Milestone Guide — Implement a Website Interface Using CSS Animations

How to use this document

On this page, you'll find an example of how to break your development project down into steps. You'll find:

- recommendations for successfully completing each step.
- common problems and important points to bear in mind.
- an estimate of project progress.

This breakdown is just a suggestion to help you plan your project. You don't have to follow the steps in this precise order.

Bear in mind that your progress through the different stages is only an estimate and will vary depending on your working pace.

Step 1: Set up Your Development Environment

5% complete

Before starting the project, it would be a good idea to familiarize yourself with the different documents, particularly the creative brief. This document defines all of the web development requirements and constraints, and aids communication between different teams. Read it carefully so you can understand the challenges of the ohmyfood website.

Before starting this step, you need to:

- study parts 1 and 2 of the Manage Your Code Project With Git and GitHub course.
- study the Produce Maintainable CSS With Sass course.

Once you've completed this step, you should have:

- a project in a GitHub repo.
- a project structure containing:
    - an index.html page.
    - a folder for your assets (CSS file, images).
    - a folder for your Sass.
    - a "restaurants" folder to store your restaurant pages.

Recommendations:

- After creating the project folder, create the "ReadMe.md" file with the project name.
- Initialize the project in Git and publish the repo on GitHub.
- Create the various files needed for the project.
- Publish the website on GitHub Pages if you want to see it live.
- Remember to structure your Sass code properly, ideally in several files.

Points to note:

- Ensure that the GitHub repo is set to public so that your mentor can access the project.
- Make sure that you commit any changes to the code to your Git repository every time you complete any stage of your site development.
- Make sure that you regularly push your updates in the local Git repository to Github.

Resources:

- GitHub Pages site

Step 2: Deploy the Mobile Version of the Home Page

15% complete

Now your project is ready, you can start by developing the mobile version of the home page without any animations.

Before starting this step, you need to:

- analyze the mockups and identify:
    - the different colors used on the site.
    - the different sections on the page.
    - the different components (i.e., the interface elements that are repeated).

Once you've completed this step, you should have:

- a website home page without animations.

Recommendations:

- Consider also analyzing the desktop mockup to give you a better idea of how to arrange the sections.
- At each stage, use the HTML and CSS validators to check your code. You can do this by entering your GitHub Pages link on the validator.

Points to note:

Remember that we're using a mobile-first approach here, which means that the main CSS will be for the mobile site and we can use media queries for the desktop format.

Resources:

- Mobile-First Design: What it is and How to Implement It (Browser Stack)

Step 3: Add Animations to the Home Page

30% complete

Now you have the structure for your page, it's time to add your first animations to the project.

Before starting this step, you need to:

- study the Create Modern CSS Animations course.

Once you've completed this step, you should have:

- a home page with working animations for the mobile version.
- animations that behave in a natural way, so if there's a hover animation on an element, the animation must be reversed when the cursor is moved away from the element.

Recommendations:

For each animation you create, check if a detailed example has been provided. If not, check whether the animation is consistent with the rest of the animations on the site.

Points to note:

For the best user experience, it's important to consider the animation from all component states. For example, if we have an animation when hovering over an element, we must also have the reverse animation when the cursor leaves the element.

Resources:

- Mozilla.org: Using CSS Transitions
- MDN: Using CSS Animations

Step 4: Produce a Responsive Design for the Home Page

40% complete

Before starting this step, you need to:

- analyze the desktop version of the home page.

Once you've completed this step, you should have:

- your final home page published on GitHub Pages.

Recommendations:

- Identify the standard breakpoints in your app to avoid having multiple media queries.

Points to note:

- Bear in mind when making your website responsive that we only need to add rules that we want to update in the media queries.

Resources:

- How to Use CSS Breakpoints and Media Query Breakpoints for Responsive Design

Step 5: Implement HTML and CSS on a Restaurant Page

75% complete

After finishing your homepage, you can repeat the same process on the first restaurant page.

Before starting this step, you need to:

- analyze the restaurant page mockup for mobile and desktop.

Once you've completed this step, you should have:

- an implemented restaurant page.
- all HTML and CSS validated by W3C.

Recommendations:

- At this point, you need to follow the same process described in steps 2 to 4, but this time for the first restaurant page.

Points to note:

- Don’t forget to add an ellipsis to the names and descriptions of dishes that are too long for the mobile version.
- Pay attention to the way the animations behave. Ensure that they comply with the system specifications.

Step 6: Copy the Restaurant Page and Adapt the Content to the Other Restaurants

95% complete

Once you’ve built the first restaurant page, you can use it to build the other restaurant pages.

Before starting this step, you need to:

- finish deploying the first restaurant page.

Once you’ve completed this step, you should have:

- all website pages deployed in all formats.

Recommendations:

- The structure should be the same for all restaurants. Only the content should be different. So, you shouldn’t need to touch the CSS code or the HTML structure, you can just replace the content (image + text).

Points to note:

- After adding content for a new restaurant, if you’ve changed the HTML or CSS, make sure that the changes are synchronized across all the pages and that the original restaurant page layout and styling is still behaving as expected.
- Don't duplicate any CSS code unnecessarily. The different “restaurant” pages share the same format and CSS code, apart from the banner.

Step 7: Carry Out a Full Project Review

100% complete

Before starting this step, you need to:

- finish deploying the mockups.

Once you've completed this step, you should have:

- a completed project with fully verified deliverables.

Recommendations:

- Take a little time to run each page through the validator to check that the code meets the expected standards.
- Check how it renders on mobile, tablet and desktop to make sure it displays correctly. The display should match the mockup visuals.

Optional step: Carry out Research

For this project, we asked you to produce animations using only CSS code. If, at a later date, ohmyfood wanted to use JavaScript on the project, what advice would you give them?

To find out more, we suggest you **search for information on creating animations in the site interface using JavaScript.**

This research will help in your thought processes and make you more agile when making changes. This is an essential skill for all developers!

Project complete!