CS 320

Module 4 Journal

Caleb Ewer

6/01/2024

* To what extent was your testing approach aligned to the software requirements? Support your claims with specific evidence.

In my testing approach I used the requirements almost as a guideline of how the testing should be done. In the first test I tested to verify that all fields must not be null. I then went through each data field and did individual testing on each one. Each data field and the setters and getters were tested on the functionality.

* Defend the overall quality of your JUnit tests for the contact service and task service. In other words, how do you know that your JUnit tests were effective on the basis of coverage percentage?

For my testing I verified that each data field met the requirements based on the tests. As an example, the first name, last name, phone number, and address had length requirements and should be updateable, so I verified those requirements in that testing and made sure each one was able to pass the test. In contrast to this the ID should not be updateable so I verified that on by just calling the get method and checked it based on the testing data.

* How did you ensure that your code was technically sound? Cite specific lines of code from your tests to illustrate.

In my coding I took into account that we do not want a repeat ID so I made it so that the ID was set once when the contact was made and we were unable to change the ID field. A black screen with white text

Description automatically generated

As we see here the ID is set in the constructor based on it being 10 characters or less and not null.

A black background with white text

Description automatically generated

Here there is a getter to get the ID but there was not created a setter to update or modify it.

This was tested like this.

A computer screen with white text

Description automatically generated

* How did you ensure that your code was efficient? Cite specific lines of code from your tests to illustrate.

One of the ways that I made sure that the code was efficient was not trying to make duplicate tests to complete the same thing. I accomplished this by only making one contact in the beginning and adjusted that one contact. This is shown here: A screenshot of a computer program

Description automatically generated