# *Development Project II (420-E63-HR)*

# *Assignment 7 – Release Candidate Production Ready and System Test*

Date assigned: Monday, April 8, 2018

Date due: **Wednesday, April 11, 2018, 11:50PM Part A,B – Test cases written & setup complete**

**Monday, April 16, 2018, 11:50PM Part C – Test cases executed, results reported**

**Friday, May 11, 2018, 11:50PM Part C – Test case executed (rerun), results reported**

**Learning Objectives**

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

* Develop a system test plan and system test cases
* Run a system test
* Document defects found as a result of a system test

**Students are required to pass this assignment in order to pass the course.** Students who complete and fail this assignment before late marks are deducted may correct and resubmit the assignment for a maximum mark of 60% less the late marks on the original assignment.

You may do research and investigation as a team, however the documentation and submissions for this assignment is **individual work only**.

To do:

# ~~Part A – System Test Plan~~

The purpose of the system test is to verify that the system functions according to the user stories by interacting with the application via the user interface and analyzing the output or results. In particular, the focus should be on the following aspects:

* Interaction of functions vs roles (note the individual functions were tested as acceptance criteria for each PBI).
* Usability
* Compatibility
* Security
* Business cycle testing

1. Create a new document named **YourUserName\_E63\_A07\_System\_Test** in your home drive.
2. The document will contain a test plan for the system you have worked on. Add a title page, a table of contents, and the following section headers to the document: Introduction, Functional Test Cases, Usability Test Cases, Compatibility Test Cases, Security Test Cases, and Business Cycle Test Cases, and Conclusion. Include a brief introduction of the contents of the document in the Introduction section. Have a minimum of 5 tests for each category.
3. For functional test cases, use the following template and example as a guide to writing the test cases. Functional tests are written from a user's perspective. These tests confirm that the system does what users are expecting it to. Functional test cases should include a series of steps that are interrelated to ensure that they function as expected and should address how two parts of the system relate to each other. These tests are feature interaction tests as individual functions are covered by PBI’s and their acceptance tests. The purpose of the system test is not to test all the error conditions. It is assumed that this was done during unit test. Include a unique system test case ID, prefixing it with FN.

**Functional Test Case Table**

| **Test ID** | **Purpose** | **Expected Result** |
| --- | --- | --- |
| FN01 | Verify the link between the scheduling of an evaluation and the availability of the evaluation to the student. | Students cannot see the evaluation before the start date.  Students can see the evaluation between the start date and the end date.  Students cannot see the evaluation after the end date. |

1. Usability testing involves testing for consistency in the user interface, navigation, and ease of use. Include usability test cases for the system in a Usability Test Case Table, using a format similar to the Functional Test Case Table and prefixing the ID with US.

1. Compatibility testing involves testing using different browsers, devices, and operating systems. Include compatibility test cases for the system in a Compatibility Test Case Table, using a format similar to the Functional Test Case Table and prefixing the ID with CO.

1. Security testing includes ensuring that only those actors specified to execute the user stories can do so and that the system is secure. Include security test cases for the system in a Security Test Case Table, using a format similar to the Functional Test Case Table and prefixing the ID with SE.

1. Business cycle testing emulates the activities performed on the system over time. Transactions and activities that would occur during at least two consecutive semesters or years should be executed. Include business cycle test cases for the system in a Business Cycle Test Case Table, using a format similar to the Functional Test Case Table and prefixing the ID with BC.

# Part B – System Test Setup

1. ~~Deploy your own private copy to CSTEST. This includes your own database and the latest version of the application. Publish to path Projects/test/~~*~~projectName~~*~~/~~*~~uname~~*
2. Start capturing the recipe on how to do this as you will need it to complete Assignment 6 – Runbook (i.e. how to deploy your application).
3. Setup the Test report based on the Test Results Template provided in Moodle. Call it **YourUserName\_E63\_Test\_Results.xlsm**

# Part C – System Test Execution and reporting

1. Run the system test cases that were identified, using the deployed application, your individual databases, and **representative** data:
2. Record the actual results in the Test Results spreadsheet as Passed/Failed, including the error if the test case failed, and record any defects in **your individual project in the E63Testing collection in TFS**. Include the steps to recreate the bug in the description in TFS. **DO NOT MAKE ANY CHANGES TO THE SYSTEM, unless you are specifically instructed to do so**. You will be evaluated based on the quantity and quality of bugs you find and on your description of the bugs, not the number of changes you make. See the “Heritage\_Test\_Results\_Template.xlsm”.
3. Perform exploratory testing on the system and record any defects in TFS. Include the steps to recreate the bug in the description.
4. Write a brief conclusion to the System Test Report, summarizing the number of defects found for each type of test and commenting on the severity of the defects and the overall quality of the system.
5. Update the table of contents in the test plan.
6. Resubmit the test plan and the test results into Moodle.

**To submit**

Submit portions to the appropriate folders in Moodle. Note the Rubric also provided in Moodle on how this assignment will be assessed.