Project Purpose

The project has a clear, well-defined purpose addressing the needs of a particular target audience (or multiple related audiences).

The project's purpose is evident to a new user without having to look in the documentation.

The project's documentation provides a clear rationale for the development of this project.

UX design

The project's documentation describes the UX design work undertaken for this project and the reasoning behind it.

Any wireframes, mockups, diagrams etc... you created as part of the design process are included in the project

Suitability for purpose

Dissatisfi

The site's design, as implemented, provides a good solution to the users' demands and expectations.

A regular user would not immediately think "there's a much better way to do this" about any part of the project.

Navigation

Dissatisfi

All resources on the site are easy to find, allowing users to navigate the layout of the site intuitively.

The site's navigation is consistent and reasoned.

There is never a need to use the Back button to move through the site.

For any external links, the target="_blank" attribute is used.

There are no broken links.

Ease of Use

Dissatisfie

It is easy and straightforward for a new user to figure out how to use your site without having to read any documentation

Have you had others (for example, family members, friends and/or other students) try out your site and they all said so?

The site is intuitive to use and never confuses the user or surprise them in a negative way.

The user has full control of their interaction with the project and at no point needs to "fight" it.

The site avoids aggressive automatic pop-ups and autoplay of audio; instead of allowing the user to initiate such actions.

All input elements are clearly labelled, and provide placeholders and default values whenever relevant.

The project follows common and consistent UI/UX conventions - there are plenty of online resources you can take inspiration from, such as GoodUI.

Information Architecture

Dissatisfied 😕

All information displayed on the site is presented in an organised fashion with each piece of information being easy to find and none feeling out of place.

Headers are used to convey structure - each section has a header that's easy to see and clear to understand.

The written language used on your sites is straightforward for the user to follow.

Whenever relevant, the site provides interactivity to make the information easier to consume.

Defensive Design

- A customer is not be able to break the site by clicking buttons out of the expected order or by providing any unexpected inputs.
- All forms intelligently handle empty or invalid input fields.
- Navigating between pages via the back/forward buttons can never break the site.
- This includes unexpected actions such as navigating back to the login page after already being logged in.
- User actions should not cause internal errors in the console
- Clear feedback to the user is given for any action disallowed by the developer.

Responsive Design

- All page elements look well on screens as small as 360 pixels wide and as big as 3840 pixels wide (4K).
- The site uses Bootstrap grid sizes or CSS3 media queries to ensure the layout changes appropriately and reflows when the screen is resized.

Image Presentation

- Graphics are consistent in style and colour.
- The background never distracts from the foreground information.
- All kinds of multimedia content used in the project work well on the different popular browsers.
- Whenever needed, multiple alternative file types are used.
- Images always maintain their original aspect ratio when the screen is resized (crop don't stretch).
- All images are of sufficient resolution to not appear pixelated.
- Image files are not bigger than is needed - full-screen images are under 3MB, while smaller images are <500kB.
- If any larger files are being loaded, there is a progress indicator.
- For large video/audio resources, prefer an external hosting platform (e.g., YouTube, S3....)

Color Scheme and Typography

- There is sufficient contrast between background and foreground colors.
- The color scheme used on the site consists of a palette of colors that work well together.
- The typefaces used complement one another.
- All text is legible; particular attention to legibility is maintained when text formatting effects are in use.
- Text is never obscured by images or colors.

Directory Structure and File Nam

Dissatisfied 🙁

Your project's files are named clearly and consistently, and located in appropriately named directories. ○

Whenever relevant, files are grouped in directories by file type (e.g., a static directory will contain all static files and may be organized into sub-directories such as CSS, images, etc.) ○

There is a clear separation between your files and any external files (for example, library files are all inside a directory named 'libraries'). ○

File names are descriptive and consistent. For cross-platform compatibility, file and directory names shouldn't have spaces in them and should be lower-case only. ○

Version Control

Your code is managed in git, with a separate well-named commit for each feature/fix.

You avoid very large commits, because this makes it harder to understand your development process and may lead the assessor to suspect plagiarism.

Testing implementation

Dissatisfied 🙁

You have conducted enough testing to convince the assessor that you legitimately believe that the site works well. ○

You conducted manual testing of your HTML/CSS for usability and responsiveness. ○

If using JavaScript or Python, you have created automated unit tests. ○

If using automated tests in JavaScript or Python are they effective and meaningful? Ensure no useless tests. ○

If you have used test driven development (TDD) for JavaScript or Python ensure this is demonstrated in your commit log. ○

Testing write-up

Dis

Your testing (both manual and automated) is well documented either in the README or a separate file.

Your write-up discusses any interesting bugs found and their fixes.

Your write-up mentions and explains any bugs that were left unfixed.

Readme file

Dissa

The project includes a readme file named README.md which is intended as an introduction for other developers who would like to use and contribute to this project. Note that submitting a project without a readme is an immediate fail.

The readme file describes the project's components, the technologies used and any other important details.

The readme is well-structured and easy to follow.

Your readme file is written in markdown and uses markdown formatting consistently and effectively.

The readme describes the project's purpose, components, technologies, and all other important details.

Comments

All code files include clear and useful comments, wherever they are relevant.	○
Consider your intent, the reasoning and any trade-offs behind your code.	○
Your comments explain the "why" rather than the "what".	○

Deployment implementation

Dissatisfied 🙁

Your site is deployed online on Github Pages, Heroku or any other system with full functionality.	○
The deployed version is identical to the development version except as explicitly mentioned in the deployment section in the documentation.	○

Deployment write-up

Dissa

The deployment procedure is fully documented in a section in the readme file.	
Any differences between the development code and deployed code are fully explained.	