# *Systems III (420-E31-HR)*

# *Assignment 3 – CES Testing*

Date assigned: Wednesday, October 15, 2017

Date due: **Tuesday, November 14, 2017, 11:50pm**

**Tbd add result reporting and metric (% tests run, % tests passed, severity/priority and conclusion)**

**Learning Objectives**

Upon successful completion of this assignment, the student will be able to:

* Derive test cases from a use case narrative
* Develop a test plan for an existing system
* Develop and execute test cases for functional testing using black box testing techniques
* Develop and execute test cases for usability, compatibility, security and system testing
* Create defect tickets in TFS as a result of testing

To do:

**Part A: CES Use Case Narrative Testing**

Save this document as **YourUserName\_E31\_A03\_Testing.docx** in your 420-E31 folderin your home drive.

1. Refer to the “Schedule Evaluation” use case narrative for CES, which is posted on Moodle as an appendix to this assignment. Develop a test case matrix, with the test case ID, condition, data elements and expected result. Complete the table, using V for valid data, I for invalid data, and N/A where the data element is not applicable for the condition.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Test case description | Input start date | Input end date | Input reminder date | Expected output | Result |
| 1. | Happy Path | V | V | V | Evaluation has been scheduled | Pass |
|  | Start date before beginning of semester | I | V | V | Please enter a start date after the semester has begun | Fail |
|  | Start date before current date | I | V | V | Please enter a start date after today | Fail |
|  | Start date comes after the end of the semester | I | I | V | Please enter a start and end date that come before the end of the semester | Fail |
|  | End date before start date | V | I | V | Please enter an end date the comes after the start date | Fail |
|  | The end date is after the end of the semester | V | I | V | Please enter an end date that comes before the end of the semester | Fail |
|  | The reminder date is before the start date | V | V | I | Please enter a reminder between the start and end dates. | Fail |
|  | The reminder date is after the end date | V | V | I | Please enter a reminder between the start and end dates | Fail |
|  | Start date and end date on same day, but first day of semester | I | V | V | Please enter a start date that is after the semester has begun | Fail |
| 1. T | The reminder date is on the start date | V | V | V | Evaluation has been schedules | Pass |
| 1. The | The reminder date is on the end date | V | V | V | Evaluation has been scheduled | Pass |
|  | The Start date and end date are the same | V | V | V | Evaluation has been scheduled | Pass |
|  | The end date is the last day of the semester | V | V | V | Evaluation has been scheduled | Pass |

1. Copy the matrix. Replace the V’s with valid test case values and the I’s with invalid test case values.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Test case description | Input start date | Input end date | Input reminder date | Expected output | Result |
| 1. | Happy Path | 25/10/17 | 1/11/17 | 31/10/17 | Evaluation has been scheduled | Pass |
| 2. | Start date before beginning of semester (imagine date was 04/08/17) | 22/08/17 | 1/11/17 | 31/10/17 | Please enter a start date after the semester has begun | Fail |
| 3. | Start date before current date | 22/08/17 | 1/11/17 | 31/10/17 | Please enter a start date after today | Fail |
| 4. | Start date comes after the end of the semester | 15/12/17 | 25/12/17 | 20/12/17 | Please enter a start and end date that come before the end of the semester | Fail |
| 5. | End date before start date | 1/11/17 | 25/10/17 | 31/10/17 | Please enter an end date the comes after the start date | Fail |
| 6. | The end date is after the end of the semester | 25/10/17 | 25/12/17 | 31/10/17 | Please enter an end date that comes before the end of the semester | Fail |
| 7. | The reminder date is before the start date | 25/10/17 | 1/11/17 | 24/10/17 | Please enter a reminder between the start and end dates. | Fail |
| 8. | The reminder date is after the end date | 25/10/17 | 1/11/17 | 2/11/17 | Please enter a reminder between the start and end dates | Fail |
| 9. | Start date and end date on same day, but first day of semester | 24/08/17 | 24/08/17 | 24/08/17 | Please enter a start date that is after the semester has begun | Fail |
| 10. | The reminder date is on the start date | 25/10/17 | 01/11/17 | 25/10/17 | Evaluation has been schedules | Pass |
| 11. | The reminder date is on the end date | 25/10/17 | 01/11/17 | 01/11/17 | Evaluation has been scheduled | Pass |
| 12. | The Start date and end date are the same | 25/10/17 | 25/10/17 | 25/10/17 | Evaluation has been scheduled | Pass |
| 13. | The end date is the last day of the semester | 25/10/17 | 11/12/17 | 1/12/17 | Evaluation has been scheduled | Pass |

**Part B: CES Test Plan and Test Execution**

1. Create a new document named **YourUserName\_E31\_A03\_CES\_Test\_Plan.docx** in your 420-E31 folderin your home drive.

The document will contain a test plan for the Course Evaluation System. Add a title page, a table of contents, and the following 8 section headers to the document: Introduction, Black Box Test Cases, Usability Test Cases, Compatibility Test Cases, System Test Cases, Security Test Cases, Exploratory Test Cases, and Conclusion. Include a brief introduction, indicating that the test plan provides a summary of the results of tests performed for each of the types of tests identified.

1. A testing version of the Course Evaluation System is available at http://cstest.cegep-heritage.qc.ca/projects/CES\_E31\_**USERID**/Login.aspx. (Note that you need to insert your userid to get your version of the system. The userid is your old firstinitial,lastname based userid, not the student number)

In order to access the testing version, the following accounts can be used:

|  |  |
| --- | --- |
| Teachers who create and schedule the evaluation of their courses and receiving the results provided by the student evaluations. The teacher is part of the evaluation team for another teacher. (The teacher will show up as Allan). | Username: **userte**  Password: **userte** |
| Another teacher in the system. (The teacher will show up as Richard). This is the teacher that is being evaluated by the evaluation team. | Username: **userce**  Password: **userce** |
| Program coordinators who create the program-common questions for the surveys and views program evaluation statistics. (The coordinator will show up as Ron). | Username: **userco**  Password: **userco** |
| Administrator of the system. (The administrator will show up as Maurice). | Username: **userad**  Password: **userad** |
| Students who provide feedback to teachers by doing the online course evaluation for their respective courses. These accounts are second year Computer Science students taking 420-C30.  There are 2 student accounts that can be used: 1111111 and 2222222.  The password is the same as the username. | Username: 1111111  Password: 1111111  Username: 2222222  Password: 2222222 |

1. The following is a list of the rules in the Course Evaluation System:
   1. Students’ responses are anonymous
   2. Students can only complete an evaluation once
   3. Students can only evaluate courses they are enrolled in
   4. Teachers’ results are private and confidential

The following is a list of features available in the Course Evaluation System:

* 1. The system has a common set of questions for all courses. The coordinator can maintain a common set of questions for the program or department. A teacher can add custom questions to their survey. The system combines the common questions, the program/department questions and the course specification questions into a single survey for each course.
  2. A teacher can create evaluations for their courses.
  3. A teacher can control the duration that the evaluation is available to the students. Deadlines assigned to the evaluations can be extended at any point before the deadline has been reached.
  4. A teacher can review their evaluation responses once the survey deadline has been reached.
  5. The system sends notifications and reminders to students to complete evaluations.
  6. A teacher can view statistical reports about their evaluations.
  7. A teacher can compare past year results against each other in order to observe any improvements or changes
  8. A coordinator can view the usage rates for their program or department.

1. Using the format below and equivalence partitioning **and** border value analysis black-box techniques, write the test cases to test the Schedule Evaluation form (teacher) **and** the Complete Evaluation (student) in the appropriate section in the test plan document. Include the role that is responsible for the testing. Run the tests and record any defects in TFS in the **E31Testing** project, with an appropriate title, description, priority, and severity. Include the steps to recreate the bug in the description. TFS can be accessed at <http://cstfs.cegep-heritage.qc.ca:8080/tfs>.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Plan for** | | | | | |
| **Test Case** | | **Sample Data** | **Calculation** | **Expected result** | **Test Passed** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 1. … | … |  |  |  |  |

1. Include at least ten different usability test cases for the system in the appropriate section in the test plan document. Usability testing includes consistency in the user interface, navigation, and ease of use. Include the role that is responsible for the testing. Run the tests and record any defects in TFS. Include the steps to recreate the bug in the description.
2. Include at least five different compatibility test cases for the system in the appropriate section in the test plan document. Run the tests and record any defects in TFS. Include the steps to recreate the bug in the description.
3. Using the format below, include at least ten different system test cases in the appropriate section in the test plan document. System test cases should include a ***chain of use cases*** that are interrelated to ensure that they function as expected. Include a unique system test case ID and title. Include the role that is responsible for the testing. Run the tests and record any defects in TFS. Include the steps to recreate the bug in the description. I’ve provided an example to start you off showing the level of detail expected. Note that in System Test cases, the chain of use cases for each test is to validate across all roles, views or functions as required.

|  |  |  |  |
| --- | --- | --- | --- |
| Test Id | Purpose | Scenario/Condition | Expected Result |
| 1 | Verify that a teacher can schedule an evaluation, a student can complete the evaluation, and the teacher can see the status and results of the evaluation. | Schedule Evaluation:  teacher schedules an evaluation to start one day in the future and end the following day for a course he/she teaches | Evaluation is not available to **students** until the start date that was scheduled.  Results are not available to **teacher** until after the end date.  **Students** can not evaluate after end date.  Ratings and comments are correct (match what was inputted).  Ratings and comments are anonymized (not identified to a specific student) to the **teacher** |

1. Include at least ten different security test cases for the system in the appropriate section in the test plan document. Security testing includes ensuring confidentiality, integrity, authentication and authorization as well as protection from attacks. Include the role that is responsible for the testing. Run the tests and record any defects in TFS. Include the steps to recreate the bug in the description.
2. Perform exploratory testing on the Course Evaluation System, recording your tests and record any defects in TFS. Include the steps to recreate the bug in the description.
3. Write a brief conclusion, which summarizes the number of defects found for each type of test and comment on the overall quality of the system.
4. Update the table of contents in the test plan.

**To submit**

When you have completed the assignment, upload the **YourUserName\_E31\_A03\_Testing.docx document** and **YourUserName E31\_A03\_CES\_Test\_Plan.docx** files to the Moodle page for this course.

**Marking Scheme**

|  |  |
| --- | --- |
|  | **Out Of** |
| **Part A: CES Use Case Narrative** |  |
| Question 1 - Test Case Matrix outline | 10 |
| Question 2 - Test Case Matrix data | 10 |
| **Part B : CES Test Plan Document** |  |
| Cover page & table of contents | 2 |
| Introduction | 5 |
| Schedule Evaluation (teacher) Test Cases | 15 |
| Complete Evaluation (student) Test Cases | 5 |
| Usability Test Cases | 10 |
| Compatibility Test Cases | 5 |
| System Test Cases | 10 |
| Security Test Cases | 10 |
| Exploratory Test Cases | 10 |
| Defects in TFS - recorded, well written with appropriate description and list of steps to reproduce problem. Appropriate priority/severity. | 10 |
| Conclusion | 5 |
| **Proper use of English** | 2 |
| **Assignment Organization** | 1 |
| **Total** | 110 |

Appendix A: Accounts/URLS

You will find your web site to test at:

[http://cstest.cegep-heritage.qc.ca/projects/CES\_E31\_**USERID**/](http://cstest.cegep-heritage.qc.ca/projects/CES_E31_USERID/)

For example, for Chris:

Website:

[http://cstest.cegep-heritage.qc.ca/projects/CES\_E31\_**CFOX**/](http://cstest.cegep-heritage.qc.ca/projects/CES_E31_CFOX/)

You will find your TFS projects to capture your defects at:

http://cstfs.cegep-heritage.qc.ca:8080/tfs/E31Testing/cfox