­­­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.

Company Email System (Main Class)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test ID | Function Name | Test Aim | Prerequisite | Inputs | Expected Output(s) | Actual Output | Test Created By | Source: Spec or code inspection | Pass / Fail | Date (Completed by) | Notes |
| 301 | Main Method | Testing the user sees the correct menus when not in a project. | Set Current Project Variable to 0 | N/A | A list of user options:   * See a list of projects * Add a new project * Viewing a project * Exit the system | A list of user options:   * See a list of projects * Add a new project * Viewing a project * Exit the system | Aidan Reed | Class Description | PASS | 01.05.2018  Aidan Reed |  |
| 302 | Main Method | Ensures the system exists at the main menu when user types X | Launch the main function | N/A | System should output Good Bye! | Good Bye! | Aidan Reed | Class Description | PASS | 01.05.2018 |  |
| 303 | Main Method | Tests the ArrayList can grow to large size | Create 10,000 Projects after initializing project array | N/A | When checking size of array list should be 10000 | Error: Complication  Error | Aidan Reed | Class Description | FAIL | 01.05.2018 | Error: Complication Error method is not public and Projects array is not public |
| 304 | List Project | Checks that the function returns the correct number of projects created | Initialize the Projects array and add 5 emails with constant **KPTITLE1 + I** the index of for loop | **KPTITLE1 + I** the index of for loop to create the projects | The list of projects in the format:   1. **kPTITLE1 + I** [Design] – 0 emails] | Error: Complication Error: No Line found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 305 | List Project | Checks that the function can handle large email counts. | Create 10,000 emails for initial phase for a project with **KSENDER kRECIPIENT KSUBJECT KBODY1** used for the inputs | N/A | A string with the count in the format:  New Project [Design] – 10 Emails  The execution time should be below 5 seconds. | Error: Complication  Error: No Line found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 306 | Add Project | Ensures the project title added as input is correctly the system provided feedback [Project added] | Initialize the Projects array and create a project with title **KPTITLE1** | Project Title **KPTITLE1** | System output [Project added] | Error: Complication  Error: No Line found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 307 | Add Project | Ensures the project is created and title is set correctly | Initialize the Projects array and create a project with title **KPTITLE1** | Project Title **KPTITLE1** | When calling get Project title **KPTITLE1** should be displayed | Error: Complication  Error: No Line found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 308 | Add Project | Checks that the project is created if the user does not enter a name | Initialize the Projects array and create a project with no title | N/A | When calling get Project title New project should be displayed | Error: Complication  Error: No Line Found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 309 | List Emails | Tests that the function returns the correct phase ID for project in the initial stage | Initialize the Projects array and create a project add email with **KSENDER kRECIPIENT KSUBJECT KBODY1** used for the inputs | N/A | Returns the sender and subject in format :  From: Subject | Error: Complication  Error: No Line Found | Aidan Reed  Aidan Reed | Class Description | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| 310 | List Emails | Tests the function with an invalid phase ID | Initialize the Projects array and create a project | Phase ID 20000 | Should not return any emails With an error message: Error: Unknown Phase | Error: Complication  Error: No Line Found | Aidan Reed | Class Description & Code Inspection | FAIL  FAIL | 01.05.2018  02.05.2018 | Error: Complication Error method is not public  Failure: no line found CompanyEmailSystem - 112 |
| ---310 | List Emails | Test the function with no phase ID | Create Project Orion and set Phase ID to 2 | N/A | Should return emails for the current project at phase ID 2 |  | Aidan Reed | Specification – Class Description |  |  |  |
| 311 | List Phases | Call the function from the user interface | Create Project Orion and Select Project and set phase ID.  Add 100 emails phase 1  Add 200 emails phase 2  Add 20 emails phase 3 | N/A gets current project from global variable | Returns the phases for the current project with email counts:  Add 100 emails phase 1  Add 200 emails phase 2  Add 20 emails phase 3 |  | Aidan Reed | Specification – Class Description |  |  |  |
| 312 | List Contacts | The function is called from user interface in the user context | Create Project Alpha and add 20 users | Select the project and list contacts button | Returns the list of 20 users set in the prerequisite stage |  | Aidan Reed | Specification – Class Description |  |  |  |
| 313 | List Contacts | Call function when not in project | Current project value set to 0 | N/A | The program should not crash and thrown an exception |  | Aidan Reed | Specification – Class Description |  |  | It’s important to check that the class implantation avoids error prone constructs and does not crash if the project variable is not set correctly. |
| 314 | Change Project Phase | Ensures the next phase function is called correctly | Create Project Beta and set project phase 1 | Select Next phase button | Calls the next phase function and prints out a message to say it was successful |  | Aidan Reed | Specification – Class Description |  |  |  |
| 315 | Change Project Phase | Tests to check the last phase message is displayed | Create Project November and set project phase to last phase | Select Next phase button | Calls the next phase function and prints out message informing user already in last phase |  | Aidan Reed | Specification – Class |  |  |  |
| 316 | List Phase | Test to check how the system responds to user request of viewing Phase Folders in a new project. | Test 311 completed.  Create a new project. | Select List Phase Folders | Display a list of empty folders / Display an empty list |  | Ram Raja |  |  |  |  |
| 317 | Main Method | Test the user cannot access a nonexistent project. | None | Enter a non-existent project number. | Inability to access the project. |  | Ram Raja |  |  |  |  |
| 318 | Main Method | Return first project | First project must exist. | Select first project. | Display first project. |  | Ram Raja | Code Inspection |  |  |  |
| 319 | Main Method | Ensure the user sees the correct menus when in a project. | A project must be selected. | N/A | Display project menu, consisting of the following options:  - List Emails  - Add Email  - List Phase Folders  - Move To Next Phase  - List Emails in Phase  - List Contacts  - Exit Project |  | Ram Raja | Code Inspection |  |  |  |
| 320 |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Change ID | GIT Commit ID | Bug ID | Description of Change | Files Changed | Date Received | Date Changed | Initiator By (Who Changed) | Change Checked By | Notes |
| CHGE001 | B2283df | BUG001 | All methods inside the main class are set to private aswell as data member variables. To be able to perform email unit testing on each method the methods need to be made available without running the main class.   * AllProjects – made public * currentProjShowing – made public * listProjects – made public * addProject – made public * listEmails – made public * listPhases – made public * listContacts – made public * addEmail – made public * ChangeProjectPhase – made public | CompanyEmail (emailMessage) | 01/05/2018 | 02/05/2018 | Aidan Reed | Ram Raja | Junit tests that were created and failing as a result of this should now start to work – Ram Raja. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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? | Who Fixed it | Does the bugged code relate to other functions? | Notes |
| BUG001 | Error when adding new project to the system. Receives No Line found. There is a method call nextLine on the scanner object passed into the Add Project method which is removing the input and infact needs to be entered twice | 110-111 | 304-310 |  | Remove the nextLine method call on line 110 | Med | 01/05/2018 |  |  | Aidan Reed |  | No – other functions call this method |  |