Congratulations! You’ve built tests for a feature.

You will notice a couple changes to your files and the addition of a web browser. First, let’s take a look at your **index.html** file. You will see we added some tags to the file and added an <h1> element with a title for our webpage.

We also linked to a **style.css** file that will provide some basic styling to the code you have written, so that we can see what our messaging app might look like in a live browser! To view your app, run npm start in the terminal component and then refresh the browser on the right.

Take a look at the **style.css** file in the editor. Feel free to add and change styles in the **style.css** file to customize your web application’s UI.

After making any changes to your **style.css** file, run npm start in your terminal, and refresh your browser to see the changes.

In this lesson, we have been writing tests at the top level of the TDD Testing Pyramid, which concerns the part of the app that users interact with. To make our feature test pass and get back into the green we need to drop down to the server level — which concerns the part of the application that makes ‘POST’ requests to the server.

**In this lesson you learned:**

* When developing a new feature and practicing outside-in development, feature tests are where we’ll typically start.
* Feature tests often incorporate every layer of the application and — using WebDriverI/O and Mocha — exercise features in the same way that a human user would. They’re a good tool for reproducing end-user behavior.
* WebDriverI/O is a Node package that interacts with a “headless” instance of PhantomJS.
* Feedback from feature tests is usually in terms of HTML (i.e. that text or button that you said would be on the page isn’t on the page).
* Because feature tests typically hit every layer of a developer’s stack, they are slower than tests at lower layers, and errors thrown in feature tests can be difficult to interpret and provide little guidance on what the developer can do to resolve them.
* Their value, however, is in developer confidence that the software functions as expected.