UDACITY

‹ Return to Classroom

DISCUSS ON STUDENT HUB

# Landing Page

REVIEW

CODE REVIEW

HISTORY

## Meets Specifications

Dear Learner,
You have done a commendable job. I like the way you have created the logic switch between the active/inactive states.

Congratulations again and all the best for the rest of the Nanodegree.

#beInsideBeSafe

## Interface and Architecture

The project should have a structure like the one shown below. All files shown must be present and the app must successfully render a home page with clear design and functionality added when index.html is loaded in the browser. No errors should display in console.

```
css
- styles.css
index.html
js
- app.js
README.md
```

Good job with the directory structure. This is a good practice, so that in the future when the project starts to grow, we know what to find where? Also, as the number of pages or use cases increases, there can be sub-folders inside those folders like, if there are multiple pages, we can have a root style, a common style, and then a page-specific style and the same goes for JS.

**All features are usable across modern desktop, tablet, and phone browsers.**

The implementation of responsiveness of the website is aptly implemented. No matter what size I make the browser screen, the features click, scroll, highlighting of section etc are working in all the sizes.

This is especially important to increase reachability of your application. As we both know, there are many different types of screen sizes and resolutions, if our app doesn't render on someone's device, they won't be interested in using our product.

**Styling has been added for active states.**

Styling has been added for both menu items and the sections in the `style.css`. This is a good thing, as the stateful design is a part of good user interaction.

**There are at least 4 sections that have been added to the page.**

Great job for adding 4 sections.

## Landing Page Behavior

**Navigation is built dynamically as an unordered list.**

You have achieved this quite efficiently. I am actually impressed with the cleanliness and readability of the code.

**It should be clear which section is being viewed while scrolling through the page.**

The above-mentioned styles in `CSS` are being programmatically applied to the section in the view or the active section.

**When clicking an item from the navigation menu, the link should scroll to the appropriate section.**

You have implemented this part pretty nicely. What I also like is that you have added the navigate to top button and how you integrate its implemented inside the default function, which is a good practice, because

eventually it is a menu button.

## Documentation

**The ReadMe file should have non-default text in it that is specific to this project. It doesn't have to be thorough, but should have some basic information, and use correct markdown.**

It's certainly useful to summarize the project and note what skills it took to complete the project. This will only become more and more important as you build even more advanced projects for your portfolio.

There are some general conventions for what it should include and I have listed them here for you to refer to in the future:

1 General description of the project or content of the repository
2 List of what software, firmware and hardware you may require.
3 Installation instructions for the software and firmware.
4 List of files included in the project.
5 Copyright and licensing information.
6 Acknowledgements and credits for any resources or blogs that helped you create the project.
7 Known bugs

Here is also the GitHub help page on READMEs: https://help.github.com/articles/about-readmes/

**Comments are present and effectively explain longer code procedure when necessary.**

**Code is formatted with consistent, logical, and easy-to-read formatting as described in the Udacity JavaScript Style Guide.**

DOWNLOAD PROJECT

RETURN TO PATH

Rate this project

START