­­­G51FSE Assessed Lab 4

|  |  |  |
| --- | --- | --- |
| **The Assemblers with Flip-Flops** |  | 09/03/2018 |

# 3. Test Plans

This section of the software specification document highlights the details of the system through the use of Unified Modelling Language and prototypes.

We have added some additional columns to the bug table. These are notes and a description for the changes are given in this paragrah.We have decided to have additional columns in the test table so we can provide more detail about the tests carried out. This will increase the traceability of the tests and make the changes easier to see. Firstly, we added **a Test ID** so that we can have a unique id for each test carried out. This will make it easier to refer back to the tests later on in the document. In addition to this, we added a column for prerequisite which will store detail about what other functions need to be performed before this test is carried out. Next, we included a column for the **actual output**. This is necessary so that we can compare if the expected output matches the actual output. If it doesn’t, a reason must be provided as to why and most likely there is an error/flow in the program. In addition to this, we added a column called ‘**test created by’**. This will hold the name of the person who created the tests and carried it out. This will increase accountability. Next, we added a column called ‘**pass or fail’**. This will make it easier to see the number of tests that has passed without the person having to read the details of each individual test. We then created a column which will store the **date** that the test has been carried out on. This will make it easier to see which version of a test is the latest if a test had been carried out multiple times. Finally, we added a column for **notes** which can be used to store additional details about the tests that does not fall into any other category. For example, if a test has failed, the notes could include a description as to why the test has failed.

Email Class

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test ID | Function Name | Test Aim (From perspective of method/how method handles this) | Pre-Requisites | Inputs (+ examples) | Expected Output(s) | Actual Output | Test Created By | Source: Spec or code inspection | Pass / Fail | Date (Completed by) | Notes (Questions/Assumptions) |
| 101 | Default Constructor | To see how the method will react when no variables are passed to the constructor. | None | None | Initialise an email ready for populating. | - | Athullya Roy | Specification Document – Class Document | - | - |  |
| 102 | Main Constructor | To see how the method will react when receiving all four string parameters in correct form. | None | Sender: [psyra7@hotmail.co.uk](mailto:psyra7@hotmail.co.uk)  Receivers Email: [psyrrr1@hotmail.co.uk](mailto:psyrrr1@hotmail.co.uk)  Subject:  “New Project – New Project Concept”  Email Body:  “Hi,  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec ullamcorper tellus id nisi mattis, consequat vulputate lorem eleifend. Sed sollicitudin odio eu elit porta, a dictum urna tempus.  Regards,  Athullya” | Initialise an email ready to be sent, with a sender’s email, receiver’s email, subject and body. | - | Ram Raja | Specification Document – Class Document | - | - |  |
| 103 | Main Constructor | How the method will react when a Receiver’s email is passed in an incorrect format. | None | psyar8.co.uk |  | - | Athullya Roy | Specification Document – Class Document |  |  |  |
| 104 | Main Constructor | How the method will react when no Subject Line is passed. | None | Subject line is null. | Returns all populated strings with values, subject remains null. |  | Ram Raja | Specification Document – Class Document |  |  |  |
| 105 | Main Constructor | How the method will react when no Receiver’s email is passed. i.e. it is null. | None | Receiver’s email is null. | Returns all populated strings with values, but fails as no receiver’s email is set. |  | Athullya Roy | Specification Document – Class Document |  |  |  |
| 106 | Getter for Sender’s email address | How the method will respond when the sender’s email is taken correctly. | None |  | Returns the sender’s email as a string. |  | Ram Raja |  |  |  |  |
|  | Getter for Sender’s email address | How the method will respond when the sender’s email is not set. | None |  | Returns all populated strings with values, but fails as no receiver’s email is set. |  | Athullya Roy |  |  |  |  |
| 10 | Getter for Receiver’s Email Address | How the method will respond when the receiver’s email is taken correctly. | None |  | Returns the receiver’s email as a string. |  | Ram Raja |  |  |  |  |
|  | Getter for Receiver’s Email Address | How the method will respond when the receiver’s email is not set. | None |  |  |  |  |  |  |  |  |
|  | Getter for Subject Line | How the method will respond when there is a subject line. | None |  |  |  |  |  |  |  |  |
|  | Getter for Subject Line | How the method will respond when there is no subject line. | None |  |  |  |  |  |  |  |  |
|  | Getter for Email Body | How the method will respond when body text has been given. | None |  |  |  |  |  |  |  |  |
|  | Getter for Email Body | How the method will respond when there is no body text. | None |  |  |  |  |  |  |  |  |
|  | Setter for Sender’s Email |  |  |  |  |  |  |  |  |  |  |
|  | Setter for Receiver’s Email |  |  |  |  |  |  |  |  |  |  |
|  | Setter for Subject Line |  |  |  |  |  |  |  |  |  |  |
|  | Setter for Email Body |  |  |  |  |  |  |  |  |  |  |
|  | Validity Checking Function | How this function will respond when all parameters have been populated with valid values. i.e. not null. | All parameters have been filled in, i.e. Sender’s Email, Receiver’s Email, Subject and Body.  Both emails are in the correct format. | Sender: [psyra7@hotmail.co.uk](mailto:psyra7@hotmail.co.uk)  Receivers Email: [psyrrr1@hotmail.co.uk](mailto:psyrrr1@hotmail.co.uk)  Subject:  “New Project – New Project Concept”  Email Body:  “Hi,  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec ullamcorper tellus id nisi mattis, consequat vulputate lorem eleifend. Sed sollicitudin odio eu elit porta, a dictum urna tempus.  Regards,  Athullya” | Function returns true – meaning no errors should be found and the email is valid and ready to be sent. |  | Ram Raja |  |  |  | The subject line and body may be kept empty. |
|  | Validity Checking Function | How this function will respond when all parameters are null except for receiver’s email. | None | Sender: NULL  Receiver: psyar8@hotmail.com  Subject: NULL  Body: NULL | Function returns false – meaning the email cannot be sent as no emails are specified. |  | Athullya Roy |  |  |  |  |
|  | toString() override |  |  |  |  |  |  |  |  |  |  |

Change Log

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Change ID | GIT Commit ID | Bug ID | Description of Change | Files Changed | Date Received | Date Changed | Initiator By (Who Changed) | Change Checked By | Notes |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

Bug Fix List

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem ID | Problem Description | Line of code | Test ID (if created) | Change ID | Proposed Fix | Priority  High, Med, Low | Date Problem identified | Fixed? (Yes/No) | Date Problem fixed | Who identified/fixed the test? | Does the bugged code relate to other functions? | Notes |
| 3 | ListContacts method does not return email address in CompanyEmailSystem Class | CompanyEmailSystem, 151 |  |  |  | Med | 09/03 |  |  |  |  |  |