

# USYD-FSF Bootcamp

## Unit 01 Mini-Project: Landing Page

### Instructions

As a client, I want a:

- ✓ Single webpage that collects a visitor's contact information.
- ✓ Landing page to have a header and footer.
- ✓ Landing page to have an image with a caption.
- ✓ Landing page to have a contact form.
- ✓ Landing page to have a polished and accessible UI.

### Acceptance Criteria

It's done when:

- ✓ The page uses semantic HTML elements.
- ✓ The page uses universal, element, and class selectors in CSS.
- ✓ The page features at least three colors in the design.
- ✓ The page uses a single font and font family for all text.
- ✓ The page uses at least two heading elements (<h1> through <h6>).
- ✓ The header is fixed to the top of the page on scroll.
- ✓ The header contains a navigation bar with three links that display inline, including a contact link.
- ✓ If the contact link is clicked, the page jumps directly to the contact form.
- ✓ The contact form includes `input` elements for name and email.
- ✓ The contact form includes a Send button.
- ✓ The image includes a descriptive `alt` attribute.
- ✓ The page is deployed to GitHub Pages.

### 💡 Notes

Follow these instructions to deploy your project to GitHub Pages:

Create a new repository on your GitHub account and clone it to your computer.

When you're ready to deploy, use the `git add`, `git commit`, and `git push` commands to save and push your code to your GitHub repository.

Navigate to your GitHub repository in the browser and then select the Settings tab on the right side of the page.

On the Settings page, select Pages on the left side of the page. On the GitHub Pages screen, choose `main` in the dropdown under Source.

Navigate to `<your-github-username.github.io/your-repository-name>` and you will find that your new webpage has gone live! For example, if your GitHub username is "lernantino" and the project is "css-demo-site", then your URL would be `<lernantino.github.io/css-demo-site>`.

### Additional Task

If you have completed this activity, work through the following challenge with your group to further your knowledge:

How can CSS be used to apply a different style to an `<a>` element when a cursor points at it?

Answer: refer the logo in my webpage -you can modify the link format as follows:

`a:link`

`a:visited`

`a:hover`     **note** `a:hover` MUST come after `a:link` and `a:visited` in the CSS definition in order to be effective.

`a:active`     **note** `a:active` MUST come after `a:hover` in the CSS definition in order to be effective.