Testing – Software Engineering Methods – Coursework 2 – 40168316

Any other features you think might enhance the system?

* When I do my string comparisons for tasks such as identifying universities and subjects, I ignore the case sensitivity by adding in “StringComparison.OrdinalIgnoreCase”. This means that users don’t need to worry about case sensitivity.
* Different custom output messages are shown at the end of the program identifying and displaying if Universities, Course type and Subjects were found. If in a certain area nothing was found, for example no University, then the custom output message will notify the user that no University was found in the entered message.
* The Json file is outputted to a file in case the system wants to use these in the future or Deserialize the content.

Briefly describe your overall testing strategy for the system. What form of testing will you do, how will you identify test cases?

I would carry out basic White and Black box testing. White box testing is where you enter an input, see how it changes in the process and get an output. This output should be expected. Black box testing is similar however you do not get to the see the process of how it changes.

By writing out the predicted and correct processes and outcomes, you can then do either white box testing to see if the variable/content changes correctly, to what you expect it to be and compare that process and outcome to the expected. You can also do black box testing where you compare the input with the output and see if they match. With white box testing you can also use breakpoints which is another useful tool, allowing you do break down the process much more and identify where the variable/content changes.

These two testing methods will allow you to identify and create test cases and test the system thoroughly.

Provide a test plan in the form of a table of required tests. You are not required to list detailed test data but should list the success or otherwise of the test?

|  |  |  |
| --- | --- | --- |
| **Test** | **Predicated Outcome** | **Actually Outcome** |
| Identify multiple Universities | Multiple Universities in message box | Multiple Universities in message box |
| Identify multiple Subjects | Multiple Subjects in message box | Multiple Subjects in message box |
| Show Multiple Universities in Json in a message box | Multiple Universities in Json  in a message box | Multiple Universities in Json  in a message box |
| Show Multiple Subjects in Json in a message box | Multiple Subjects in Json in a message box | Multiple Subjects in Json in a message box |
| Application can handle upper and lowercase messages being entered | Yes, the application can handle upper and lower case messages along with a mixture | The application can handle upper and lower case messages along with a mixture |
| Application can handle numbers being entered | Yes, it will not crash | The application continues as if it a normal message and displays a message saying no university, course type or subjects were found |
| Application will produce an error if no files are found – the files being tested are subjectslist, universitieslist and textwordslist | Yes, it will not crash and it produce and error message | An error message was produced saying file not found |
| Application remove invalid message to correct file | Yes, the application will remove invalid messages if words from the textwords file are found in the message. The message will then be added to the Quarantine file. | Application produces an error message when invalid message is found, writes it to the quarantine file |