Rubric: Coding Foundations

Course: Coding Foundations

This Rubric criteria focuses on a student's ability to:

- **Demonstrate skill in building applications with HTML, CSS, and Flexbox.**
- **Build dynamic programs using HTML, CSS, and Flexbox.**
- **Show how to style applications with Scss.**
- **Learn Management and Teamwork Capabilities.**
- **Basic command-line functions.**
- **Developing your Soft Skills.**

Delivery

Demonstrate what you have learned about HTML and CSS by replicating the CodePen.io homepage using the screenshot below.

CodePen.io is a relatively simple webpage that combines a number of things you already know, such as working with images, creating buttons, and text fields.

Weight: 40%

Excellent

- Completed between 90-100% of the requirements.
- Delivered in the correct format via a GitHub URL space provided on devCamp.
- Notify a Teaching Assitant via the Support App that you have completed the project.
- Include the following elements in your project.
  - Navbar that includes "Pens, Projects, Posts, Collections, and Spark with a down chevron (does not need to actually dropdown).
  - Images - use available tools to recreate colors, etc.
  - Don't worry about replicating any functionality features such as drop-down, scrolling, etc. Simply replicate the landing page.

Good

- Completed between 80-90% of the requirements.
    - Delivered in the correct format via a GitHub URL space provided on devCamp.
    - Notify a Teaching Assitant via the Support App that you have completed the project.
    - Include the following elements in your project.
    - Navbar that includes "Pens, Projects, Posts, Collections, and Spark with a down chevron (does not need to actually dropdown).
    - Images - use available tools to recreate colors, etc.
    - Don't worry about replicating any functionality features such as drop-down, scrolling, etc. Simply replicate the landing page.

Satisfactory

- Completed between 70-80% of the requirements.
    - Completed between 90-100% of the requirements.
    - Delivered in the correct format via a GitHub URL space provided on devCamp.
    - Notify a Teaching Assitant via the Support App that you have completed the project.
    - Include the following elements in your project.
    - Navbar that includes "Pens, Projects, Posts, Collections, and Spark with a down chevron (does not need to actually dropdown).
    - Images - use available tools to recreate colors, etc.
    - Don't worry about replicating any functionality features such as drop-down, scrolling, etc. Simply replicate the landing page.

Unsatisfactory

- Completed less than 70% of the requirements.
- Not delivered in the correct format via a GitHub URL space provided on devCamp.
    - Not notifying a Teaching Assitant via the Support App that you have completed the project.
    - Not Including the following elements in your project and are in the wrong area of the project.
    - Not having a Navbar that includes "Pens, Projects, Posts, Collections, and Spark with a down chevron (does not need to actually dropdown).
    - Not using the provided images - use available tools to recreate colors, etc.

Runtime

This criteria component focuses on the student's ability to create websites with HTML and CSS and programs that function properly.

Weight: 30%

Excellent

- The web page has no errors.
- Code runs without errors.
- Includes all required elements.
- Accurate representation of the CodePen homepage.

Good

- The web page has no errors.
- Code runs without errors.
- Includes all required elements.
- Accurate representation of the CodePen homepage.

Satisfactory

- The web page has no errors.
- Code runs without errors.
- Includes all required elements.
- Accurate representation of the CodePen homepage.

Unsatisfactory

- The web page has errors.
- Coding runs with errors.
- Does not include all required elements.
- Does not show an accurate representation of the CodePen homepage.

Final

This criteria relates to the student's performance on the final exam.

Weight: 20%

Excellent

- A score of greater than or equal to 90%

Good

- A score ranging from 80-89%

Satisfactory

- A score ranging from 70-79%

Unsatisfactory

- A score of less than 70%

Documentation

This criteria component should analyze how easy it is to read a student's code. This can include the proper use of comments, self-documenting code, etc.

Weight: 10%

Excellent

- Clearly and effectively documented including descriptions of all variables.
- The student’s code is clean and not cluttered(proper spacing).
- Utilizes effective names for classes and methods for self-documenting code.

Good

- Clearly and effectively documented including descriptions of all variables.
- The student’s code is clean and not cluttered(proper spacing).

Satisfactory

- Basic documentation has been completed including descriptions of all variables.
- The purpose is noted for each function.

Unsatisfactory

- No documentation is included.