**5. System Testing**

**5.1 System Test Cases**

There are a total of 22 System test cases covering the implemented use cases: Login, add class schedule, edit class schedule and password conflicts. For each user case covered, there are 3 sunny day and 3 rainy day test cases.

**5.1.1 Test Identification and Objective**

|  |  |  |
| --- | --- | --- |
| Case ID | | Purpose |
| System\_Login\_Sunny01 | | To verify that a registered user can login to the PMS. |
| System\_Login\_Sunny02 | |
| System\_Login\_Sunny03 | |
| System\_Login\_Rainy04 | | To verify that an unregistered user cannot login to the system. |
| System\_Login\_Rainy05 | |
| System\_Login\_Rainy06 | |
| System\_AddClass\_Sunny07 | | To verify that the user can properly register classes to the database |
| System\_AddClass\_Sunny08 | |
| System\_AddClass\_Sunny09 | |
| System\_AddClass\_Rainy10 | | To verify that the user can’t register a class with an incorrect value to the database. |
| System\_AddClass\_Rainy11 | |
| System\_AddClass\_Rainy12 | |
| System\_EditSchedule\_Sunny13 | | To verify that a professor can update his schedule |
| System\_EditSchedule\_Sunny14 | |
| System\_EditSchedule\_Sunny15 | |
| System\_EditSchedule\_Rainy16 | | To verify that an incorrect class ID or an employee without the proper credentials can’t update the database. |
| System\_EditSchedule\_Rainy17 | |
| System\_EditSchedule\_Rainy18 | |
| System\_PasswordConflicts\_Sunny19 | | To verify that the app closes if enough failed logins are attempted. |
| System\_PasswordConflicts\_Sunny20 | |
| System\_PasswordConflicts\_Sunny21 | |
| System\_PasswordConflicts\_Rainy22 | To verify the app closes if enough failed logins are attempted | |
| System\_PasswordConflicts\_Rainy23 |
| System\_PasswordConflicts\_Rainy24 |

**5.1.2 Test Criteria and Procedures**

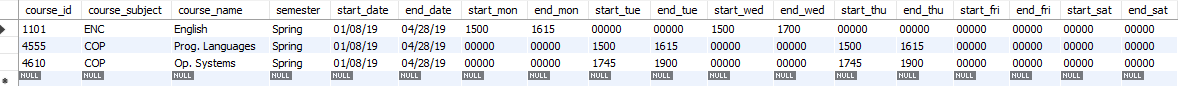
Test Criteria

Each of the implemented test cases result in the display of a new window, the change of data in the database, display of an error message, or to no output. Therefore, valid inputs will result in the display of a new window or that alter the information in the database. Invalid inputs will result in the display of an error message or to no output.

Test Procedure

A MySQL server running on localhost:3306 is used to store the schedule. The system tests are performed using Rational Functional Manager v9.2.1. The corresponding SQL connector libraries were added to RFT in order to reinitialize the database before tests. Sunny days and rainy days scripts and datapools are used to perform the tests.

Initial State

****

**5.1.3 Test Cases**

|  |  |
| --- | --- |
| Test ID | System\_Login\_Sunny01 |
| Test Purpose | To test that a registered user can properly login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “pclarke”  Password: “1234” |
| Expected output | The app opens PSM Main Menu |

|  |  |
| --- | --- |
| Test ID | System\_Login\_Sunny02 |
| Test Purpose | To test that a registered user can properly login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “user1”  Password: “1234” |
| Expected output | The app opens PSM Main Menu |

|  |  |
| --- | --- |
| Test ID | System\_Login\_Sunny03 |
| Test Purpose | To test that a registered user can properly login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “user2”  Password: “5678” |
| Expected output | The app opens PSM Main Menu |

|  |  |
| --- | --- |
| Test ID | System\_Login\_Rainy04 |
| Test Purpose | To test that an unregistered user can’t login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “pclarke”  Password: “wrongpassword” |
| Expected output | The app displays “Incorrect Username/Password” |

|  |  |
| --- | --- |
| Test ID | System\_Login\_Rainy05 |
| Test Purpose | To test that an unregistered user can’t login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “wronguser”  Password: “1234” |
| Expected output | The app displays “Incorrect Username/Password” |

|  |  |
| --- | --- |
| Test ID | System\_Login\_Rainy06 |
| Test Purpose | To test that an unregistered user can’t login. |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Username: “wronguser”  Password: “wrongpassword” |
| Expected output | The app displays “Incorrect Username/Password” |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Sunny07 |
| Test Purpose | To verify that the user can properly register classes to the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Add Class Schedule”  Semester: “Spring”  Start Date = “01/07/19”  End Date = “04/25/19”  Subject = “COP”  Course Number = “4710”  Course Name = “Database”  Start time Monday = “1825”  End Time Monday = “1940”  Start Time Wednesday = “1825”  End Time Wednesday = “1940” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Sunny08 |
| Test Purpose | To verify that the user can properly register classes to the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Add Class Schedule”  Semester: “Spring”  Start Date = “01/11/19”  End Date = “04/29/19”  Subject = “CEN”  Course Number = “4072”  Course Name = “Software Testing”  Start time Tuesday = “1500”  End Time Tuesday = “1700”  Start Time Thursday = “1500”  End Time Thursday = “1700” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Sunny09 |
| Test Purpose | To verify that the user can properly register classes to the database |
| Test Setup | Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Add Class Schedule”  Semester: “Spring”  Start Date = “08/12/19”  End Date = “12/10/19”  Subject = “CHM”  Course Number = “1045”  Course Name = “Chemistry 1”  Start time Monday = “1225”  End Time Monday = “1440”  Start Time Wednesday = “1225”  End Time Wednesday = “1440” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Rainy10 |
| Test Purpose | To verify that the user can’t register a class with an incorrect value to the database. |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234”   Click “Add Class Schedule” |
| Test Input | Semester: “Spring”  Start Date = “08/12/19”  End Date = “12/10/19”  Subject = “CHM”  Course Number = “AAA”  Course Name = “Chemistry 1”  Start time Tuesday = “1200”  End Time Tuesday = “1400”  Start Time Thursday = “1200”  End Time Thursday = “1400” |
| Expected output | No expected output |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Rainy11 |
| Test Purpose | To verify that the user can’t register a class with an incorrect value to the database. |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234”   Click “Add Class Schedule” |
| Test Input | Semester: “Summer A”  Start Date = “05/10/19”  End Date = “07/27/19”  Subject = “COP”  Course Number = “-4555”  Course Name = “Fundamentals of Programming Languages”  Start time Tuesday = “1200”  End Time Tuesday = “1500”  Start Time Thursday = “1200”  End Time Thursday = “1500” |
| Expected output | No expected output |

|  |  |
| --- | --- |
| Test ID | System\_AddClass\_Rainy12 |
| Test Purpose | To verify that the user can’t register a class with an incorrect value to the database. |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234”   Click “Add Class Schedule” |
| Test Input | Semester: “Spring”  Start Date = “800/12/2019”  End Date = “12/10/2019”  Subject = “ENC”  Course Number = “1012”  Course Name = “Writing 2”  Start time Tuesday = “1200”  End Time Tuesday = “1400”  Start Time Thursday = “1200”  End Time Thursday = “1400” |
| Expected output | No expected output |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Sunny13 |
| Test Purpose | To verify that the user can properly register changes to their classes to the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select: 4555  Start time Monday = “1200”  End Time Monday = “1400”  Start Time Wednesday = “1200”  End Time Wednesday = “1400”  Start Time Friday = “1200”  End Time Friday = “1400”  Start Time Tuesday =””  End Time Tuesday = “”  Start Time Thursday =””  End Time Thursday = “” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Sunny14 |
| Test Purpose | To verify that the user can properly register changes to their classes to the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select: 4555  Start Time Tuesday =”1500”  End Time Tuesday = “1700”  Start Time Thursday =”1500”  End Time Thursday = “1700” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Sunny15 |
| Test Purpose | To verify that the user can properly register changes to their classes to the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select 4555  Semester: “Spring”  Start Date = “01/15/19”  End Date = “05/10/19” |
| Expected output | The app closes “Schedule Setup” |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Rainy16 |
| Test Purpose | To verify that the user can’t change certain attributes when updating the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select 4555  Course Name = “Prin. Software Testing” |
| Expected output | The app closes “Schedule Setup” and no changes are made to course\_name of the selected class. |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Rainy17 |
| Test Purpose | To verify that the user can’t change certain attributes when updating the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select: 4555  Semester: “Summer A”  Start Date = “05/20/19”  End Date = “07/28/19” |
| Expected output | The app closes “Schedule Setup” and no changes are to Semester for the selected class |

|  |  |
| --- | --- |
| Test ID | System\_EditClass\_Rainy18 |
| Test Purpose | To verify that the user can’t change certain attributes when updating the database |
| Test Setup | Reinitialize the database to its initial state  Login as:   * user: “Pclarke” * Password: “1234” |
| Test Input | Click “Edit Schedule”  Select: 4555  Course Number = “4073” |
| Expected output | The app closes “Schedule Setup” and no changes are made to the course\_id of the selected class. |

|  |  |
| --- | --- |
| Test ID | System\_PasswordConflict\_Sunny19 NEW |
| Test Purpose | To verify that the system closes with enough failed login attempts |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Login as:   * user: “pclarke” * Password: “1244” * User:”pclarke1” * Password:”1234” * User: ”pclakre1”   Password:”1244” |
| Expected output | The app closes and the user cannot log in. |

|  |  |
| --- | --- |
| Test ID | System\_PasswordConflict\_Sunny20 NEW |
| Test Purpose | To verify that the system closes with enough failed login attempts |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Login as:   * user: “pclarke” * Password: “1244” * User:”pclarke1” * Password:”1234” * User: ”user1” * Password:”5678” |
| Expected output | The app closes and the user cannot log in. |

|  |  |
| --- | --- |
| Test ID | System\_PasswordConflict\_Sunny21 NEW |
| Test Purpose | To verify that the system closes with enough failed login attempts |
| Test Setup | Reinitialize the database to its initial state |
| Test Input | Login as:   * user: “pclarke” * Password: “1244” * User:”pclarke” * Password:”5678” * User: ”pclarke1” * Password:”5678” |
| Expected output | The app closes and the user cannot log in. |

**5.2 Actual Test Results**

|  |  |
| --- | --- |
| Case ID | Result |
| System\_Login\_Sunny01 | Pass |
| System\_Login\_Sunny02 | Pass |
| System\_Login\_Sunny03 | Pass |
| System\_Login\_Rainy04 | Pass |
| System\_Login\_Rainy05 | Pass |
| System\_Login\_Rainy06 | Pass |
| System\_AddClass\_Sunny07 | Pass |
| System\_AddClass\_Sunny08 | Pass |
| System\_AddClass\_Sunny09 | Pass |
| System\_AddClass\_Rainy10 | Pass |
| System\_AddClass\_Rainy11 | Fail |
| System\_AddClass\_Rainy12 | Fail |
| System\_EditSchedule\_Sunny13 | Pass |
| System\_EditSchedule\_Sunny14 | Pass |
| System\_EditSchedule\_Sunny15 | Pass |
| System\_EditSchedule\_Rainy16 | Pass |
| System\_EditSchedule\_Rainy17 | Pass |
| System\_EditSchedule\_Rainy18 | Pass |
| System\_PasswordConflict\_Sunny19 | Pass |
| System\_PasswordConflict\_Sunny20 | Pass |
| System\_PasswordConflict\_Sunny21 | Pass |

Test Summary

System Testing

The coverage tool used for system testing is cobertura with apache ant. For the second deliverable, more than 80% of line coverage was obtained, and the reason branch coverage did not reach 80% is because many branches cannot be reached during a system test because they are in unimplemented methods.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Deliverable Number | Test Cases | Bugs | Line Coverage | Branch Coverage |
| 1 | 18 | 1 | 67% | 35% |
| 2 | 21 | 1 | 82% | 66% |

