

The smart criteria we are going to automatically encode into our pipeline will be:

Application reliably stores a maximum of 10 liked recipes that can be accessed at any point by the user.

We will create an integration test that communicates between our repository controller to pull a testing user account, likes/saves 10 recipes to it, and then verifies that the backend successfully returns those 10 accounts to the controller. The test will then attempt to save an 11<sup>th</sup> recipe and will then verify that we have some logic implemented to prevent the request from being processed and returns proper error handling. Tony installed the essentials for Jest on our repository and tested that we can setup automatic tests through Github actions. We chose this test as it is relatively straightforward to implement and does not require much more to be developed before we can setup the testing script.

We will not have an actual unit/integration test implemented for this deliverable as our project isn't at the point to where it will be useful for verifying the quality of our code. Most of our working project lies in the backend database interactions, so implementing unit tests for database interactions is relatively redundant. We should have the necessary framework implemented to have an integration test setup by next week's lab section though.

We will be using Jest for our code coverage assessments also as it has the capability to track code coverage along with running our testing scripts.