­­­G51FSE Assessed Lab 4

|  |  |  |
| --- | --- | --- |
| **The Assemblers with Flip-Flops** |  | **04/05/2018** |

# Unit Testing

The following documentation contains details of the tests that we have created to test the different classes. We held our first group meeting on Tuesday 24th April 2018, where we discussed the various approaches we could take to carry out the project. Among the many ideas that were discussed, we decided to use GitHub to commit our projects online. On GitHub we created 6 branches so that everyone could save changes to their own local code repository. Once our tests were build, we merged it back to the master branch. In the duration of writing the tests and developing the documentation for the tests, all the team members kept pulling and committing their work to Git. This made it easier to work as a group as this improved traceability and visibility of the project.

As a group we also decided to use paired programming while developing the JUnit tests. Each pair was assigned a different class. Therefore, we met as pairs and developed the tests so that one person is at the machine typing while the other person work on the code. The pairs kept changing roles so that there is an equal amount of input by both individuals. On Friday 27th April 2018, we met again as a team. At this stage all the pairs have completed nearly half of the tests they were assigned. In this meeting we ensured that all the pairs were documenting their work and that they were following the correct layout and testing format that was discussed during the initial meeting.

All the documents contain a list of all the constants which we created for the tests. We created this to make the program more efficient and to have better readability. In addition to this, the document contains 3 different tables- test table, change log, and bug fix list. Test table contains details of all the tests that we created. Change log contains the details of all the changes we made to the program. Bug fix list contains a list of all the bugs we found in the code and contains details as to how we fixed it. While creating the tests and developing the program, all the members of the team kept updating the document so that all the information provided is accurate. All the tables contain unique ID’s so that the tests, changes and bugs can be easily identified. This also makes it easier to cross-reference the values in other tables. Alongside this, in the change table we have included a Git commit ID so that we can easily trace back to the change that was made.

All tests were performed on both Windows and Unix-based machines throughout the test phase, however they were verified at the final stage on a Windows-based system. Line breaking conventions are platform-dependent; therefore, it is advised that tests are performed on machines with both OS architectures. We have attempted to address this issue by identifying the user’s platform, accessing their respective line separator using a built-in method (System.lineSeparator();), assigning this to a constant and applying this wherever a line break is required. However, despite of this, we advise further tests are performed on both platforms.

More details on the tables and the formatting of the files can be found [here](https://github.com/psyar8/FSE_CourseWork_4).

(<https://github.com/psyar8/FSE_CourseWork_4>)

**Contents**

|  |  |  |
| --- | --- | --- |
| Class | Section | Page Number |
| CompanyEmail | Notes | 2 |
| Unit Tests | 3 |
| Change Log Table | 10 |
| Bug List | 11 |
| CompanyProject | Notes | 12 |
| Unit Tests | 13 |
| Change Table | 27 |
| Bug List | 28 |
| CompanyEmailSystem | Notes | 30 |
| Unit and Integration Tests | 31 |
| Change Log Table | 42 |
| Bug List | 45 |

# Email Class Unit Testing

The following tests use references to constants defined in the Junit test files at the start of the document. As such for easier comparison in the place the constants are defined below and are used in the Input, Expected Output and Actual Output columns to reduce repeating data and if the test inputs change in the future the document can be updated at a single point.

**kSENDER** = [joe.bloggs@gmail.com](mailto:joe.bloggs@gmail.com)  
**kBADEMAIL1 =** joe.bloggs@   
**kBADEMAIL2 =** [joe.@bloggs](mailto:joe.@bloggs)  
**kBADEMAIL3 =** [joe.bloggs@gmail@.com](mailto:joe.bloggs@gmail@.com)  
**kBADEMAIL4 =** @joe.bloggs@gmail.com   
**kRECIPIENT** = [max.power@live.com](mailto:max.power@live.com)  
**kSUBJECT** = “RE: Lorem Ipsum”  
**kBODY1** = “Lorem ipsum dolor sit amet, consecteutur adipiscing elit.”  
**kBODY2** = “This is a test email for unit testing”   
**kBODY3** = “”   
**kBODY4** = “Test Email”  
**nullEmail** (CompanyEmail Object)  
**populatedEmail** (CompanyEmail Object)  
**noRecipientEmail** (CompanyEmail Object)  
**noEmailBody** (CompanyEmail Object)

# Email Class Unit Testing

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Function Name** | **Test Aim (From perspective of method/how method handles this)** | **Source: Spec or code inspection** | **Test Created By** | **Pre-Requisites** | **Inputs (+ examples)** | **Expected Output(s)** | **Actual Output** | **Pass / Fail** | **Date** | **Tested By** | **Notes (Questions/Assumptions)** |
| **101** | Default Constructor | To see how the method will react when no variables are passed to the constructor. | Class Document | Ram Raja | N/A | N/A | Initialises an email ready for populating. | Stack overflow error | FAIL | 26/04/2018 | Ram Raja | Error as “emailMessage()” method returns itself rather than the variable “emailMessage”. |
| Initialised object | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **102** | Main Constructor | To see how the method will react when receiving all four string parameters in correct form. | Class Document | Ram Raja | N/A | **Sender’s Email:**  **kSENDER**  **Recipient’s Email:**  **kRECIPIENT**  **Subject:**  **kSUBJECT**  **Email Body: kBOD1** | Initialises an email ready to be sent, with a sender’s email, recipient’s email, subject and body | Stack overflow error | FAIL | 26/04/2018 | Ram Raja | Error as “emailMessage()” method returns itself rather than the variable “emailMessage”. |
| Initialised object with correct values that were passed | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **103** | Main Constructor | To test interactivity between both the Default and Main Constructor when all parameters are not populated. In this case all but the recipient’s email has been given. | Code Inspection | Ram Raja | N/A | Recipient’s email is null. | Initialises an email with all but the recipient’s email given, ready to be populated. | Stack overflow error as | FAIL | 26/04/2018 | Ram Raja | Error as “emailMessage()” method returns itself rather than the variable “emailMessage”. |
| Initialised object without email | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **104** | Main Constructor | To test interactivity between both the Default and Main Constructor when only one parameter is given, in this case the sender’s email. | Code Inspection | Athullya Roy | N/A | **Sender’s Email:**  **kSENDER** | Initialises an email with the sender’s email; with recipient’s email, subject and body as null, ready to be populated. | Stack overflow error | FAIL | 26/04/2018 | Ram Raja | Error as “emailMessage()” method returns itself rather than the variable “emailMessage”. |
| Initialised object with just senders email set | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **105** | Getter for Sender’s email address | To ensure the getter outputs the sender’s email when specified. | Class Document | Ram Raja | N/A | **Sender’s Email:**  **kSENDER** | Returns the sender’s email as a string. | The given sender’s email is returned. | PASS | 26/04/2018 | Ram Raja |  |
| **106** | Getter for Sender’s email address | To test how the method will respond when the sender’s email is not set. | Class Document | Athullya Roy | N/A | N/A | Returns sender’s email as null. | The sender’s email is returned as null. | PASS | 26/04/2018 | Ram Raja |  |
| **107** | Getter for Recipient’s Email Address | To ensure the getter outputs the recipient’s email when specified. | Class Document | Ram Raja | N/A | **Recipient’s Email:** **kRECIPIENT** | Returns the recipient’s email as a string. | The given recipient’s email is returned. | PASS | 26/04/2018 | Ram Raja |  |
| **108** | Getter for Recipient’s Email Address | To test how the method will respond when the recipient’s email is not set. | Class Document | Athullya Roy & Ram Raja | N/A | N/A | Return’s recipient’s email as null. | The recipient’s email is returned as null. | PASS | 26/04/2018 | Ram Raja |  |
| **109** | Getter for Subject Line | How the method will respond when there is a subject line. | Class Document | Athullya Roy & Ram Raja | N/A | **Subject:**  **kSUBJECT** | **kSUBJECT** | **kSUBJECT** | PASS | 26/04/18 | Athullya Roy |  |
| **110** | Getter for subject Line | How the method will respond when there is no subject line | Class Document | Athullya Roy & Ram | N/A | null | null | null | PASS | 26/04/18 | Athullya Roy |  |
| **111** | Getter for Email body | How the method will respond when body text has been given. | Class Document | Athullya Roy & Ram | N/A | **Message:**  **kBODY1** | **kBODY1** | Stack over flow | FAIL | 26/04/18 | Athullya Roy | StackOverFlow error |
| **kBODY1** | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **112** | Getter for Email body | How the method will respond when there is no body text | Class Document | Athullya Roy & Ram | N/A | null | null | Stack overflow error | FAIL | 27/04/18 | Athullya Roy | StackOverFlow error |
| Null | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **113** | Setter for Senders Email | To see if it could take a string as a parameter, ensuring that it follows the email format | Class Document | Athullya Roy & Ram | N/A | **Sender:**  **kSENDER** | null | null | PASS | 27/04/18 | Athullya Roy |  |
| **114** | Setter for Senders Email | To check if class variable is set to null and it fails if invalid email address is entered. | Class Document | Athullya Roy & Ram | N/A | Joe.bloggs | Function returns false | Null | FAIL | 27/04/18 | Athullya Roy | Fails because it doesn’t return the correct type. Also assumes fails means returning error message |
| Function returns false | PASS | 02/05/18 | Athullya Roy | Passes because the function setFrom has been changed. See change CHGE104 |
| **115** | Setter for Receivers Email | To see if it could take a string as a parameter, ensuring that it follows the email format | Class Document | Athullya & Ram | N/A | Recipient:  kRECIPIENT | Null | Null | PASS | 27/04/18 | Athullya Roy |  |
| **116** | Setter for Receivers Email | To check if class variable is set to null and it fails if invalid email address is entered. | Class Document | Athullya & Ram | N/A | “Max.power” | Warning Message – Return false | Null | FAIL | 27/04/18 | Athullya Roy | Fails because it doesn’t return the correct type. Also assumes fails means returning error boolean value false. |
| Function return False | PASS | 02/05/18 | Athullya Roy | Passes because the function setTo has been changed. See change CHGE104 |
| **117** | Set method for Subject Line | To check if it takes a String as a parameter. | Class Document | Athullya & Ram | N/A | **Subject:**  **kSUBJECT** | Null | Null | PASS | 27/04/18 | Athullya Roy |  |
| **118** | Set method for Subject Line | To check if warning produced if null is passed. | Class Document | Athullya Roy & Ram | N/A | null | Warning Message – return false | null | FAIL | 27/04/18 | Athullya Roy | Fails because system doesn’t respond to null being passed to function. |
| Warning Message – return false | PASS | 27/04/18 | Athullya Roy | Fixed after changes made inn BUG103 |
| **119** | Setter for Email Message Body  **(setMessage)** | Test the method correctly sets the email message and then compare that to what is retrieved from getMessage method | Class Document | Aidan Reed & Ram Raja | Uses **noEmailBody** Object with Message set using setMessage method and **kBODY1** Constant | **Message:**  **kBODY1** | **kBODY1**  After calling the getMessage method once the message has been set | Stack overflow error | FAIL | 26.04.2018 | Aidan Reed | Unable to determine if message was set correctly as when trying to retrieve the message using emailMessage method received a stack overflow error. |
| **kBODY1** | PASS | 27/04/2018 | Ram Raja | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **120** | Setter for Email Message Body  **(setMessage)** | Test the method correctly sets the email message to an empty string “” and then compare that to what is retrieved from getMessage method | Class Document | Ram Raja & Aidan Reed | Uses **noEmailBody** Object with Message set using setMessage method and **kBODY3** Constant | **Message:**  **kBODY3** | **kBODY3**  After calling the getMessage method once the message has been set | Stack overflow error | FAIL | 26.04.2018 | Aidan Reed | Unable to determine if message was set correctly as when trying to retrieve the message using emailMessage method received a stack overflow error. |
| **kBODY3** i.e. an empty string | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE101 BUG102 |
| **121** | Check Validity of Email Function  **(isValid)** | Tests the email object with no attributes set in the constructor meaning they are null | Class Document | Aidan Reed & Ram Raja | Uses **nullEmail** Object | N/A | False | False | PASS | 26.04.2018 | Ram Raja | The worst case where no values are set |
| **122** | Check Validity of Email Function  **(isValid)** | Tests the email object with All attributes set in the constructor | Class Document | Ram Raja & Aidan Reed | Uses **populatedEmail** object created with **kSENDER**, **kRECIPIENT**, **kSUBJECT**, and **kBODY1** constants | **Sender**:  **kSENDER**  **Recipient**:  **kRECIPIENT**  **Subject:**  **kSUBJECT**  **Message**:  **kBODY1** | True | True | PASS | 26/04/2018 | Aidan Reed | The best case where all values are set |
| **123** | Check Validity of Email Function  **(isValid)** | Tests the email object with one attribute set – the from address | Class Document | Aidan Reed & Ram Raja | Uses **nullEmail** object and setFrom method passing **kSENDER** constant | **Sender:**  **kSENDER** | False | False | PASS | 26.04.2018 | Ram Raja | The following 4 tests of similar nature test the individual cases for each attribute to ensure all attributes are checked in the isValid Function |
| **124** | Check Validity of Email Function  **(isValid)** | Tests the email object with one attribute set – the To address | Class Document | Ram Raja & Aidan Reed | Uses **nullEmail** object and setTo method passing **kRECIPIENT** constant | **Recipient**  **kRECIPIENT** | False | False | PASS | 26/04/2018 | Aidan Reed |  |
| **125** | Check Validity of Email Function  **(isValid)** | Tests the email object with one attribute set – the Subject | Class Document | Aidan Reed & Ram Raja | Uses **nullEmail** object and setSubject method passing **kSUBJECT** constant | **Subject:**  **kSUBJECT** | False | False | PASS | 26.04.2018 | Ram Raja |  |
| **126** | Check Validity of Email Function  **(isValid)** | Tests the email object with one attribute set – the message body | Class Document | Ram Raja & Aidan Reed | Uses **nullEmail** object and setMessage method passing **kBODY1** constant | **Message**:  **kBODY1** | False | False | PASS | 26/04/2018 | Aidan Reed |  |
| **127** | toString() override  **(toString)** | Checks the toString override method returns a value of type string | Class Document | Aidan Reed & Ram Raja | Uses **populatedEmail** object created with **kSENDER**, **kRECIPIENT**, **kSUBJECT**, and **kBODY1** constants | N/A | (String) | (String) | PASS | 26.04.2018 | Ram Raja |  |
| **128** | toString() override  **(toString)** | Checks the correct subject is returned when set during the test | Class Document | Ram Raja & Aidan Reed | Uses **nullEmail** object and setSubject method passing **kSUBJECT** constant | **Subject:**  **kSUBJECT** | **kSUBJECT** | **kSUBJECT** | PASS | 26/04/2018 | Aidan Reed |  |
| **129** | toString() override  **(toString)** | Checks the correct subject is returned when set during the test to an empty string | Class Document | Aidan Reed & Ram Raja | Uses **nullEmail** object and setSubject method passing “” empty string | Subject:  “” | “[no subject]” | “no subject” | PASS | 26.04.2018 | Ram Raja |  |
| **130** | toString() override  **(toString)** | Checks the correct subject is returned when not set i.e null | Class Document | Ram Raja & Aidan Reed | Uses **nullEmail** object | N/A | “[no subject]” | Null pointer exception | FAIL | 26/04/2018 | Aidan Reed | This fails as the method tries to return null as type string and causes a null pointer exception. Although the class description does not include this behavior the test was included to find potential bugs that cause the system to crass |
| “[no subject]” | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE102 |
| **131** | Check class member variables are private | Checks from address, to address, subject line and email message variables are all set to private | Code review | Ram Raja & Aidan Reed | N/A | N/A | True for all 4 variables | True | PASS | 27.04.2018 | Ram Raja |  |
| **132** | Check validity of Full email in setFrom **(setFrom)** | Checks the validity of the email address instead of just checking for a @ | Code Review | Ram Raja & Aidan Reed | Uses **nullEmail** object | **Sender**  **kBADEMAIL1 kBADEMAIL2 kBADEMAIL3 kBADEMAIL4** | Null when calling fromAddress | **kBADEMAIL1 kBADEMAIL2 kBADEMAIL3 kBADEMAIL4** | FAIL | 27/04/2018 | Aidan Reed |  |
| **Null** | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHGE103 |
| **133** | Check validity of Full email in setTo **(setTo)** | Checks the validity of the email address instead of just checking for a @ | Code Review | Ram Raja & Aidan Reed | Uses **nullEmail** object | **To**  **kBADEMAIL1 kBADEMAIL2 kBADEMAIL3 kBADEMAIL4** | Null when calling toAddress | **kBADEMAIL1 kBADEMAIL2 kBADEMAIL3 kBADEMAIL4** | FAIL | 27/04/2018 | Aidan Reed |  |
| **Null** | PASS | 27/04/2018 | Aidan Reed | Test passed after change to code with ChangeID CHG003 |

# Email Class 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** |
| **CHGE101** | **D2c7cf7** | **BUG102** | Return statement of emailMessage() getter method changed from “emailMessage()” method to “emailMessage” variable. | CompanyEmail (emailMessage) | 26/04/2018 | 27/04/2018 | Aidan Reed |  | Changed code removing method call and setting return value to emailMessage member variable |
| **CHGE102** | **5aa2fcf** | **BUG103** | Null Pointer exception when using toString when the subject has not been set. Add a condition to the method to check if null and print “[no subject set]” if null | CompanyEmail toString() | 26/04/2018 | 27/04/2018 | Aidan Reed |  | Added a null check to toString method to prevent null pointer exception |
| **2239fe2** | The conditions for the if statement in the function toString has been reversed. It now checks if the string if null before checking if its an empty string. If its null, it goes onto execute the statements inside the if statement. | CompanyEmail | 02/05/18 | 02/05/2018 | Athullya Roy | Aidan Reed |  |
| **CHGE103** | **Edd9154** | **BUG101** | SetTo and From methods check for @ in address but not position. Include regular expression to validate combinations of addresses. To do so I have added a new function called Email Parser which takes an address as input and returns true or false depending on whether it is valid or not. I have updated setTO and setFom methods to use the boolean result of email parser in the if statement | CompanyEmail | 26/04/018 | 27/04/2018 | Aidan Reed | Ram Raja | The regular expression included validates all possible combinations of emails. The set to and from methods were updated to include to call on this new regular expression and return a true or false value |
| **CHGE104** | **4e4f97b** | **N/A** | The return value of the functions setTo and setFrom has been changed from void to Boolean. It now returns true if the email format is correct and false if its incorrect. | CompanyEmail | 27/04/2018 | 02/05/2018 | Athullya Roy | Aidan Reed | See test IDs 114 and 116 |

# Email Class 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? | Who Fixed it | Does the bugged code relate to other functions? | Notes |
| BUG101 | The setFrom() and setTo() email methods only check for a “@” being present in the entered string, not at any specific position. Therefore, an incorrect email address could be entered. i.e. “psyrr1nottinghamac.uk@” | Lines 65-71 Lines 87-93 | 132  133 | CHGE103 | Specify clearly where the @ should be. | Med | 09/03/2018 | Yes | 27/04/2018 | Ram Raja | Aidan Reed | N.A | Added new method email parser to validate emails using regular expressions |
| BUG102 | emailMessage() method returns itself, not the value stored in the variable “emailMessage”. The method should be performing as a getter for the variable “emailMessage”. | Lines 46-48 | 101 102 103 104 111 112 119 120 | CHGE101 | Change return statement to the variable “emailMessage”, instead of the method “emailMessage()”. | High | 26/04/2018 | Yes | 27/04/2017 | Ram Raja Athullya Roy Aidan Reed | Aidan Reed | Both main and default constructors within the CompanyEmail class. | Found whilst performing tests on default/main constructors, Test IDs 101-104 – Ram Raja  Found this when performing tests 119-120 -Aidan Reed |
| BUG103 | Null Pointer exception when trying to print the subject of a email that has not been set | Line 138 | 130 | CHGE102 | Perform a check to see if the member variable is null and return the no subject string | High | 26.04.2018 | Yes | 27/04/2018 | Aidan Reed | Aidan Reed | N/A | Although the class description does not include this behavior the test was included to find potential bugs that cause the system to crash  27/04/2018 – Added OR condition to toString if statement |

# Project Unit Testing

The following tests use references to constants defined in the Junit test files at the start of the document. As such for easier comparison in the place the constants are defined below and are used in the Input, Expected Output and Actual Output columns to reduce repeating data and if the test inputs change in the future the document can be updated at a single point.

**kPTITLE1 =** “12345678”  
**kPTITLE2 =** “123456789000”  
**kDEFAULTTITLE =** “New Project”  
**kCONTACT1 = “**[test@gmail.com](mailto:test@gmail.com)”  
**kCONTACT2 =** “raiu9s@gmail.com”   
**kCONTACT3 =** [q39ikdf@outlook.com](mailto:q39ikdf@outlook.com)  
**companyProjectFirst\_Empty** (CompanyProject Object)  
**companyProjectSecond\_Phases** (CompanyProject Object) – Empty used for testing the phases   
**companyProjectThird\_Complete**(CompanyEmail Object) – Title: **kPTITLE1** Contacts: **kCONTACT1, kCONTACT2, kCONTACT3**)  
**companyProjectForth\_Phases** (CompanyProject Object) – Used for phase testing adding emails and moving between stages

# Project Unit Testing

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Function Name** | **Test Aim** | **Source: Spec or code inspection** | **Test Created By** | **Pre-Requisites** | **Inputs** | **Expected Output(s)** | **Actual Output** | **Pass / Fail** | **Date (Completed)** | **Tested By** | **Notes** |
| **201** | Company Project Constructor | Checks if default constructor sets up correctly with array lists initialized | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object no values have been set | N/A | 1 (Int) - PID  **kDEFAULT TITLE** (String)  True (Boolean) -Empty Contacts  0 (Int) - PhaseID  0 (Int) – Number Emails | 1 (Int) - PID  **kDEFAULT TITLE** (String)  True (Boolean) -Empty Contacts  1 (Int) - PhaseID  0 (Int) – Number Emails | FAIL | 27.04.2018 | Christian Stubbs | The test failed as the output for the phase id was initialized to 1 not the start of the array bug created– JNG |
| Same as above with Phase ID = 0 (INT) | PASS | 02/05/2018 | Aidan Reed | The test ow produces the correct output after the BUG201 and CHGE201 were acted up on. AR – 02/05/2018 |
| **202** | Company Project Constructor | Checks if main constructor sets up correctly with title parameter passed | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object no values have been set | N/A | 3 (Int) - PID  **kPTITLE1 TITLE** (String)  False (Boolean) -Empty Contacts  0 (Int) - PhaseID  0 (Int) – Number Emails | 3 (Int) - PID  **kPTITLE1 TITLE** (String)  False (Boolean) -Empty Contacts  1 (Int) - PhaseID  0 (Int) – Number Emails | FAIL | 30.04.2018 | Christian Stubbs | The test failed as the output for the phase id was initialized to 1 not the start of the array bug created– CS |
| Same out put as above apart from Phase ID was 0 (Int | PASS | 02/0/.2018 | Aidan Reed | The test ow produces the correct output after the BUG201 and CHGE201 were acted up on. AR – 02/05/ 2018 |
| **203** | getPID | Checks if the correct project ID is returned when a project is 1st created | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object no values have been set | N/A | 1 (Int) | 1 (Int) | PASS | 25.04.2018 | Christian Stubbs |  |
| **204** | getPID | Checks if the project ID increments correctly with the addition of two new projects | Class Document | Justin Ng & Christian Stubbs | Uses **companyThird** **\_Complete** object | N/A | 3 (Int) | 3 (Int) | PASS | 27.04.2018 | Christian Stubbs | Objects for these tests are created using the before class when the unit test is initiated |
| **205** | getPTitle | Checks if the correct project Title is returned when a project is 1st created | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object | N/A | **kPDEFAULTTITLE** (String) | **kDEFAULT TITLE** (String) | PASS | 25.04.2018 | Christian Stubbs |  |
| **206** | setPTitle | Checks if it is possible to set a project title which is less than 10 characters long | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object set title method and **kPTITLE1** | **kPTITLE1**  (String) | **kPDEFAULTTITLE** (String) | **kPDEFAULT TITLE** (String) | PASS | 25.04.2018 | Christian Stubbs |  |
| **207** | setPTitle | Checks if it is possible to set a project title which is more than 10 characters long | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object set title method and **kPTITLE2** | **kPTITLE2** (String) | **kPTITLE2** (String) | **kPTITLE2**  (String) | PASS | 25.04.2018 | Christian Stubbs |  |
| **208** | isContact | Checks if specified contact exists in project | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object | **kCONTACT1** (String) | False (Boolean) | False (Boolean) | PASS | 25.04.2018 | Christian Stubbs |  |
| **209** | isContact | Should add new contact then check if it can be read back | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object and add contact method | **kCONTACT1** (String) | True (Boolean) | True (Boolean) | PASS | 25.04.2018 | Christian Stubbs |  |
| **210** | addContact | Should be able to add two new contacts and use isContact to check if they exist in the Array List | Class Document | Justin Ng & Christian Stubbs | Uses **companyFirst** **\_Empty** object and add contact method | **kCONTACT2** (String)  **kCONTACT3** (String) | True  (Boolean)  True  (Boolean) | True (Boolean) True (Boolean) | PASS | 25.04.2018 | Christian Stubbs |  |
| **211** | getEmailForPhase | Tests to see if the function returns the emails in the current phase of a new project with no emails. | Class Document | Christian Subbs | Uses **companyForth** **\_Phases** | N/A | 0 (Int) | 0 (Int) | PASS | 27.04.2018 | Christian Stubbs |  |
| **212** | getEmailForPhase | Tests to see if the function returns the emails in the current phase of a new project after adding one email | Class Document | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** and adds email with **kSENDER, kRECPIENT, kSUBJECT, kBODY** | N/A | 1 (Int) | 1 (Int) | PASS | 27.04.2018 | Justin Ng |  |
| **213** | getEmailForPhase | Tests when moving to next stage email count is 0 when moving to next stage after adding 1000 emails at the previous stage | Class Document | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** and adds 1000 email with **kSENDER, kRECPIENT, kSUBJECT, kBODY** | N/A | 0 (Int) | Error: null pointer | FAIL | 02.05.2018 | Justin Ng | This fails due to a bug in the program where the array list is not initialized at the next stage. See BUG204 |
| 0 (Int) | PASS | 03.05.2018 | Justin Ng | After the changes made in BUG204 the null pointer exception no longer appears |
| **214** | getEmailForPhase | Iterates through the phases creating emails for each phase checking the size is equal to the number created and cycling through all stages – each stage has different number of emails | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** and adds 0 – first stage 1 – second – 2 third – 3 forth – 4 fifth and 5 in 6th STAGE | Each stage should have the number of emails outlined in pre requisite column | 0  1 2 3 4 5 | Error: Null pointer exception | FAIL | 02.05.2018 | Justin Ng | This fails due to a bug in the program where the array list is not initailised at the next stage. See BUG204 |
| 0 1 2 3 4 5 | PASS | 03.05.2018 | Justin Ng | After the changes made in BUG204 the null pointer exception no longer appears |
| **215** | getEmailsForPhase(**int** thePhase) | Test to see if this function returns a value of 0 when accessing a new project in the first phase | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  with no emails in first phase (0) | N/A | 0 (Int) | Error: Null pointer exception | FAIL | 02.05.2018 | Justin Ng | Failed initially because the constructor initializes the Emails array at value 1 not 0 – see BUG201 |
| 0 (Int) | PASS | 03.05.2018 | Justin Ng | The test passes after the BUG201 was rectified. See Change ID for bug for futher details. |
| **216** | getEmailsForPhase(**int** thePhase) | Iterates through all phases adding emails at each stage and moving to the next phase in the outer loop moves to the next stage and checks the previous stage | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  adds 0 – first stage 1 – second – 2 third – 3 forth – 4 fifth and 5 in 6th STAGE | Each stage should have the number of emails outlined in pre requisite column | 0  1 2 3 4 5 | Error: Null pointer exception | FAIL | 02.05.2018 | Justin Ng | This fails due to a bug in the program where the array list is not initailised at the next stage. See BUG204 |
| 0 1 2 3 4 5 | PASS | 03.05.2018 | Justin Ng | The test passes after the BUG201 was rectified. See Change ID for bug for further details. |
| **217** | getEmailsForPhase(**int** thePhase) | Adds email to first phase and then moves to next stage and checks the previous stage size is equal to 1 | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  and adds email with **kSENDER, kRECPIENT, kSUBJECT, kBODY** | N/A | 1 (Int) | Error: Null pointer exception | FAIL | 02.05.2018 | Justin Ng | Failed initially because the constructor initializes the Emails array at value 1 not 0 – see BUG201 |
| 1 (Int) | PASS | 03.05.2018 | Justin Ng | The test passes after the BUG201 was rectified. See Change ID for bug for further details. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **218** | nextPhase() | Tests to ensure the function moves a project from it’s initial stage to the next phase once | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** | N/A | 1 (Int) | 2 (Int) | FAIL | 02.05.2018 | Justin Ng | Failed initially because the constructor initializes the ProjectPhase to 1 |
| 1 (Int) | PASS | 03.05.2018 | Justin Ng | This test further identified issues that were found in constructor tests. Issue was fixed by fixing BUG201 |
| **219** | nextPhase() | Test moves a project from the initial stage until the last stage and ensures it’s phase id is 5 -phases 0 - 5 | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** | N/A | 5 (Int) | 7 (Int) | FAIL | 02.05.2018 | Justin Ng | Failed because the PhaseID is initialized to 1 not 0 – Arrays go from 0 – 5 in this implementation and because the project phase is incremented before checking |
| 5 (Int) | PASS | 03.05.2018 | Justin Ng | Test now produces the successful result as a result of the issue been fixed in the constructor in BUG201 and BUG202 the issue was fixed |
| **220** | nextPhase() | Test moves project to final stage and tries to go past the final stage | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases** | N/A | 5 (Int) | 7 (Int) | FAIL | 02.05.2018 | Justin Ng | The reason this failed is because the project phase is incremented before the conditional check |
| 5 (INT) | PASS | 03.05.2018 | Justin Ng | After the implementation of changes in BUG202 the projects cannot move past the last phase. AR – 03.05.2018 |
| **221** | getPhaseName | Will see if the correct name of the first phase is returned | Code Inspection | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  object with no emails added in initial stage | N/A | Feasibility | Design | FAIL | 02.05.2018 | Justin Ng | The test fails as it appears to miss the first stage of names form the project phases array –JNG 02.05.2018 |
| Feasibility | PASS | 03.05.2018 | Justin Ng | The initialization of project set the phase to 1 instead of 0 – as such the fixes in BUG 201 have rectified this issue – JNG 03.05.2018 |
| 222 | getPhaseName | Will see if the correct name of phase is returned on the second phase after moving along 1 phase | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  object with no emails moved to next stage | N/A | Design | Implementation | FAIL | 02.05.2018 | Justin Ng | The test fails as it appears to miss the first stage of names form the project phases array – CS 02.05.2018 |
| Design | PASS | 03.05.2018 | Justin Ng | The initialization of project set the phase to 1 instead of 0 – as such the fixes in BUG 201 have rectified this issue |
| 223 | getPhaseName | Will see if the last name of phase is returned even though attempted moving past the last phase | Class Documents | Christian Stubbs & Justin Ng | Uses **companyForth** **\_Phases**  object with no emails moved to next stage | N/A | Completed | Null | FAIL | 02.05.2018 | Justin Ng | The test fails as it appears to miss the first stage of names form the project phases array |
| Completed | PASS | 03.05.2018 | Justin Ng | The initialization of project set the phase to 1 instead of 0 and the next phase method allowed the phase id to move past the end of the project phases array. When BUG201 and BUG202 were fixed the system produced the correct result. |
| 224 | Get Phase ID | Checks if the correct phase ID is returned when a project is 1st created - should be initialized to represent start of array | Class Documents | Aidan Reed & Christian Stubbs | Uses **companyFirst** **\_Empty**  object | N/A | 0 | 1 (Int) | FAIL | 25.04.2018 | Christian Stubbs | This is because the constructor initialises it to 1 instead of 0 |
| 0 (Int) | PASS | 03.05.2018 | Christian Stubbs | Following the changes made to resolve Bug 201 the initial id is now correct |
| **225** | Get Phase ID | Checks if the correct phase ID is returned when project is moved from the 1st stage to the next | Class Document | Aidan Reed & Christian Stubbs | Uses **companyFirst** **\_Empty** object and moves to next stage | N/A | 1 (Int) | 2 (Int) | FAIL | 25.04.2018 | Christian Stubbs | The tests fails for the same reasons test 224 fails for relating to BUG201 |
| 1 (Int) | PASS | 03.05.2018 | Christian Stubbs | After Justin made the amendments for BUG201 the system now produces the correct result – CS 03.05.2018 |
| **226** | Get Phase ID | Checks the phase ID moving the project through all stages | Code Inspection | Aidan Reed & Christian Stubbs | Uses **companySecond** **\_Phases** object and checks id at all stages | N/A | 0  1 2 3 4 5 | 1 2 3 4 5 6 | PASS | 25.04.2018 | Christian Stubbs | The phase ID is one off in all stages due to a bug logged as BUG201. |
| 0 - 5 | PASS | 03.05.2018 | Christian Stubbs | Produces correct output after bug fix CS – 03.05.2018 |
| **227** | Get Phase ID | Checks if the correct phase ID is returned when project attempted to be moved passed 6th stage | Class Document | Aidan Reed & Christian Stubbs | Uses **companySecond** **\_Phases** object and moves to last stage using for loop | N/A | 5 (Int) | 7 (Int) | FAIL | 25.04.2018 | Christian Stubbs | Fails due to BUG201 and because the next phase is incremented before it is checked BUG202 |
| 6 (Int) | FAIL | 02.05.2018 | Christian Stubbs | Test still fails due to a bug in comparison of nextPhase – BUG202 see changelog for bug for more information |
| 5 (Int) | PASS | 03.05.2018 | Christian Stubbs | Passes after later change made for BUG202 – Change ID 202 |
| **228** | Get Phase ID | Checks if the correct phase ID is returned when project is moved from the last stage to the previous | Class Document | Aidan Reed & Christian Stubbs | New Project Object | N/A | 4 (Int) | 7 (Int) | FAIL | 25.04.2018 | N/A | Currently no functionality to go to previous stage.  Feature Not implemented – Test commented out on line 512 – CS 03.05.2018 |
| **229** | Get Project Contacts | Checks if the get project contacts method returns an object of type array list | Class Document | Aidan Reed & Justin Ng | Uses **companyFirst** **\_Empty** object | N/A | (Array List) Instance of Array List = true | Instance of Array List = true | PASS | 25.04.2018 | Justin Ng |  |
| **230** | Get Project Contacts | Checks the get project contacts method returns a string equal to the first element input using the add contact method | Class Document | Aidan Reed & Justin Ng | Uses **companyThird** **\_Complete** object which has 3 contacts kCONTACT 1/2/3 | **kCONTACT1** (String) | **kCONTACT1** (String) | **kCONTACT1** (String) | PASS | 25.04.2018 | Aidan Reed | Note the constant for the input is named : kCONTACT1 located in top of unit test |
| **231** | Get Project Contacts | Checks the type of the first element added into the contacts array list is of type string | Class Document | Aidan Reed & Justin Ng | Uses **companyThird** **\_Complete** object which has 3 contacts kCONTACT 1/2/3 | **kCONTACT1** (String) | (String) | (String) | PASS | 25.04.2018 | Justin Ng |  |
| **232** | toString Overide | Checks the type of the return value is of type string | Class Document | Aidan Reed & Justin Ng | Uses **companyFirst** **\_Empty** object | N/A | (String) | (String) | PASS | 25.04.2018 | Aidan Reed |  |
| **233** | toString Overide | Checks the toString is in the format Project Title [Phase Name] | Class Document | Aidan Reed & Justin Ng | Uses **companyThird** **\_Complete** object with title **kPTITLE1** with the phase in the initial stage | N/A | “kPTITLE1 [Feasibility]”  (String) | “kPTITLE1 [Design]” | FAIL | 25.04.2018 | Justin Ng | kPTITLE1 is a constant defined in the unit test. Fails due to BUG201 where phase id is 1 not 0 and misses the first element from project phases |
| “kPTITLE1 [Feasibility]” | PASS | 03.05.2018 | Justin Ng | After the code was reviewed and bug 201 was fixed the test produces correct result. Will update unit test comments |
| **234** | toString Overide | Checks the toString is in the format Project Title [Phase Name] for all stages of the project lifecycle | Class Document | Aidan Reed & Justin Ng | Uses **companyThird** **\_Complete** object with title **kPTITLE1** with the phase in the initial stage but will be incremented to the last stage | N/A | “**kPTITLE1** [Feasibility]”  “**kPTITLE1** [Design]”  “**kPTITLE1** [Implementation]”  “**kPTITLE1** [Testing]”  “**kPTITLE1** [Deployment]”  “**kPTITLE1** [Completed]”  All of type  (String) | “**kPTITLE1** [Design]”  “**kPTITLE1** [Implementation]”  “**kPTITLE1** [Testing]”  “**kPTITLE1** [Deployment]”  “**kPTITLE1** [Completed]”  Missing the first stage | FAIL | 25.04.2018 | Aidan Reed | All constants used are defined in this document above this table and also at the beginning of the Unit tests. The test fails because of an issue with the phase id which results in the wrong phase name been pulled back – AR 25.04.2018 |
| Same as expected result | PASS | 02.05.2018 | Aidan Reed | Fixed following changes on 02.05.2018 by Justin |

# Project Change Log

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Change ID** | **Bug ID** | **GIT Commit ID** | **Description of Change** | **Files Changed** | **Date Received** | **Date Changed** | **Initiator By (Who Changed)** | **Change Checked By** | **Notes** |
| **CHGE201** | **BUG 201** | **f51dc08** | The ProjectPhase is to be set to 0 in both CompanyProject constructors so that “Feasibility” is not skipped in CompanyEmailSystem | CompanyProjcet.java (line 23) | 02.05.2018 | 02.05.2018 | Justin Ng | Aidan Reed |  |
| **CHGE 202** | **BUG 202** | **f51dc08** | The comparison statement is changed to be compared to the length of project phases - 1 and the project phase incrementor was moved inside the else statement | CompanyProjcet.java (line 91 & line 94) | 02.05.2018 | 02.05.2018 | Justin Ng | Aidan Reed | The change didn’t fully resolve the issue as the project could still go past the end of the array because the conditional was not >= - AR 02.05.2018 |
| **fbea9fd** | The conditional only checked when the phase was greater than which resulted in the phase going 1 past the length of the array. The conditional should include a = parameter for greater or equals. | CompanyProjcet.java (line 91 & line 94) | 02.05.2018 | 02.05.2018 | Justin Ng | Aidan Reed | This now passes the test and ensures the phase does not go past the end of the projects array. - AR 02.05.2018 |
| **CHGE203** | **BUG 204** | **ca6ce13** | Add a statement to initialize the next phase of the ProjectEmail array list after the phase has been checked and incremented | CompanyProjcet.java (line 94-95) | 02.05.2018 | 03.05.2018 | Justin Ng | Christian Stubbs | The change included a new line of code after changing phases to initialize the next part of the array list. – CS 03.05.2018 |

# Project Bug List Fix

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |
| **BUG 201** | Project Phase Name set to Design when created instead of Feasibility - This also impacts the default constructor and Phase ID tests | Line 23 & 38: Company Project.class | 201  202 | CHGE201 | Set the Project phase to 0 more details in CHGE201 | Med | 25.04.2018 | YES | 02.05.2018 | Justin NG | N/A | The tests listed in test id are tests that first outlined this bug. |
| **BUG 202** | Project Phase ID goes over the number of different phases when going to next stage. | Lines 97-105: Company Project.class | 219 | CHGE202 | Move the phase++ inside the else clasue and change condition to length -1 | Med | 25.04.2018 | YES | 02.05.2018 | Aidan Reed / Christian Stubbs | N/A | The initial fix put in place did not fully fix the issue as the project phase was still incremented before the condition. After the second amendment the code was correct. – AR 02.05.2018 |
| **BUG 203** | No method to go to previous phase inside the CompanyProject Class |  |  |  |  | Low | 25.04.2018 |  |  | Aidan Reed | N/A | Low priority as description does not directly say such functionality should exist but was flagged during testing as test failed. This has been put on hold to be included in next iteration for manageability with new features |
| **BUG204** | When moving the project to the next stage the program fails with a null pointer exception when adding an email. This is because the constructor only initializes the first part of the Projects email array. | Line 102: CompanyProject. Class | 213 | CHGE203 | Add a statement to initialize the current phase after it has moved | High | 02.05.2018 | YES | 03.05.2018 | Christian Stubbs | N/A |  |

# Company Email System Unit / Unit Testing

The following tests use references to constants defined in the Junit test files at the start of the document. As such for easier comparison in the place the constants are defined below and are used in the Input, Expected Output and Actual Output columns to reduce repeating data and if the test inputs change in the future the document can be updated at a single point.

All constants referenced in the test tables are in bold and red.

kPTITLE1 = "Email System";

kDEFAULTTITLE = "New Project";

kSENDER = "joe.bloggs@gmail.com";

kRECIPIENT = "max.power@live.com";

kCONTACT1 = "test@gmail.com";

kCONTACT2 = "raiu9s@gmail.com";

kCONTACT3 = "q39ikdf@outlook.com";

kBADEMAIL1 = "joe.bloggs@";

kBADEMAIL2 = "joe.@bloggs";

kBADEMAIL3 = "joe.bloggs@gmail@.com";

kBADEMAIL4 = "@joe.bloggs@gmail.com";

kSUBJECT = "RE: Lorem ipsum";

kBODY1 = "Lorem ipsum dolor sit amet, consectetur adipiscing elit.";

String kBODY2 = "This is a test email for using unit testing";

String kBODY3 = "";

String kBODY4 = "Test Email";

To provide input into the system we have used two different approaches. The first approach is to create an input stream and then launch the main method. This is primarily used for testing the menu options of the system. The second approach used is to create a scanner object with a string inside and then pass this directly to the method. The string contains new line characters to simulate multiple user inputs for example: sysInput = "\n" + kSENDER + "\n" + kRECIPIENT + "\n" + kSUBJECT + "\n" + kBODY1;

# Company Email System Unit / Unit Testing

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Function Name** | **Test Aim** | **Source: Spec or code inspection** | **Test Created By** | **Prerequisite** | **Inputs** | **Expected Output(s)** | **Actual Output** | **Pass / Fail** | **Date (Completed by)** | **Tested By** | **Notes** |
| **301** | Main Method | Testing the user sees the correct menus when not in a project. | Class Description | Aidan Reed & Athullya Roy | N/A | 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 | PASS | 01/05/2018 | Athullya Roy |  |
| **302** | Main Method | Ensures the system exists at the main menu when user types X | Class Description | Aidan Reed & Athullya Roy | Create input streams and buffer the required inputs before launching main | “X” | System should output Good Bye! | Good Bye! | PASS | 01/05./2018 | Athullya Roy |  |
| **303** | Main Method | Tests the ArrayList can grow to large size | Class Description | Aidan Reed & Athullya Roy | Create 10,000 Projects after initializing project array | N/A | When checking size of array list should be 10000 | Error: Complication | FAIL | 01/05/2018 | Athullya Roy | Error: Complication Error method is not public and Projects array is not public |
| Error: No Line found | FAIL | 02/05/2018 | Athullya Roy | Failure: no line found CompanyEmailSystem - 112 |
| 10000 | PASS | 03.05.2018 | Athullya Roy | Passed after adding a new line to the input stream before hand full notes can be found under BUG301 |
| **304** | List Project | Checks that the function returns the correct number of projects created | Class Description | Aidan Reed & Athullya Roy | 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 | FAIL | 01/05/2018 | Aidan Reed | Error: Complication Error method is not public |
| Error: No Line found | FAIL | 02/05/2018 | Aidan Reed | Failure: no line found CompanyEmailSystem – 112 |
| 5 Emails created in expected output format | PASS | 03.05.2018 | Aidan Reed | Passed after adding a new line to the input stream before hand full notes can be found under BUG301 |
| **305** | List Project | Checks projects are listed correctly with the correct stages and email count. 6 projects should be created one for each phase each with 0 email counts | Class Inspection | Aidan Reed & Athullya Roy | Creates 6 email projects and sets the correct phase | N/A | 1. **kPTITLE**1 + I [Feasibility] – 0 Emails 2. **kPTITLE1** + I [Design] – 0 Emails | Error: Complication | FAIL | 01/05/2018 | Athullya Roy | Error: Complication Error method is not public |
| Error: No Line found | FAIL | 02/05/2018 | Athullya Roy | Failure: no line found CompanyEmailSystem – 112 see BUG301 |
| … 0emails | FAIL | 03/05/2018 | Athullya Roy | No space between count and word emails – logged |
| Expected Result | PASS | 03.05.2018 | Athullya Roy | After changes were made relating to BUG303 the system now produces string in correct format |
| **306** | List Project | Checks the correct formatting when | Code Inspection | Aidan Reed & Athullya Roy | Creates 1 project and adds 10000 emails to the project | N/A | 1. kPTITLE1 [Feasibility] – 10000 emails | Error: Complication | FAIL | 01/05/2018 | Aidan Reed | Error: Complication Error method is not public |
| Error: No Line found | FAIL | 02/05/2018 | Aidan Reed | Failure: no line found CompanyEmailSystem – 112 see BUG301 |
| kPTITLE1 [Feasibility] – 10000 emails | PASS | 03.05.2018 | Aidan Reed | After changes made to inputs – see BUG301 The tests stopped producing errors and produced the correct result – Note test was run after BUG303 was fixed so formatting bug would of impacted this too |
| **307** | Add Project | Ensures the project title added as input is correctly the system provided feedback [Project added] | Class Description | Aidan Reed & Athullya Roy | Initialize the Projects array and create a project with title **KPTITLE1** | Project Title **KPTITLE1** | System output [Project added] | Error: Complication | FAIL | 01/05/2018 | Athullya Roy | Error: Complication Error method is not public |
| Error: No Line found | FAIL | 02/05/2018 | Athullya Roy | Failure: no line found CompanyEmailSystem – 112 – See BUG301 |
| System output [Project added] | PASS | 03.05.2018 | Athullya Roy | After changes made to inputs – see BUG301 The tests stopped producing errors and produced the correct result |
| **308** | Add Project | Ensures the project is created and title is set correctly | Class Description | Aidan Reed & Athullya Roy | 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 | FAIL | 01/05/2018 | Aidan Reed | Error: Complication Error method is not public |
| Error: No Line found | FAIL | 02/05/2018 | Aidan Reed | Failure: no line found CompanyEmailSystem – 112 – See BUG301 |
| **KPTITLE1** | PASS | 03/05/2018 | Aidan Reed |  |
| **309** | Add Project | Checks that the project is created if the user does not enter a name the project should be created with a default name “New Project” | Class Description | Aidan Reed & Athullya Roy | 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 | FAIL | 01/05/2018 | Athullya Roy | Error: Complication Error method is not public |
| Error: No Line Found | FAIL | 02/05/2018 | Athullya Roy | Failure: no line found CompanyEmailSystem – 112 – See BUG301 |
| Empty String | FAIL | 03.05/2018 | Athullya Roy | The system does not check the title of the project and instead creates empty string |
| New Project | PASS | 03/05/2018 | Athullya Roy | After fixes implemented in BUG303 System now produces correct result. |
| **310** | List Emails | Tests that the function returns the emails set in the current stage – in this test this will be the initial stage | Class Description | Aidan Reed & Athullya Roy | 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 | FAIL | 01/05/2018 | Aidan Reed | Error: Complication Error method is not public |
| Error: No Line Found | FAIL | 02/05/2018 | Aidan Reed | Failure: no line found CompanyEmailSystem – 112 |
| Returns the sender and subject in format :  From: Subject | PASS | 03/05/2018 | Aidan Reed | After input stream issues were fixed highlighted in bug 301 the test was able to run successfully |
| **311** | List Emails | Tests the function with an invalid phase ID | Code Inspection | Ram Raja | Initialize the Projects array and create a project | Phase ID 20000 into method ListEmails | Should not return any emails With an error message: Error: Unknown Phase | Error: Unknown Phase | PASS | 03.05.2018 | Christian Stubbs | Although system outputted error this was expected as the phaseID entered was invalid |
| **312** | List Phase | Checks the initial phase of a project with 5 emails added to the project | Code Inspection | Ram Raja & Aidan Reed | Initialize a project and add five emails – reset output streams then call list phases | N/A | 1. Feasibility – 5 Emails | [] Empty String | FAIL | 03/05/2018 | Aidan Reed | The system did not output anything – upon further investigation a bug within the function has been logged BUG304 |
| Feasibility – 5 Emails | PASS | 03/05/2018 | Aidan Reed | The fixes applied to BUG 304 through CHGE305 |
| **313** | List Phase | Creates a project and moves it through the 6 stages adding more emails to each phase as each phase passes. E.g:  Feasibility - 1 Email  Design – 2 Emails  At feasibility stage 1 email is added and then another 2 are added at Design – total 3 – repeat for rest of stage | Code Inspection | Ram Raja & Aidan Reed | Initialize the project array and add email with **KSENDER kRECIPIENT KSUBJECT KBODY1**  When iterating through the for loop | N/A | 1. Feasibility - 1 Emails 2. Design – 2 Emails 3. Implementation – 3 Emails 4. Testing – 4 Emails 5. Deployment – 5 Emails 6. Completed – 6 Emails | 1. Completed – 2 Emails 2. Completed – 3 Emails 3. Completed – 4 Emails 4. Completed – 5 Emails | FAIL | 03/05/2018 | Aidan Reed | The test result has two issues – the first is the stages are all the same with different email counts BUG304. The second is the number of phases returned back. The first and last stage are missing from the results BUG305 |
| Expected output – not included because lengthy | PASS | 03/05/2018 | Aidan Reed | After the issues were resolved in BUGS 304 and 305 the system now produces the correct result when calling this function. |
| **314** | List Contacts | Tests the function returns the correct number of contacts added to project | Code Inspection | Ram Raja & Aidan Reed | Initialize the project array adding contacts **KCONTACT1 kCONTACT2 kCONTACT3** | N/A | 1. **KCONTACT1** 2. **KCONTACT2** 3. **KCONTACT3** | 1. **KCONTACT1** 2. **KCONTACT2** 3. **KCONTACT3** | PASS | 03/05/2018 | Ram Raja | Note Constants used in these tests can be found above this table and also in the Junit test at the top of the code. |
| **315** | Add Email | Tests the function provides the correct prompts to the user in a normal scenario | Class Description | Ram Raja & Aidan Reed | Initialize a project and with title **kPTITLE1** | **kSENDER kRECIPIENT kSUBJECT1 kBODY1** | Which email address is this from?  Which email address is this to?  What is the Subject?  What is the Message?  Email added to **kPTITLE1** | Correct question flow but the system prompt [Email added to **kPTITLE1** [Feasibility]] was not expected | FAIL | 03/05/2018 | Ram Raja | The test needs expected output needs editing to include the current phase – The test was did not anticipate this. |
| Same as above but  [Email added to **kPTITLE1** [Feasibility]] | Correct Output | PASS | 03/05/2018 | Ram Raja | The test was modified due to a slight formatting error in the expected output |
| **316** | Main Method | Ensure the user sees the correct menus when in a project. | Code Inspection | Ram Raja & Aidan Reed | A project must be selected | Enter a number to access a project | 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 | Displays 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 | PASS | 03/05/2018 | Ram Raja |  |
| **317** | Next Phase | Ensures the system produces the correct system prompt when a user moves a project from the initial stage to the next | Class Document | Ram Raja & Aidan Reed | Create a project and move it to the next phase. | N/A | [Phase Changed: Email System [Design]] | [Phase Changed: Email System [Design]] | PASS | 03/05/2018 | Aidan Reed |  |
| **318** | Next Phase | Checks that the system provides the correct feedback when moving a project through all phases | Class Document | Ram Raja & Aidan Reed | Create a project and move it to through all stages project should have title “Email System| | N/A | [Phase Changed: Email System [Design]]  Phase Changed: Email System [Implementation]]  Phase Changed: Email System [Testing]]  Phase Changed: Email System [Completed]] | [Phase Changed: Email System [Design]]  Phase Changed: Email System [Implementation]]  Phase Changed: Email System [Testing]]  Phase Changed: Email System [Completed]] | PASS | 03/05/2018 | Ram Raja |  |
| **319** | Next Phase | Checks the user is presented with a message if the project is in the last stage and tries to move past | Code Inspection | Ram Raja & Aidan Reed | Create a project move to the last stage and call the next phase method one more time | N/A | Project already in last stage | Project already in last stage | PASS | 03/05/2018 | Ram Raja |  |
| **320** | Main Method | When a user provides a invalid input the user receives “Command not recognized” | Code Inspection | Ram Raja & Aidan Reed | N/A | “Z” | “Command not recognized” | Something went wrong : exception .. | FAIL | 03/05/2018 | Aidan Reed | The system throws an exception when an invalid input is provided and displays this to the user. Created BUG310 |
| “Command not recognized” | PASS | 04/05/2018 | Aidan Reed | The system no longer displays exception information after BUG FIX 310 |
| **321** | Main Method | Ensure the user sees the correct menus when in a project | Code Inspection | Ram Raja & Aidan Reed | 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 | Displays 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 | PASS | 03/05/2018 | Aidan Reed |  |
| **322** | Main Method | Test the user cannot access a nonexistent project. | Code Inspection | Ram Raja & Aidan Reed | N/A | Enter non-existent project numbers. I.e. negative numbers and non-existent positive numbers.  Numbers input:   1. “-1” 2. “-7” 3. “7” | Inability to access the projects requested.  “Command not recognized” | Passed for input of “-1”  Projects menu displayed for inputs of “-7”, “7”, “99”. | FAIL | 03/05/2018 | Ram Raja | The Project menus were displayed when passing values -7 7 and 99 |
| Command not recognized for all out of bounds inputs | PASS | 03/05/2018 | Aidan Reed | The Project menus were not displayed at all after the out of bounds measures were put in place in BUG 309 Fixes |
| **323** | Main Method | Return and display options to interact with existing projects. | Code Inspection | Ram Raja & Aidan Reed | projects (at least 3) must exist. | Select first project.  Select second project.  Select third project. | Display menu for first project.  Display menu for second project.  Display menu for third project. | First project not accessible. Displays main menu whenever “1” is input.  Second and third projects correctly display project menus. | FAIL | 03/05/2018 | Ram Raja | Unable to access the first as the system decrements the input value for selection by 1.  See BUG308/CHGE308. |
| Display menu for first project.  Display menu for second project.  Display menu for third project. | PASS | 04/05/2018 | Aidan Reed | After the changes were made in BUG fix 308 The system now performs as expected. See BUG308 and CHGE308 for more details |
| **324** | Main Method | Be able to exit a project and return to the main menu | Code Inspection | Ram Raja & Aidan Reed | A project must exist. | Select a project.  “2”  Exit project.  “X” | Return user to main menu from selected project. | Returns user to main menu from selected project. | PASS | 03/05/2018 | Ram Raja |  |
| **325** | Add Email Funcion | Tests when bad data is inputted into the system the system asks for the data again before moving onto the next stage. For example Invalid sender provided the system asks for the sender until it is valid. This tests should follow through the whole process | Code Inspection | Aidan Reed | A project is created and the add email function is called – project should have title **Kptitle1** | Senders:  **kBADEMAIL1 kSENDER**  Recipients: **kBADEMAIL2 kRECIPIENT**  Subjects: **kSUBJECT1**  Message: **kBODY1** | Which email address is this from?  Invalid Email: Which email address is this from?  Which email address is it to?  Invalid Email: Which email address is this to?  What is the subject?  What is the Message?  Email added to **Kptitle1** | Which email address is this from?  Which email address is it to?  What is the subject?  What is the Message?  Email added to **Kptitle1** | FAIL | 04/05/2018 | Aidan Reed | The test fails as there is no checking from within side the Add email method inside the main class. As such I have logged a bug for this BUG312 |
| Produces the correct output | PASS | 04/05/2018 | Aidan Reed | The test successfully passed after the fixes that were implemented to fix BUG312. The actions taken to fix the bug revolved around |

# Company Email System 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** |
| **CHGE301** | **B2283df** | **N/A** | 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 System (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. |
| **CHGE302** | **N/A** | **BUG301** | In method add project and add email there is a call to in.nextline that removes the user input and results. After looking at this further the line is required to prevent the program trying to add projects with no input as such for testing a \n needs to be added to the inputs | CompanyEmail System (emailMessage | 01/05/2018 | N/A | Aidan Reed | N/A | This change was not required |
| **CHGE303** | [**7c9152a**](https://github.com/psyar8/FSE_CourseWork_4/commit/7c9152ab6708ee232bb5750661200fb54d648271) | **BUG302** | Add an additional space into the string formatting of the List Projects Method | CompanyEmail System Line 112 | 03/05/2018 | 03/05/2018 | Christian Stubbs | Aidan Reed | After this change the tests now pass successfully with a change to the formatting – CS 03.05.2018 |
| **CHGE304** | **7e70150** | **BUG303** | User can create project with no subject. No checking is performed to see if the length of the title is less than 10. If so the default constructor for Company Project should be called. | CompanyEmail System  Line 120 | 03/05/2018 | 03/05/2018 | Christian Stubbs | Aidan Reed | After implementing the condition in add project the projects are created either with titles that are greater than 10 characters or with new project otherwise |
| **CHGE305** | **6a2c8a8** | **BUG304** | The for loop inside the List Phases method uses the less than operator this means that the method cuts off the first project phase and the last phase when calling the method. The comparison operator should be <=. | Line165 | 03/05/2018 | 03.05.2018 | Christian Stubbs | Aidan Reed | The original condition meant the first stage was never printed as the first phase id is 0 and also the last one never printed – this has now been resolved. – AR 03/05/2018 |
| **CHGE306** | **1cc84dd** | **BUG305** | Create a overloaded method for getPhaseByName that takes an integer and checks if the phase id passed is less than the current phase id if so it should return the phase name for that phase otherwise return the current phase – preventing been able to access past the end of the phase. | Company Project Class -119 | 03/05/2018 | 03/05/2018 | Aidan Reed | Christian Stubbs | The new function allows access to historical phases. This is only the first stage to fixing BUG305 as changes now need to be made in the ListPhase method – AR 03/05/2018 |
| **CHGE307** | **37859f0** | **BUG305** | The ListPhase method should pass a parameter to the newly created getPhaseByName method in CHGE306 | CompanyEmail System Line 177 | 03/05/2018 | 03/05/2018 | Aidan Reed | Christian Stubbs | The for loop now produces the correct phase names when listing the project phases. – AR 03/05/2018 |
| **CHGE308** | **4cf18f3 & 670fca8** | **BUG308** | Set the default state for current Project variable to -1 as it isn’t attainable by what the user needs to access – then update the conitionals for menus to only display when the variable is -1. When X is selected inside a project it should se it back to -1 | CompanyEmail System Line  Main Method | 03/05/2018 | 03/05/2018 | Aidan Reed | Ram Raja | The system still works as intended but now allows a user to enter 1 and select the first element instead of been displayed with a menu again. In future versions the menu should be it’s own method with simpler logic RR 03/05/2108 |
| **CHGE309** |  | **BUG309** | Add bounds when selecting a project. The value input must be greater than 0 and less than/equal to the number of existing projects in the ArrayList. | CompanyEmailSystem (Main Method) | 03/05/2018 | 03/05/2018 | Ram Raja | Aidan Reed |  |
| **CHGE310** | **5bf53c8** | **BUG310** | The system displays the exception outputs to the user when a simple system out and a helpful message can be used. Change includes a message saying “Command Unknown” | CompanyEmailSystem (Main Method) line 127 | 03/05/2018 | 04/05/2018 | Aidan Reed | Ram Raja |  |
| **CHGE311** | **715d2cc** | **BUG311** | When the test data is created the current project remains at the first project and should be reset back to the none project state -1. | CompanyEmailSystem (Main Method) line 78 | 03/05/2018 | 04/05/2018 | Aidan Reed | Ram Raja |  |
| **CHGE312** | **709a1f4** | **BUG312** | The add email method inside the Main method has been re-implemented to loop through each data input until the correct values are provided. This is done by calling the set methods on the Email object that provide validation of email addresses with regular expressions | CompanyEmailSystem (Main Method) | 04/05/2018 | 04/05/2018 | Aidan Reed | Ram Raja | This is likely to be the largest change to the overall structure of the system. The method has been implemented using the helper function provided from the email objects class rather than reinventing the wheel. |

# Company Email System Bug Fix

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 |
| BUG301 | 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 in fact needs to be entered twice | 110-111 | 304-310 | CHGE302 | Remove the nextLine method call on line 110 | Med | 01/05/2018 | YES | 03/05/2018 | Aidan Reed | N/A | No – other functions call this method | No fix was required a new line needed to be added to th test inputs before hand. |
| BUG302 | The format of the list projects function is slightly off – should output Title – number email instead of Title – numberemail | 112 | 305 | CHGE303 | Add an additional Space into the formatting of string | Low | 01/05/2018 | YES | 03.05.2018 | Aidan Reed | Christian Stubbs | N/A | See GIT and CHANGE ID for more details but a space was added to formatting |
| BUG303 | When a project is created an no title is provided the project should be called New Project- currently system produces empty subject | Line 120 | 308 | CHGE304 | Add a check for the length of the title provided in add projects function | Med | 03/05/2018 | YES | 03.05.2018 | Aidan Reed | Christian Stubbs | N/ | After implementing the proposed fix the system now adds projects correctly |
| BUG304 | When calling list phases function from a project in it’s initial stage it does not display the current phase and it’s email count it just displays the menu options again. | Line 165 | 312 | CHGE305 | Change for loop condition from < to <= | Med | 03/05/2018 | YES | 03.05.2018 | Aidan Reed | Christian Stubbs | N/A |  |
| BUG305 | When Calling List phases function when a project has gone through all the stages it should list each stage and the email count. The email count is correct but the phase name for each stage is the same for all stages meaning it reflects the current phase i.e.   1. Completed – 1 Emails 2. Completed – 2 Emails 3. Completed – 3 Emails | Line 177 |  | CHGE306  CHGE307 | Create an overloaded function for getPhaseByName in EmailProjects Class – That should output the phase name | Med | 03/05/2018 | YES | 03/05/2018 | Aidan Reed | Aidan Reed | getPhaseByName in CompanyProjects | After implementing the changes proposed in CHGE306 and CHGE307 the system now behaves as expected and produces the correct phase – AR 03/05/2018 |
| BUG308 | User is unable to access the first project, as well as methods in the correct order as “-1” is always being applied to the user’s inputted project selection.  This also creates ambiguity when selecting other projects, as selecting project 3 could take the user to project 2, etc. | 86-112 | 323 | CHGE308 | ~~Remove “-1” from value input/read by system when selecting project.~~  ~~Change: “currentProjShowing = Integer.parseInt(s)-1;”~~  ~~To: “currentProjShowing = Integer.parseInt(s);”~~  Change the not in project state from 0 to -1 and change some of the conditionals | High | 03/05/2018 | Yes | 03/05/2018 | Ram Raja | Aidan Reed | N/A |  |
| BUG 309 | User is shown project menu when inputting non-existent project values. | 71 | 322 | CHGE309 | Add bounds to project selection. User should only be able to select a project with a value greater than 0 in the ArrayList, and a value less than or equal to the size of the ArrayList.  Change: “else if (Integer.parseInt(s) != -1 );”  To: “else if (Integer.parseInt(s) > 0 && Integer.parseInt(s) <= AllProjects.size());” | Med | 03/05/2018 | Yes | 03/05/2018 | Ram Raja | Ram Raja | N/A |  |
| BUG310 | When a user provides an invalid input the user receives an exception presented to the screen | 127 | 320 | CHGE310 | Instead of outputting message to user the system should display “Command Unkown” | Med | 03/05/2018 | YES | 04/05/2018 | Aidan Reed | Ram Raja | N/A |  |
| BUG311 | During the main method when test data is created The current project is left in a state of the first project. | 78 | N/A | CHGE311 | Before the while loop the system should initialize the currentProject back to -1 | Low | 03/05/2018 | YES | 04/05/2018 | Aidan Reed | Ram Raja | N/A |  |
| BUG312 | Emails added to projects through the main method are not subject to validation and are added regardless | CompanyEmailSystem Add Email Method, 246-312 | 325 | CHGE312 | Reimplement the add email method to check for validation at each stage of user input | High | 04/05/2018 | YES | 04/05/2018 | Aidan Reed | Aidan Reed | N/A | The current method includes no validation. The proposed fix ensures emails have no errors at any stage before adding the email. |