Testing

- We will implement unit testing of our code to ensure each component works as expected.
    - We will use Jest for testing out React components.
        - Jest is the most common testing framework for React projects.
        - Jest allows us to not only test the functional side of the React components but also allows us to test the usability of the website to simulate user interactions.
    - We will use Mocha for testing our NodeJS components.
        - Mocha is one of the most well know testing frameworks for NodeJS and is similar to Jest in its structure.
- We will have unit tests for each component we write, and updating of components will require these unit tests to be updated as well.
- Integration testing will be used to ensure that the React components interact with the service side of the project in order to ensure the frontend works with the backend.
- We will present the system to the client and allow them to test the entirety of the system to ensure everything works as expected.
    - It will be presented to the client at the end of the sprints the client defines to ensure that we met their expectations after completing a sprint.

Integration

- Our team will utilize Github Actions in order to ensure our code does not break when new features are merged into the main branch.
    - In Github Actions, we will implement the tests for Linux, Windows, and macOS.
- As part of the pull-request process, all tests in the project are required to pass. If a single test does not pass then merging the feature branch into main will be blocked until the failing test is investigated and fixed.