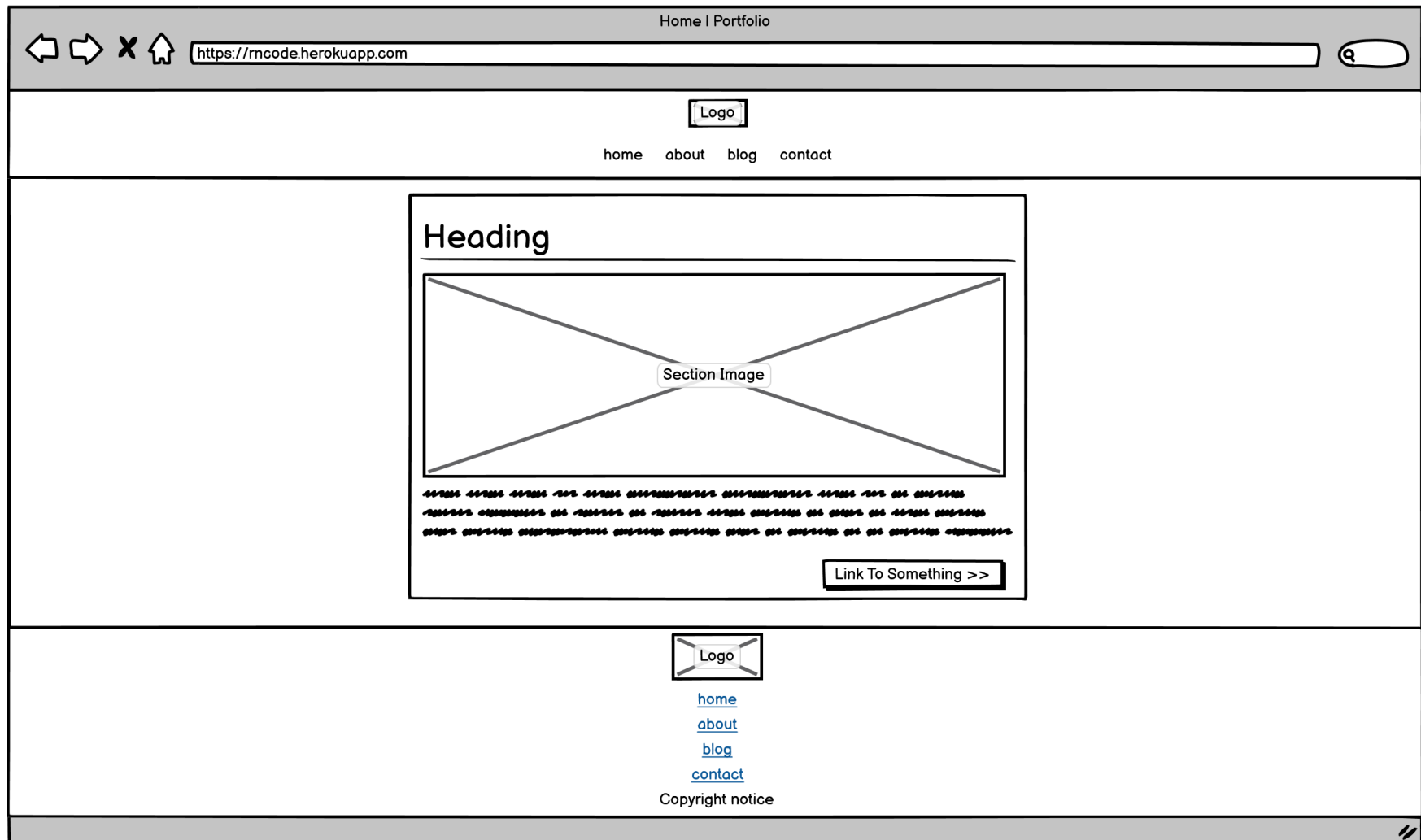
(Hamburger
will verticalise)

(Menu open)



Desktop/Tablet

Header content is center aligned vertically – nav visible always



Notes On Layout



- All sections have consistent 'card-like' styling
 - Border radius (5px)
 - Box shadow
- Sections within 'main' are centered along the vertical axis and have a bottom margin to for spacing.
- Using combination of 'width: 100%;' and 'max-width: 800px;' so that the sections are responsive to the size of their container ('main'), but also do not exceed a threshold.
- I like having main content centered when it is primarily textual.
- Some sections do not contain images or links – just text



No Tablets?

- The use of Flexbox and the combination of 'width' and 'max-width' eliminates the need for tablet breakpoints in my design

Page Section Breakdown



- Home page
 - Hi, I'm Rob
 - My interests
 - Current learning →
 - Current job ↓ resume.pdf
- About
 - My Story
 - Education
 - What I've Learned
 - I Make Music Too ↙
- Contact
 - Contact (includes a form for messaging me)
 - Connect
 - Github
 - LinkedIn
- Discography
 - Single discography section (an index – similar to blog post index on blog page)
- Blog
 - Single blog post index section
- Blog post (x5)
 - Single section containing a title, an image, the date + time, written content, and code snippets

NOTE: the arrows indicate links in sections on some pages



More Wireframes...

****NOT INCLUDED****

- home page
- about page

REASON: they share the same site-generic design, as outlined in slides 4 – 7.

NOTE: all sections to be included in the home and about pages have been detailed in the previous slide.

Contact Page

Contact | Portfolio

https://mcode.herokuapp.com/contact

Logo

home about blog contact

Contact

Name

Email

Message

Reset Send

Logo

[home](#)
[about](#)
[blog](#)
[contact](#)

Copyright notice

The only difference between the desktop/tablet and mobile views is that the header will change – all components on this page resize responsively

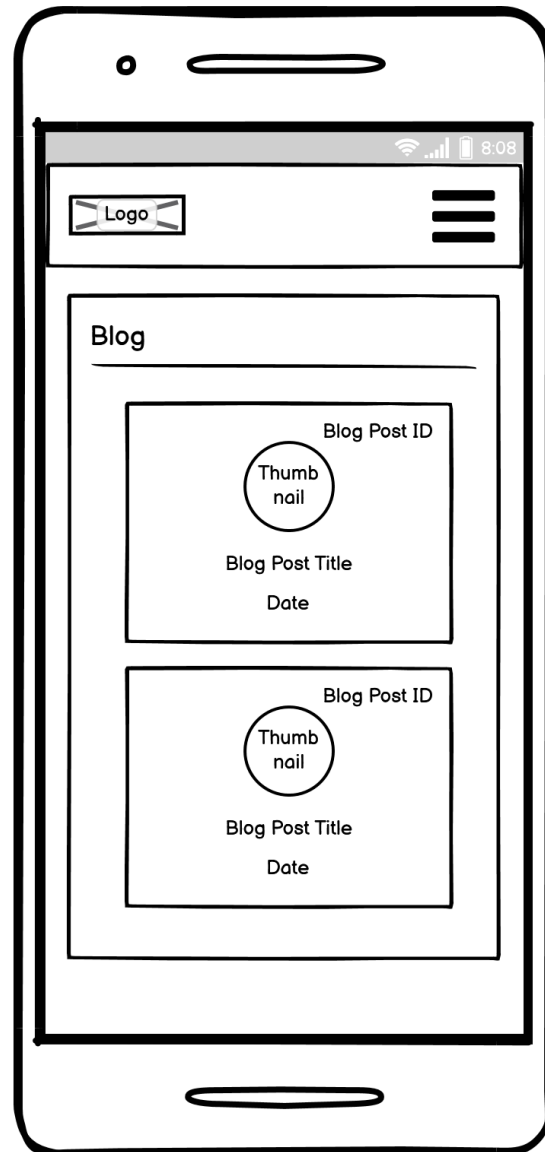
Additional section below the contact form to contain 'connect' (socials + GitHub) links



404 Page

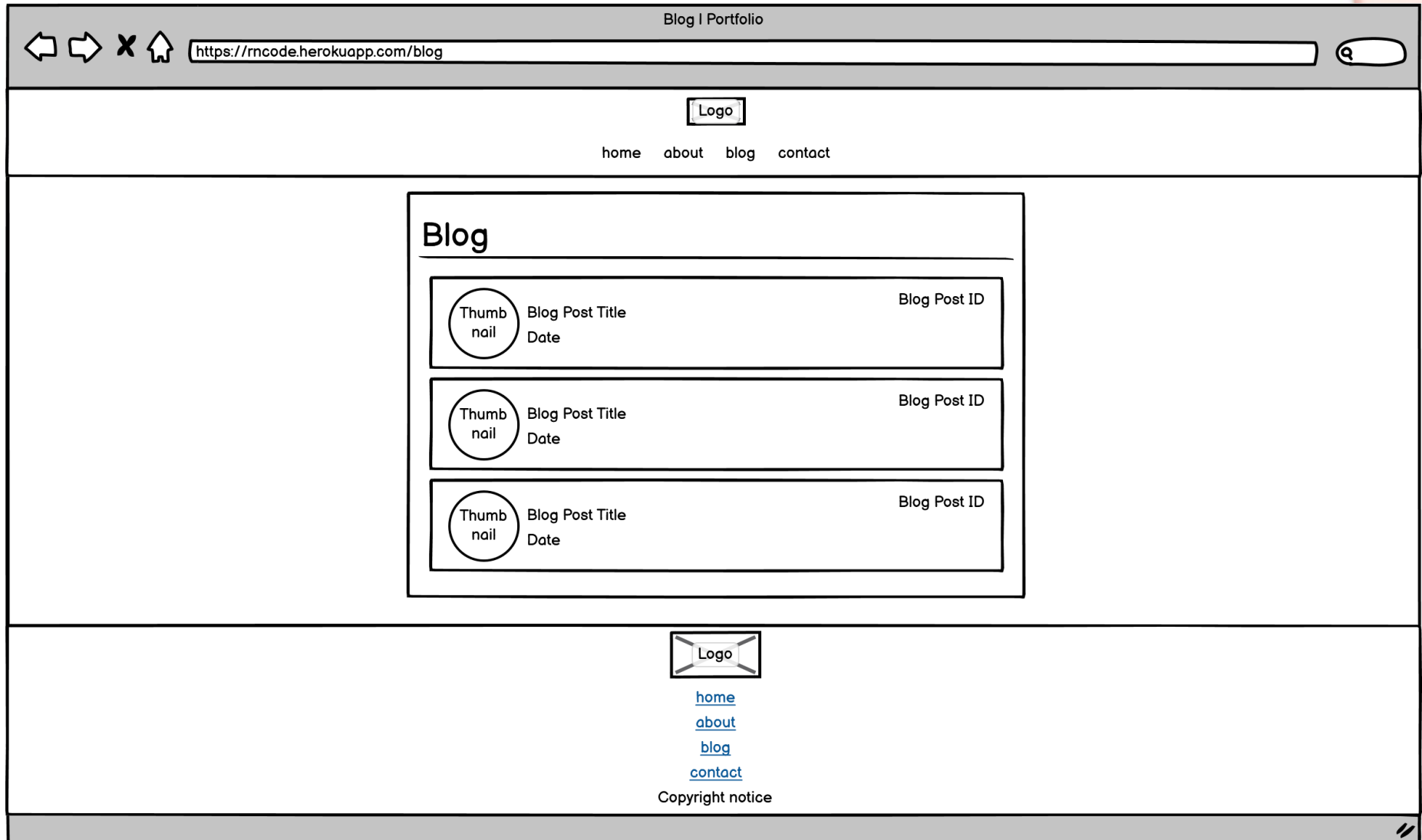
- To include a section that ‘explains the situation’ to the user...
- And! A contact form for the user to send a message if there’s a broken link on the site
- Maintains the same global design

Blog Page Mobile



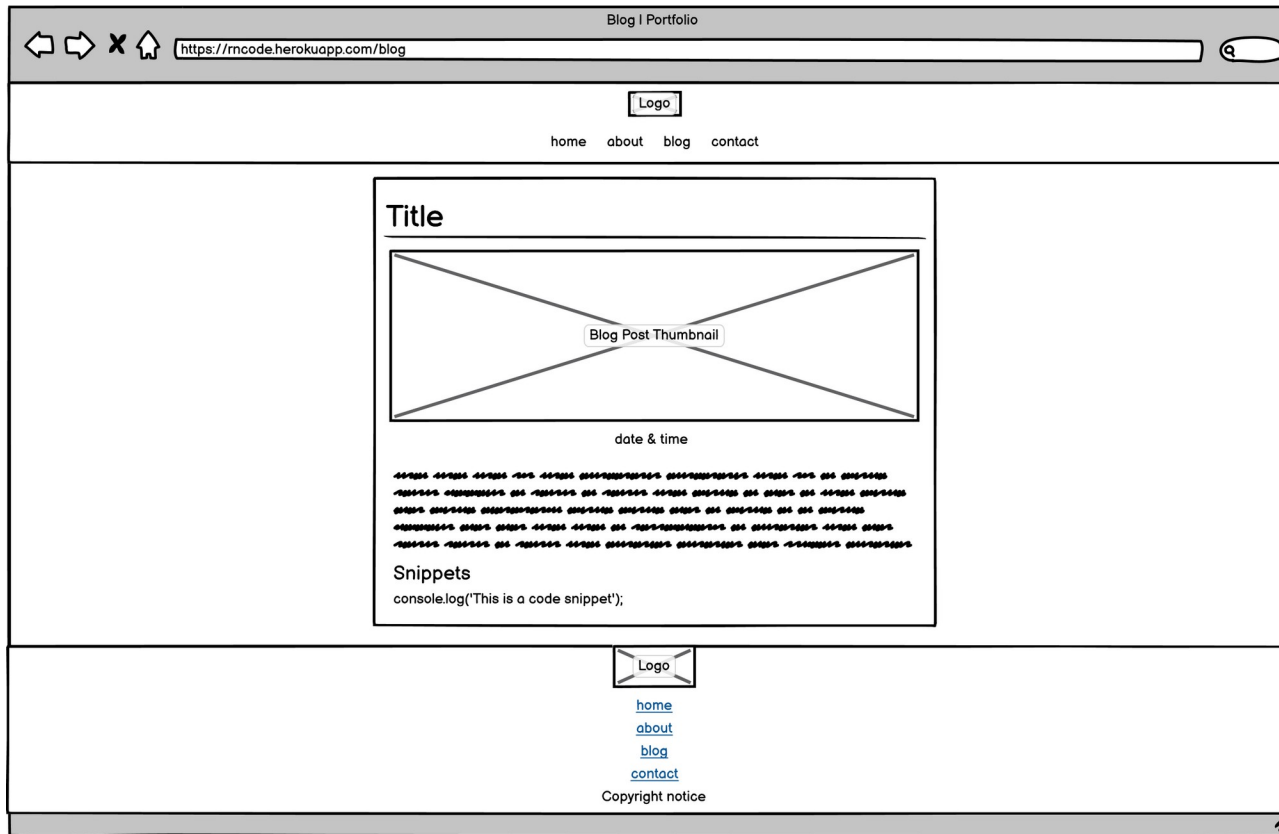
**The implemented blog page contains 5 blog post links

Blog Page Desktop/Tablet



**The implemented blog page contains 5 blog post links

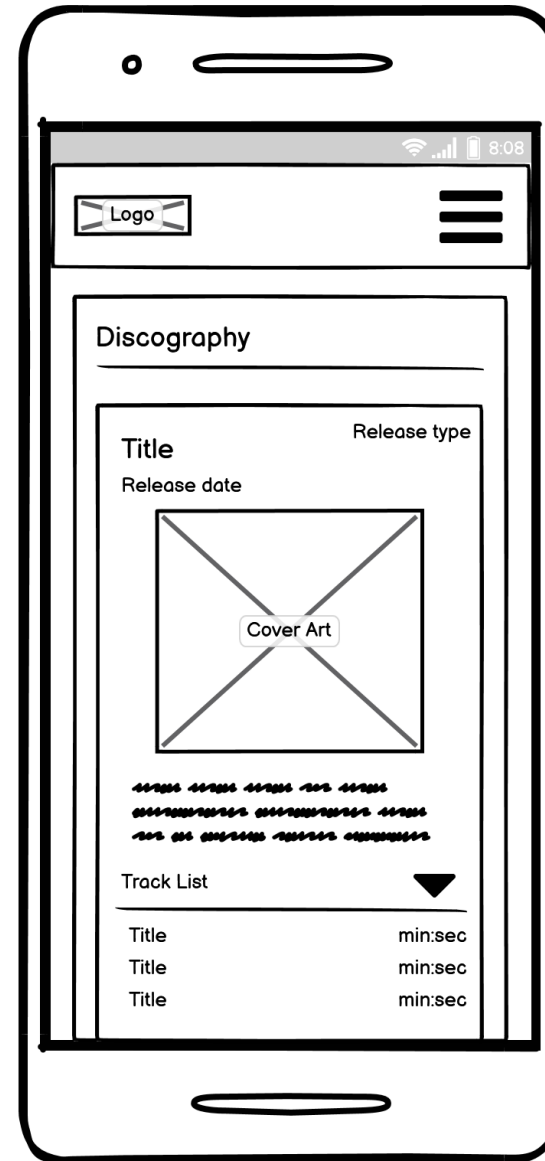
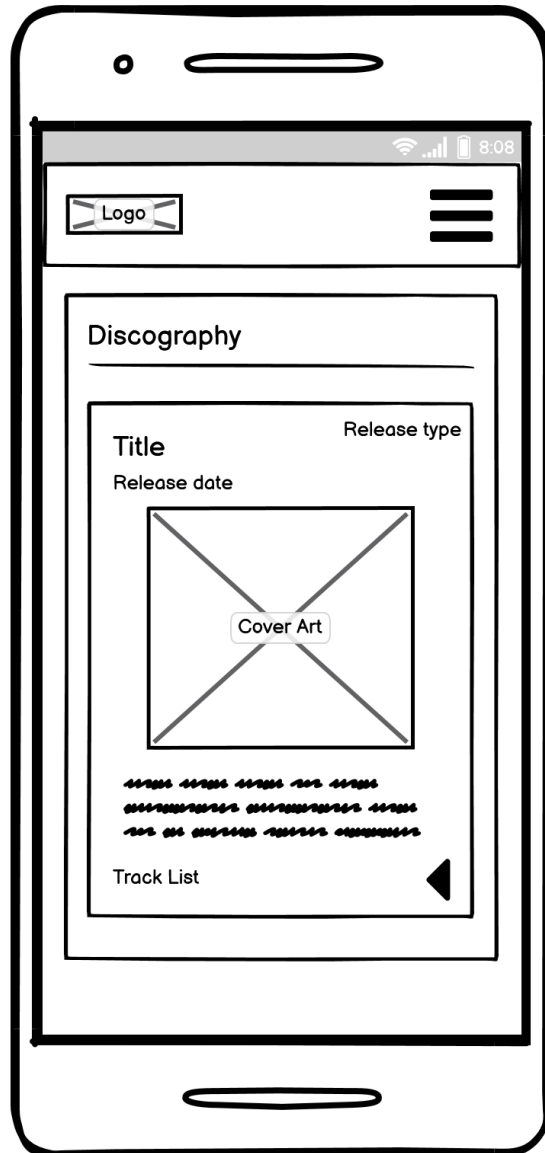
Blog Post Page



The only difference between the desktop/tablet and mobile views is that the header will change – all components on this page resize responsively

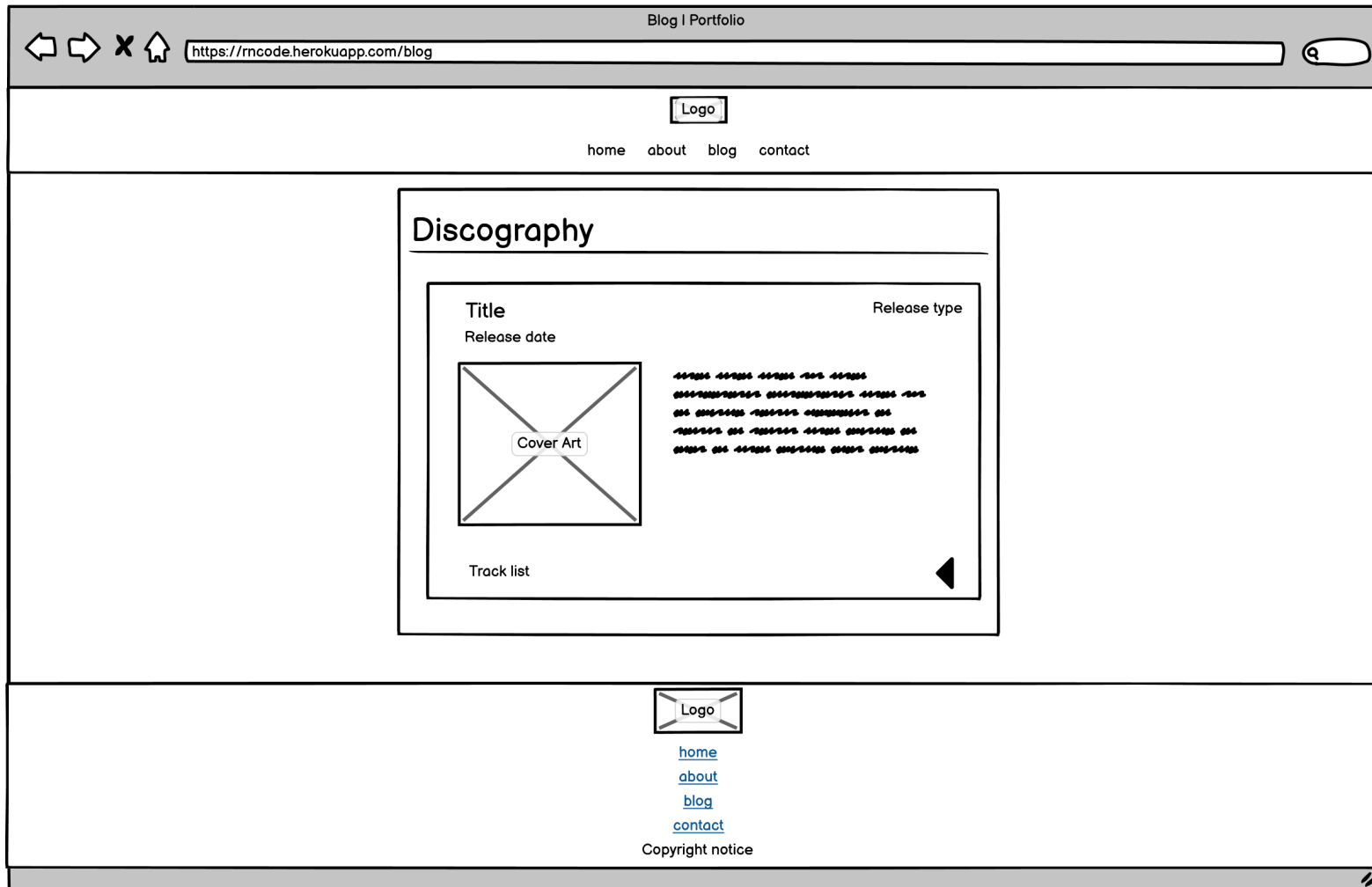
- Each blog post will include code snippets from the project discussed that will be highlighted with highlight.js (CDN import)

Discography Page Mobile

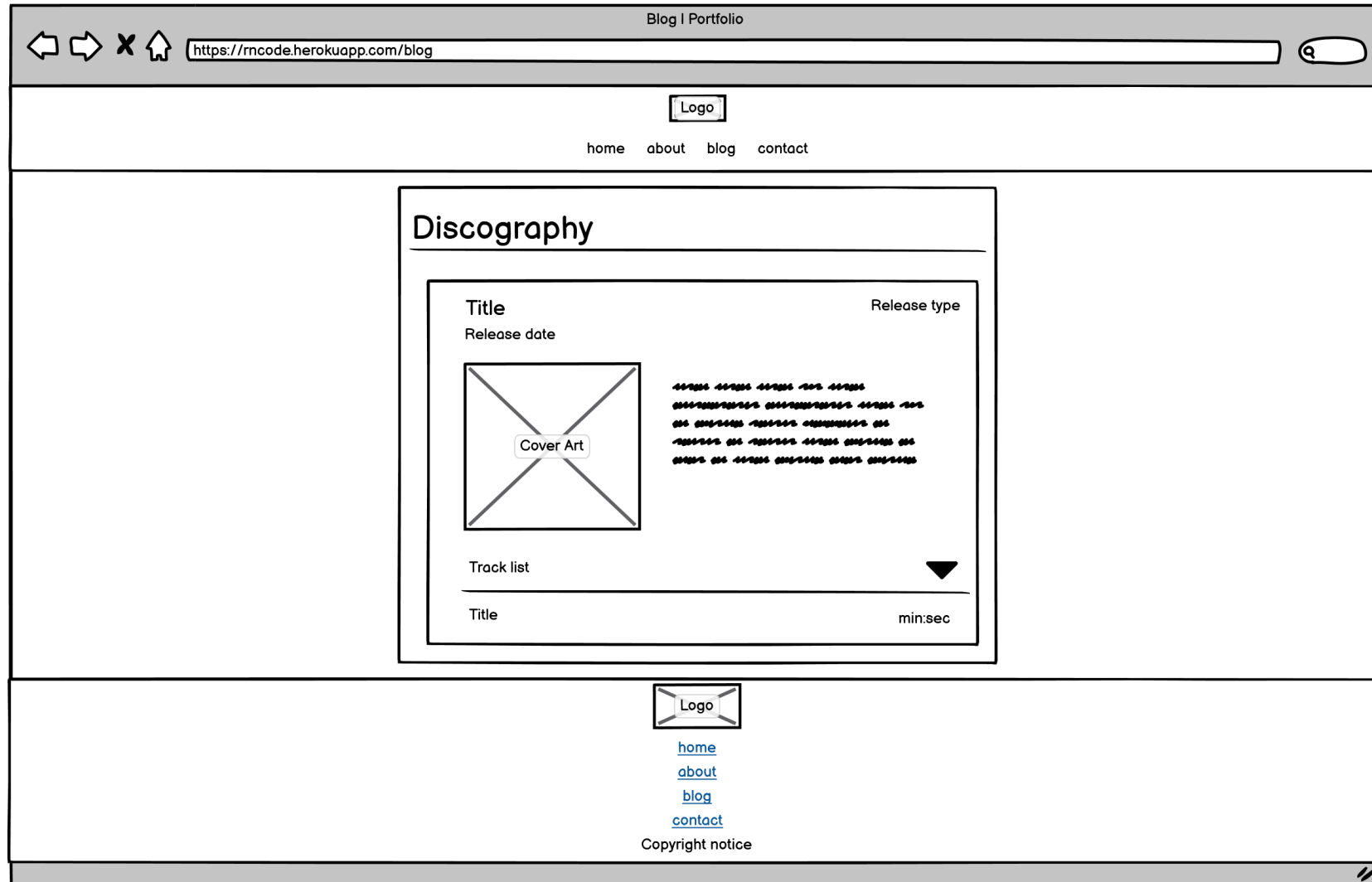


(on hover)

Discography Page Desktop/Tablet



Discography Page Desktop/Tablet (on hover)



Components

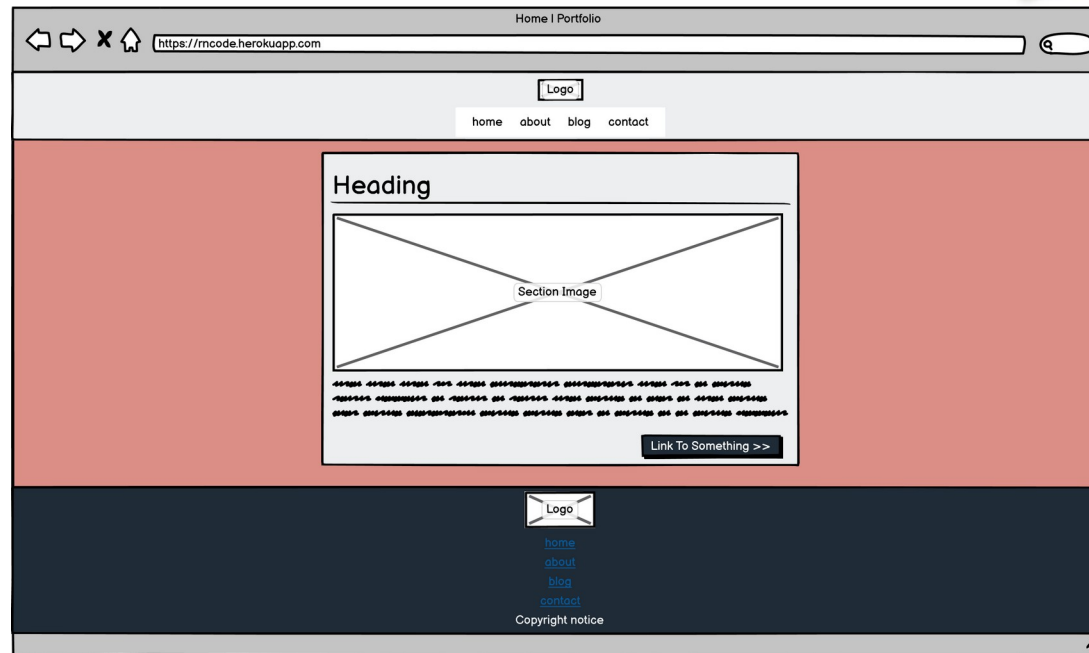


- Global:
 - Header (G)
 - Hamburger (mobile)
 - Menu (G)(mobile)
 - Fixed background (G)
 - Main? (more a component container, but present on all pages)
 - Section (G?)
 - Footer (G)
- Page specific:
 - Blog post link (G)
 - Blog post code snippet
 - Contact form
 - Message card (G)
 - Sending spinner (G)
 - Discography item (G)
 - Connect link (G)

Colour Scheme

- Simple & consistent across all pages in the site
- I am using pastel colours – off white, red, & navy
- The red features on a fixed background behind the main content
- Also incorporating a texture on the red background

Colour Scheme Mockup



- Navbar is the same colour as the header
- Header text and logo colour is the same as the footer bg
- Footer text and logo colour is the same as the header bg
- Section bg is the same as the header bg, and section text is the same as the footer bg
- All section links and buttons will have a slightly lighter hue of the footer bg – transitioning to footer bg on hover

Fixed background



- The red will be overlayed with this texture
- This colour and texture will be the background of a fixed 'div' that is only visible behind the 'main' element on each page.
- Content moves over background on scroll
 - looks nice (adds more character)
 - engaging effect

Font



- I decided to go with a monospace Google font because... programming (and I reckon they look good on websites)

Roboto Mono
Christian Robertson

Variable

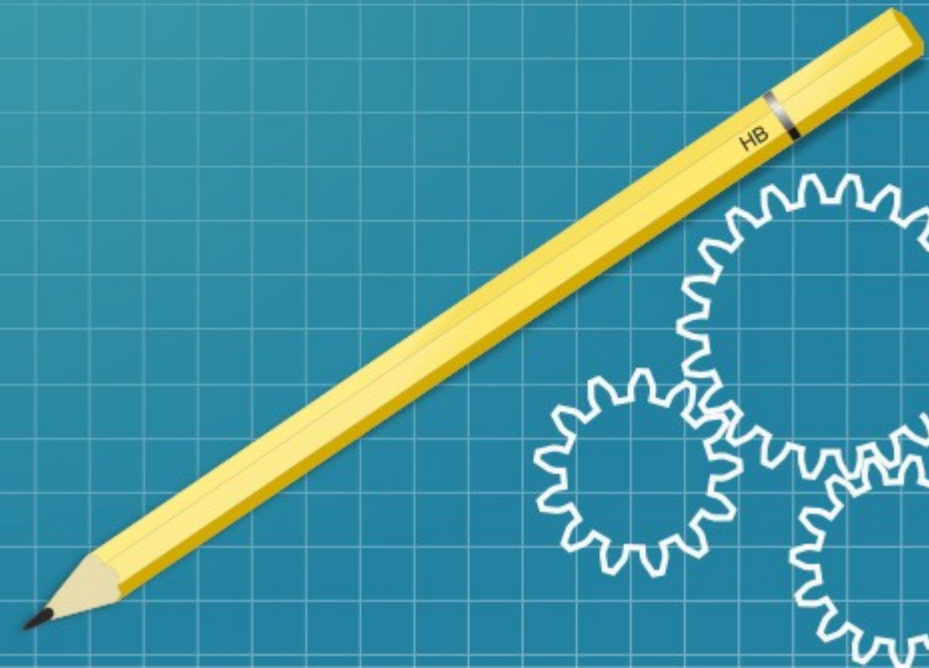
Almost before we
knew it, we had
left the ground.

Additional Stylistic Features



- Emojis in headings (borrowing from GitHub)
 - Combined with images reduces theme uniformity (irregularities catch the eye)
- Header nav links will display bottom border on hover
- Footer nav links will display lighter bg on hover
- Blog post links will invert text and bg colours on hover and 'jiggle'
- ^^ Same for 'connect' links on contact page

Review of Development Process





- No major issues to report.
- The only problem I encountered was setting up SSH keys for two GitHub accounts on the one machine – I ended up just using SSH for my CoderAcademy account and HTTP for my personal account.
- Build process was fairly smooth as I have experience with other build tools (g++, MinGW, Mono, and webpack)
- No ethical issues other than avoiding publishing my personal contact details online (hence the contact forms)
- I enjoyed the whole process, but particularly using Sass and adding JavaScript for the menu and contact forms
- Favourite (coolest) part was fine tuning my build process using npm scripts (concurrently, cross-env, nodemon, sass)

Thank you for your time!

