***420-K30-HR – Assignment 2 System Testing, Meeting the Customer, Backlog Grooming, Sprint Planning***

Date Assigned: Tuesday September 17th

Date due: **Thursday 26th, 2023 @ 11:59PM in Moodle.**

You will be given time during class to work on this assignment. Each class will begin with an overview of the topic for the day and assignment expectations. As a team, you know your deliverables and should agree on when the group work vs individual work will be done.

**Note that if you believe you will not have time to complete the individual sections in the class for that day, it is your responsibility to get started on the work in advance. Even if you have completed the work for the day in advance, you are expected to attend the class that day to ensure you have completed the work for that day in accordance with expectations and are available to work on team efforts.**

|  |  |  |  |
| --- | --- | --- | --- |
| Part A | Sept 17 | Individual | System Test Plan |
| Part B | Sept 19 | Individual | System Test Execution and Reporting |
| Part C | Sept 19 | Individual, Group | Backlog Grooming, Bug Consolidation |
| Part D | Sept 24 | Group | Customer meeting |
| Part E | Sept 26 | Group | Sprint Planning and Commit |
| Part F | Sept 26 | Individual | Reflection and Peer assessment |

**No assignment submissions will be accepted after the final submission date, i.e., nothing submitted after Sept 26 @ 11:59PM in Moodle will be accepted.**

**If earlier parts are submitted late, the usual CS late policy (as per the course outline) applies up to the cut-off date. Missing interim delivery deadlines is strongly discouraged.**

Note: If you have completed the part to submit on that day early, work ahead on individual parts, or team parts if everyone else is ready.

**Learning Objectives**

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

* Develop a test plan for an existing system;
* Understand how to deploy the 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;
* Meet with the end user to gather requirements and priorities;
* Analyze, prioritize, and clean-up the backlog items; and
* Come up with an initial plan for the maintenance project.

To Start:

1. **Make sure you read the entire assignment to understand the scope of work and the delivery schedule.**

To Do:

## **Part A – System Test Plan (Individual work)**

## 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 aspects defined in the System Test template document provided with this assignment. The goal is to get good initial test coverage of functionality in your system.

## Create a new document named **YourUserName\_K30\_A02\_SystemTest** in your one drive based on the template provided with this assignment. Make sure you read the template and guidelines it contains. Remember, guidelines provide directions and should not be left in the finished product.

## The document will contain a test plan for the maintenance system you have been assigned to. Prepare the document to ensure you test the system and leave the necessary artifacts for the intended audience.

## Each test section should have a paragraph explaining the purpose for the tests, i.e., in the functional test section, define and describe the purpose of functional tests, etc.

## For functional test cases, use the following template and example for the Course Evaluation System (CES) as a guide for 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 or more parts of the system relate to each other. These tests are feature interaction tests (since individual functions would be covered by user stories 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 testing of the system. Include a unique system test case ID, prefixing it with FN. **Provide a minimum of 12 different functional tests**.

## **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. |

## 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. **Provide a minimum of 5 different usability tests.**

## 

## Compatibility testing involves testing using different browsers, devices, and operating systems (as applicable to the requirements of the system). 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. **Provide a minimum of 2 different compatibility tests.**

## 

## 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. **Provide a minimum of 4 different security tests.**

## 

## 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. **Provide a minimum of 2 different business cycle tests.**

## Scale testing requires that you prove your system runs at the scales specified by the user. Capture the scale values and explain your plan for what testing will be covered at scale.

## Performance testing requires that you prove your system runs at the performance limits specified by the user. Capture the performance metrics and explain your plan for capturing the metrics and identifying any failures. Note, the performance standard for this course is that every page must be rendered within 2 seconds (worst-case time).

## **Part B – System Test Execution and Reporting (Individual work)**

1. Deploy your own private copy of the latest version of your project with your own database.
2. Make sure you note any changes to your project Runbook that will be needed. Capture this in a file called **YourUserName\_K30\_Deployment\_RunbookUpdates.docx.** To do this, copy the relevant portions of the existing Runbook and update them as needed for this part.
3. Set up the Test report based on the Test Results Template provided with this assignment. Rename the file **YourUserName\_K30\_Test\_Results.xlsm.**
4. Run the system test cases that you identified in Part A, using your personal deployed application, your individual databases, and **representative** data.
5. 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 the Bugs tab of the spreadsheet. (For the bugs, you will be evaluated based on the quantity/quality of bugs you find and on your description of the bugs.)
6. Perform additional exploratory testing on the system and record any defects in the test report as well. Try to reproduce items you found in Assignment 1 Part B. Ensure that any new items/bugs that you find and any new ones you found in Assignment 1 are added to the bottom of the bug list on the Bugs tab. Include the steps to recreate the bug in the description.
7. Write a brief conclusion/summary to the System Test Plan/Report (i.e., copy **YourUserName\_K30\_A02\_SystemTest** from part A, and rename it **toYourUserName\_K30\_A02\_SystemTest\_updated)**, 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.
8. Update the revision table, and table of contents in the test plan.
9. Resubmit the updated test plan (the word document), the test results (excel file), and your run book updates (word document) to Moodle.

## **Part C – Consolidating Bugs and Backlog Grooming (Group and individual)**

1. Meet with your team to come up with a consolidated list of bugs from your individual system testing.
2. Ensure there are no duplicates.
3. Ensure that a different member of the team is able to reproduce the problem.
4. Check to see if the bug/issue already exists in Azure DevOps.
5. For the ones that do not already exist, raise a bug/issue in Azure DevOps, being sure to apply good problem reporting rules to the process.
6. As a team, prepare a list of all the new items raised in Azure DevOps (using the format shown below) and have one team member submit it as **YourTeamName\_K30\_A02\_PartCNewAzureIssues.docx**

Raised by me:

|  |  |
| --- | --- |
| Azure DevOps # (& link) | Description |
| [Task 2185](https://csazure.cegep-heritage.qc.ca:8080/F2023-DevProject/Project%203%20-%20Nursing%20App/_workitems/edit/2185) | Option to delete a course if it doesn’t contain any course runs should be added |

1. Go back to your original bug list and provide a status update for each or your bugs. If it was a duplicate of one already in Azure DevOps, provide the Azure DevOps number. If it was part of a new bug, indicate the new Azure DevOps number raised. If it was part of a new bug that was raised, indicate the name of the other team member who was able to reproduce it. You may choose to submit this as a table in word, or as an updated version of your excel spreadsheet Bugs tab, with columns added. Name your submission **YourName\_K30\_A02\_PartDBugStatus.docx or .xlsm.**
2. As a team, groom the backlog. Make sure that all user stories, tasks, and bugs that are not complete are assigned to the future release in your Azure DevOps project and are prioritized. Make sure you consider the input you received from your customer when you are doing this.

## **Part D – Meeting the Customer**

1. Part C is a group effort. One member of the team should submit the final document to Moodle. The meeting with the customer will be scheduled for class time on September 24th
2. Create a new document named **YourTeamName\_K30\_A02\_MeetingTheCustomer.docx.**
3. Meet with your team to review the results of your system testing and to discuss key issues you believe should be addressed during the maintenance project. Record those notes in a first section of **YourTeamName\_K30\_A02\_MeetingTheCustomer.docx** called “Planning”.
4. Prepare for the meeting with the customer and agree on note taking and demonstration duties for the meeting. Record this decision in the planning section of your document. Your plan should consider that you will have 40 minutes for this first meeting.
5. Meet with the customer to gather their input on the system and what their priorities are for the updates to the system.
6. Generate Meeting Minutes and send them via email to all attendees. Use the template provided on Moodle. Be sure to capture who was present, what went on, key decisions and agreed upon next steps or action items. Make sure this is reviewed and agreed upon by the team and your professor. Include a copy of those minutes with your Part C submission, and reference them in a second section of **YourTeamName\_K30\_A02\_MeetingTheCustomer.docx** called “Customer Meeting”.
7. Capture new user requests from the meeting in Azure DevOps. Use user story format, “As an administrator, ….”. Make sure that you also update any previous user stories or tasks with new information. List the newly created Azure DevOps/updated items in a table like the one shown below in a third section of **YourTeamName\_K30\_A02\_MeetingTheCustomer.docx** called “Backlog Updates”.

|  |  |
| --- | --- |
| Azure # (and link) | Description |
|  |  |
|  |  |
|  |  |

## **Part E – Sprint Planning and commit**

1. Meet as a team to discuss the plan for Sprint 1. At the beginning of the work period I will identify the Assistant Project Managers (APMs) for Sprint 1. I will also review the process that you are expected to follow.
2. Determine what you intend to complete for Sprint 1. Your team budget for Sprint 1 is (2 weeks x ponderation hours per week per person (12) x number of team members). So, for teams of 5 your budget is 120 hrs, and for the team of 4 it is 96 hrs.
3. If your project requires a port to .NET 8.0, make sure that as part of this session you spend a little time determining the scope / investigate the changes anticipated. This is a must for sprint 1.
4. Decompose and size your top priority items on the backlog list. You only need to go as far as you think you could possibly get during the first sprint/iteration. i.e., if there’s 10 items and you only think the first 8 are achievable, you don’t need to size the items you wouldn’t be able to address. Meet as a team to discuss what needs to be done. Come up with a consensus on steps, who is doing what, etc. Come up with work items that are needed and attach to defects and user stories as needed. If new development, think of user stories. Include time/effort estimates in the task work items/defects/user stories. Use either planning poker or relative estimation for your sizing. Make sure you enter your estimates as **remaining work** in Azure DevOps.
5. Move your items to the Fall 2023 Sprint 1 Backlog.
6. Record and submit your proposed plan as **YourTeamName\_K30\_A02\_PartE\_Sprint1\_Plan.docx**. This plan should summarize the sprint goal, the items to be addressed, and the budget. Use the template provided on Moodle. With your prof, review your plan, estimates, proposal on the process, roles, timeframe and content and start your first sprint.

## **Part F – Reflection (Individual work)**

1. Complete the Teammates Peer assessment providing thought-out, professional feedback.
2. Create a file called **YourName\_K30\_A02\_PartF\_Reflection.docx** for the answers to this section.
3. List three technical difficulties you ran into during this assignment. For each one, describe how you resolved it, and what you learned from the experience.
4. List three interpersonal or team dynamics difficulties you ran into during this assignment. For each one, describe how you resolved it, and what you learned from the experience.
5. What are your top 3 concerns about Sprint 1? Describe what they are and how you intend to mitigate these issues.

*Note: If you haven't experienced 3 interpersonal conflict issues or team dynamics issues, please identify as many as you can and supplement them with other things that you have experienced in past teams (for instance HVK last year, work, sports teams, etc. and how you overcame them and how that could apply in a situation like ours).  Please make it clear which ones are about your Systems Maintenance team vs. other team experiences.  Same would apply to the technical issues items.*

**Marking Scheme**

|  |  |
| --- | --- |
| **Part A - System Test Plan (Individual) /65** |  |
| Report Format, introduction & conclusion | 5 |
| Functional test cases (12) | 24 |
| Usability test cases (5) | 10 |
| Compatibility test cases (2) | 4 |
| Security Test cases (4) | 8 |
| Business cycle test cases (2) | 4 |
| Scale | 5 |
| Performance | 5 |
| **Part B - System Test Execution and Reporting (Individual) /25** |  |
| Deployment | 1 |
| Run book updates | 1 |
| Results captured (spreadsheet) | 12 |
| Bugs correctly captured in spreadsheet | 8 |
| Ad hoc /exploratory tests | 3 |
| **Part C - Consolidating Bugs and Backlog Grooming – 20 - (Group / 15 Individual /5)** |  |
| Consolidated list of bugs(group) | 5 |
| Individual bug status (individual) | 5 |
| Backlog in AzureDevOps is groomed (group) | 10 |
| **Part D - Meeting the Customer – 30 - (Group/20) Individual (/10)** |  |
| Plan for the meeting | 5 |
| Meet with the customer | 5 |
| Generate minutes and send | 5 |
| Capture new and updated requests in Azure and list in assignment | 5 |
| Individual participation in customer meeting | 10 |
| **Part E - Sprint Planning – 20 - (Group /15 Individual /5)** |  |
| Decompose and size top priority for first sprint | 10 |
| Record and submit plan | 5 |
| Individual participation in sprint planning effort | 5 |
| **Part F - Reflection (Individual) /9** |  |
| Technical difficulties | 3 |
| Interpersonal or team dynamics issues | 3 |
| Top 3 concerns & mitigation | 3 |
|  |  |
| **Total** | 169 |