**Your Task**

As you proceed in your journey to becoming a full-stack web developer, it’s likely that you’ll be asked to complete a coding assessment, perhaps as part of an interview process. A typical coding assessment is a combination of multiple-choice questions and interactive coding challenges.

To help you become familiar with these tests and give you a chance to apply the skills from this module, this week’s Challenge invites you to build a timed coding quiz with multiple-choice questions. This app will run in the browser, and will feature dynamically updated HTML and CSS powered by JavaScript code that you write. It will have a clean, polished, and responsive user interface. This week’s coursework will teach you all the skills you need to succeed in this assignment.

**NOTE**

Coding assessments are an important part of the interview process for developers. In fact, employers often use them to filter out job candidates with a cut-off score. To help you prepare, we’ve developed a set of technical interview questions that you can answer throughout the course. The difficulty level of these questions will increase as you become a more proficient developer. If you take advantage of these opportunities to practice, you should be well-prepared to shine during the technical interview process toward the end of this course!

**User Story**

AS A coding boot camp student

I WANT to take a timed quiz on JavaScript fundamentals that stores high scores

SO THAT I can gauge my progress compared to my peers

**Acceptance Criteria**

GIVEN I am taking a code quiz

WHEN I click the start button

THEN a timer starts and I am presented with a question

WHEN I answer a question

THEN I am presented with another question

WHEN I answer a question incorrectly

THEN time is subtracted from the clock

WHEN all questions are answered or the timer reaches 0

THEN the game is over

WHEN the game is over

THEN I can save my initials and score

**Mock-Up**

The following animation demonstrates the application functionality:

**Grading Requirements**

This challenge is graded based on the following criteria:

**Technical Acceptance Criteria: 40%**

* Satisfies all of the above acceptance criteria.

**Deployment: 32%**

* Application deployed at live URL.
* Application loads with no errors.
* Application GitHub URL submitted.
* GitHub repository that contains application code.

**Application Quality: 15%**

* Application user experience is intuitive and easy to navigate.
* Application user interface style is clean and polished.
* Application resembles the mock-up functionality provided in the Challenge instructions.

**Repository Quality: 13%**

* Repository has a unique name.
* Repository follows best practices for file structure and naming conventions.
* Repository follows best practices for class/id naming conventions, indentation, high-quality comments, etc.
* Repository contains multiple descriptive commit messages.
* Repository contains a high-quality README file with description, screenshot, and link to deployed application.

**How to Submit the Challenge**

You are required to submit BOTH of the following for review:

* The URL of the functional, deployed application.
* The URL of the GitHub repository. Give the repository a unique name and include a README describing the project.