

1. INTRODUCTION

The main purpose of our application is to quickly create landing pages that effortlessly showcase stakeholders' projects and encourage future contributors to donate.

The main element of the template we created is the ability for users to contribute to the collection. This is made possible for them by the CTA buttons "Contribute" on the page. If the user does not have the polkadot-js wallet plugged in, they will receive a message about the need to install the extension after clicking on the contribution button. This extension provides communication with the blockchain. It is possible to install the extension only in Chrome and Firefox.

Once the extension is installed - then, after clicking the CTA button "Contribute", the user will be shown a modal to be filled while still being on the collection project page. Then he will be redirected to the modal of the polkadot-js extension.

The founder of the collection, after downloading the code which will be used to set up a crowdfunding project page will also receive instructions on how to set up their collection.

2. BENCHMARKING

Before creating our application, we started with the direct and indirect analysis of the competition. We analyzed existing websites on the market to understand the design patterns for the industry and the product. This allowed us to find a way to stand out from the competition. We also wanted to take inspiration from other products. So we worked in Figma, adding screenshots of different products, writing out notes, and pasting design elements we found interesting to generate insights as we designed.

We looked at several types of pages - these were typically crowd loan project pages, about what we were to design. We also looked at sites that included crowdfunding - but they were not ultimately the same thing we were creating but were an expanded version of crowd loan.

We also dived into Tokenomics in some depth.

3. DESIGN DIRECTION PROTOTYPE

After benchmarking existing products, we decided to find a design direction. We usually do this by basing it on products already in use, but we also do it on concepts - so-called concept shots, on Dribbble and Pinterest. We looked there for more daring designs. All because our target audience is *tech-savvy* - who are up to date with the latest technologies and appreciate unorthodox and new designs.

Our work on the design direction was a process of gathering many different ideas, brainstorming, collecting a lot of screenshots, and a lot of inspiration in terms of using colors, typography, and graphic elements that make the site stand out - key visuals.

Then, we searched among these inspirations for solutions that best suit our users and their expectations. To highlight potential design development paths, we have created two main proposals.

The first proposal was more daring - we used a dominant color in the background, rounded buttons, and two types of fonts:

- the font for headlines that is more decorative;
- body text - that is more readable.

In the second proposal, we opted for a minimalist design with a bright, clean background and subtle gradients.

Web3 Foundation chose the second option, which we also recommended because the main goal was to create a theme that would be reusable by design teams. When we have a minimalist design and one guiding color that we use in the buttons and graphic elements - it allows the user to make and redesign a nice user interface, but in a fairly simple way, because we only change the colors of the buttons and graphic elements, not the whole background.

In the first proposal, we opted for a background in a darker color. If someone wanted to use a template in which they changed the color of the font - there would then be a good chance that

this original design might not be accessible and usable because the colors that would be used might not be contrasting enough.

The main goal is to design the stakeholder's site with her or his branding, colors that relate to the brand, or a specific design.

In preparing the *Style Guide*, we focused on checking each component - on its accessibility - whether the contrasts are appropriate, whether the fonts are the right size, whether there are different button states, how the hover buttons, the active buttons, the disabled buttons works - the main version, which is created as a template, is ready to use or to be reworked to suit stakeholder's needs - in Figma or Astro, both which we focused on.

4. ASTRO / REACT

Our template is created in Astro, using React. Developers can download the template in code form and change the modules and content on it. This template is also directed toward more technical users. It is highly customizable - the founders of the collection can customize it to their needs, flip sections, remove them, and change their layout and order;

There are several proposed sections:

- a timer indicating how much time is left until the end of the collection, the projected amount of the collection, after which the collection ends, and how much has been deposited so far;
- a section with a logo, showing which firms support the founder of the collection;
- a section about the project - those setting up the collection can describe their project in placeholder text. The placeholder can also be used to describe how to donate. The founder of the collection can substitute copywriting - we suggested our template as an example;
- a section about the token - statistical information, how much percentage is in the ecosystem, the total value in conversion in the polkadot token;
- a roadmap - fully customizable; the founder of the collection can add bullet points, indicating where the project is on the timeline;

- A table that displays recent payments. They are displayed in the template of the test network - when the collection's founder changes it to a real network, to a real blockchain that is live - this table will pull real data then. This table also includes a link to directly open the donation history on polkascan;
- placeholders for additional links;
- FAQ - the founder of the collection can add her or his questions and answers;
- an additional timer showing how much time is left until the end of the collection;
- Footer, containing placeholders for links to external portals (Facebook, Twitter, etc.).

5. WEBFLOW

We created a one-pager in both React and Webflow versions.

Unfortunately, there was a problem with integrating the external API in Webflow. Polkadot doesn't seem to host versions of their libraries on CDN (*Content Delivery Network*), so we could not include it in Webflow. Even if we implemented a piece of code from React onto Webflow, we would still have to host it.

We learned during the process that Webflow is perfect as a simple one-pager that doesn't pull data from external APIs and doesn't have custom-written code in JavaScript.

The prepared Webflow template is the equivalent of the project implemented in React. The difference between those two is that on Webflow there is no contribution flow built in - all "Contribution" CTAs show a pop-up informing that if the user wants to contribute, he has to go to the external polkadot-js extension manually. Such a solution, directing the user to an external service, can still find a use for non-technical project founders who want to set up a collection after all.