SECP3204: Software Engineering WBL

System Testing Documentation

UTM Student Feedback System

Version 1.0

17/06/2023

School of Computing, Faculty of Engineering

Prepared by: Limapuluh

Revision Page

a.   **Overview**

In this version of system description, we have completed the System Testing Description (STD) section. We will be listing out every test case necessary and also the test approach we have taken to test our system.

b.   **Target Audience**

The targeted audience for our system description are the students, lecturers of UTM, and CDex staff who take charge in handling the ePPP feedback system. Through this system description, they will understand how to conduct system testing and the testing outcomes.

c.   **Project Team Members**

Team Member	Task	Status
Lim Jing Yong	TC001, TC009	Complete
Soh Han Wei	TC002, TC004, TC006	Complete
Lee Soon Der	TC005	Complete
Marcus Joey Sayner	TC003, TC010	Complete
Yong Wern Jie	TC007, TC008	Complete

d. **Version Control History**

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Yong Wern Jie	Completed the solution proposal and produced Use Case Diagram (PR1.0)	11/05/2023
Version 2.0	Lim Jing Yong Soh Han Wei	Completed the SRS - Software Requirement Description (PR2.0)	28/05/2023
Version 3.0	Lee Soon Der	Completed the SDD - Software Design Description (PR3.0)	20/06/2023
Version 4.0	Marcus Joey Sayner	Completed the STD - Software Test Description (PR4.0)	23/06/2023

Table of Contents

Test Cases

8.1 TC001: Test <Authentication and Security Module> Subsystem: <Log In (UC001)>

This test contains the following test cases:

(a) TC001_01: Test <Login (SD001)>

8.1.1 TC001_01: Test <Login (SD001)>

Test Case ID	TC001_01	Test Case Description		Test the Login functionality in Student Feedback System			
Created By	Jing Yong	Reviewed By		Han Wei		Version	1.0
QA Tester's Log	None						
Tester's Name	Jing Yong	Date Tested		21/06/2023		Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:		S#	Test Data
1	Access to Internet Browser		1	username = Jyng14
2	Registered User Account		2	password = abc123

Test Scenario	Verify on loging in into the student feedback system

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Navigate to the main menu of Student Feedback System	Login screen should show	As expected	Pass
2	Input username and password	Credentials can be entered	As expected	Pass
3	Click Login	User is brought to menu based on user type	As expected	Pass

8.2 TC002: Test <Authentication and Security Module> Subsystem: <Reset Password (UC002)>

This test contains the following test cases:

(a) TC002_01: Test <Reset Password (SD002)>

8.2.1 TC002_01: Test <Reset Password (SD002)>

Test Case ID	TC002_01	Test Case Description		Test the reset password function in Student Feedback System			
Created By	Han Wei	Reviewed By		Jing Yong		Version	1.0
QA Tester's Log	None						
Tester's Name	Han Wei	Date Tested		22/06/2023		Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:		S#	Test Data
1	Access to Internet Browser		1	student_username = jingyong14
			2	student_email = Jyng14@graduate.utm.my
			3	student_new password = im@jingyong123
			4	lecturer_username = hanwei04
			5	lecturer_email = hanwei04@utm.my
			6	lecturer_new password = im@hanwei456
			7	admin_username = soonder15
			8	admin_email = soonder15@utm.my
			9	admin_new password = im@soonder789
			10	TAC code = 012345

Test Scenario	Verify on reseting password for student, lecturer and admin

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Navigate to the main menu of Student Feedback System	Login screen show	As expected	Pass
2	Input username	Input textarea function normally	As expected	Pass
3	Click on the "Forget Password?" button	User is brought to the enter email address interface	As expected	Pass
4	Enter user email address	Input textarea function normally	As expected	Pass
5	Enter TAC code received in email	User should receive TAC code through email and the input textarea function normally	As expected	Pass
6	Enter new password	Input textarea function normally	As expected	Pass
7	Reenter new password	Input textarea function normally	As expected	Pass
8	Click "Submit" button	User's old password should be replaced by new password	As expected	Pass

8.3 TC003: Test <Front-end Interface> Subsystem: <Give Feedback (UC004)>

This test contains the following test cases:

(a)   TC003_01: Test <Give Feedback (SD003)>

(b)   TC003_02: Test <Edit Feedback (SD004)>

8.3.1 TC003_01: Test <Give Feedback (SD003)>

Test Case ID	TC003_01	Test Case Description	Give Feedback - Submitting feedback with all questions answered		
Created By	Jing Yong	Reviewed By	Marcus Joey Sayner	Version	1.0
QA Tester's Log	None				
Tester's Name	Jing Yong	Date Tested	22/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:		S#	Test Data
1	Accessed to the Internet Browser		1	student_username = Jyng14
2	Registered Student Account		2	student_password = abc123
3	The student has chosen the lecturer in UC003		3	lecturerID = L001 (Han Wei)
4			4	classID = 35

Test Scenario	Student opening the feedback form and making a feedback to their lecturer.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	The student opens the feedback form.	The feedback form is displayed.	As expected	Pass
2	The student answers all the questions.	All questions are answered.	As expected	Pass
3	The student leaves additional comments.	Relevant comments are provided.	As expected	Pass
4	The student clicks on the "Submit" button.	The feedback is successfully submitted.	As expected	Pass
5	The system displays a success message.	The success message is displayed.	As expected	Pass

8.3.2 TC003_02: Test <Edit Feedback (SD004)>

Test Case ID	TC003_02	Test Case Description	Edit Feedback - Allows the student to view and edit their previously submitted feedback.		
Created By	Jing Yong	Reviewed By	Marcus Joey Sayner	Version	1.0
QA Tester's Log	None				
Tester's Name	Jing Yong	Date Tested	22/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:		S#	Test Data
1	Accessed to the Internet Browser		1	student_username = Jyng14
2	Registered User Account		2	student_password = abc123
3	The student has a valid account and login credentials.		3	
4	The student has previously submitted feedback.		4	

Test Scenario	Student log in the system and tries to edit the previous feedback.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	The student logs into the student feedback system.	The student is successfully logged in.	As expected	Pass
2	The student selects the "Edit Feedback" option.	The previously submitted feedback is displayed for editing.	As expected	Pass
3	The student modifies their answers to the questions.	The changes are applied to the feedback entry.	As expected	Pass
4	The student saves the updated feedback.	The feedback is successfully saved with the modifications.	As expected	Pass
5	The system displays a success message.	The success message is displayed.	As expected	Pass

8.4   TC004: Test <Front-end Interface> Subsystem: <View Feedback (UC005)>

This test contains the following test cases:

(a)   TC004_01: Test <View Feedback (SD005)>

8.4.1 TC004_01: Test <View Feedback (SD005)>

Test Case ID	TC004_01	Test Case Description		Test the View Feedback functionality in Student Feedback System		
Created By	Han Wei	Reviewed By		Soonder	Version	1.0
QA Tester's Log	None					
Tester's Name	Han Wei	Date Tested		22/6/2023	Test Case (Pass/Fail/Not)	Pass
S#	Prerequisites:			S#	Test Data	
1	Access to Internet Browser			1	username = hanwei04	
2	Login to Student Feedback System			2	password = def456	
3				3		
4				4		
Test Scenario	Verify on view feedback					

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "View Feedback" icon	Redirects you to the view Feedback page	As expected	Pass
2	Scrow and click on a student's feedback with feedbackID#001	Redirects you to the Feedback interface with the student's reply	As expected	Pass

8.5   TC005: Test <Back-end Interface> Subsystem: <Manage Feedback (UC006)>

This test contains the following test cases:

(a)   TC005_01: Test <Add Comment (SD006)>

(b)   TC006_01: Test <Remove Feedback (SD007)>

8.5.1 TC005_01: Test <Add Comment (SD006)>

Test Case ID		TC005_01	Test Case Description		Test the fucntion of adding comments to students'feedbacks		
Created By		Soon Der	Reviewed By		Han Wei	Version	1
QA Tester's Log		None					
Tester's Name		Soon Der	Date Tested		21/06/2023	Test Case (Pass/Fail/Not)	Pass
S#		Prerequisites:			S#	Test Data	
1		Access to Internet Browser			1	username = hanwei04	
2		Logged into system as Lecturer			2	password = def456	
3					3	feedbackID = 001	
4					4		
Test Scenario		Verify on adding comments to students'feedbacks					

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "View Feedback" icon	Redirects you to the view Feedback page	As expected	Pass
2	Scrow and click on a student's feedback with feedbackID#001	Redirects you to the Feedback interface with the student's reply	As expected	Pass
3	Click on the "Leave a reply?" button	Redirects you to the LeaveReply interface	As expected	Pass
4	Enter comments and click on the "Submit" button	Redirects you back to the view Feedback page	As expected	Pass

8.5.2 TC005_02: Test <Remove Feedback (SD007)>

Test Case ID	TC005_02	Test Case Description		Test to remove student's feedback from the feedback list		
Created By	Soon Der	Reviewed By		Han Wei	Version	1

QA Tester's Log	None					

Tester's Name	Soon Der	Date Tested		21/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Access to Internet Browser			1	username = soonder15	
2	Logged into the system as admin			2	password = ghi789	
3				3	feedbackID = 001	
4				4		

Test Scenario	Verify on delete student's feedback from the feedback list

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "Feedback List" icon	Redirects you to the FeedBack List page	As expected	Pass
2	Scrow and click on the feedback with feedbackID#001	Redirects you to the View Feedback page	As expected	Pass
3	Click on the "Delete" icon	A notification pops up and asks you to confirm whether remove the feedback or not	As expected	Pass
4	Click on the "Sure" button	A notification pops up and show that "You have successfully deleted the feedback#001	As expected	Pass
5	Click on the "Back" button	Redirects you to the FeedBack List page	As expected	Pass

8.6   TC006: Test <Back-end Interface> Subsystem: <Download Excel Feedback (UC007)>

This test contains the following test cases:

(a)   TC006_01: Test <Download Excel (SD008)>

8.6.1 TC006_01: Test <Download Excel (SD008)>

Test Case ID	TC006_01	Test Case Description		Test the Download Excel functionality in Student FeedBack System		
Created By	Han Wei	Reviewed By		Marcus	Version	1.0

QA Tester's Log	None					

Tester's Name	Han Wei	Date Tested		22/6/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Access to Internet Browser			1	username = hanwei04	
2	Login to Student Feedback System			2	password = def456	
3				3		
4				4		

Test Scenario	Verify on download excel

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "Feedback List" icon	Redirects you to the Feedback List page	As expected	Pass
2	Click on the "Download Feedback List" button	The button function normally and redirect the user to select the format of file	As expected	Pass
3	Select "Excel Format"	The feedback list can be generated in excel format	As expected	Pass
4	Click the "Confirm" button	The "Confirm" button function normally to allow the user download the excel file	As expected	Pass
5				

8.7 TC007: Test <Reporting Module> Subsystem: <Generate Report (UC010)>

This test contains the following test cases:

(a)   TC007_01: Test <Generate Report (SD009)>

8.7.1 TC007_01: Test <Generate Report (SD009)>

Test Case ID	TC007	Test Case Description		To verify the functionality of the Generate Report feature in the system		
Created By	Wern Jie	Reviewed By		Jing Yong	Version	1.0
QA Tester's Log	None					
Tester's Name	Wern Jie	Date Tested		22/6/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Access to internet browser			1	username = hanwei04	
2	The lecturer is logged into the system interface.			2	pw = def456	
3	The lecturer has the necessary permissions to generate reports.			3	Timerange = June2023	
4	The lecturer has access to student feedback data for the specified time range, course code, and questions.			4	CourseID = 01	
				5	QuestionID = 01	
				6	reportID = 001	

Test Scenario	Generate Report

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "Generate Report" button.	The system opens the report generation interface.	As expected	Pass
2	Choose the time range, course code, and select specific questions (multiple choices) to include in the report.	The system displays the selected criteria in the report generation interface.	As expected	Pass
3	Click the "Submit" button.	The system processes the selected criteria and generates the report.	As expected	Pass
4	Verify that the generated report is displayed in the system interface.	The system presents the generated report with the selected criteria.	As expected	Pass
5	If the option to generate another report is available, click on the "Generate Another Report" button.	The system clears the selected criteria and allows the user to specify new criteria for another report.	As expected	Pass
6	Verify that the system generates the report based on the specified criteria.	The system generates a report that includes a summary of the feedback data.	As expected	Pass
7	If the option to download the report is available, click on the "Download Report" button.	The system generates the report in PDF format and initiates the download.	As expected	Pass
8	Review the generated report.	The report contains the relevant data based on the selected criteria.	As expected	Pass
9	If applicable, verify that the downloaded report matches the displayed report.	The downloaded report matches the report displayed in the system interface.	As expected	Pass
10	If the user chooses to generate another report, repeat steps 2 to 8 with different criteria.	The system allows the user to generate multiple reports with different criteria.	As expected	Pass

8.8 TC008: Test <Reporting Module> Subsystem: <View Report (UC011)>

This test contains the following test cases:

(a)   TC008_01: Test <View Report (SD010)>

8.8.1 TC008_01: Test <View Report (SD010)>

Test Case ID		TC008		Test Case Description		To verify that the system allows the administrator to view reports, either by selecting from a list of existing reports or specifying criteria for a specific report.			
Created By		Wern Jie		Reviewed By		Jing Yong		Version	1.0
QA Tester's Log		None							
Tester's Name		Wern Jie		Date Tested		23/6/2023		Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data
1	Access to internet browser			1	username = soonder15
2	The user must have valid administrator credentials to log into the system.			2	password = ghi789
3	The necessary permissions must be assigned to the user to view reports.			3	ReportID = 001
4	At least one report must be available in the system database for viewing.			4	courseID = 01
				5	Timerange = June2023

Test Scenario	Viewing a Report

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the "View Report" option.	The system presents options for viewing reports.	As expected	Pass
2	Verify that the system presents a list of available and existing reports.	A list of reports is displayed, showing their names or identifiers.	As expected	Pass
3	Select a report from the list by clicking on its name.	The selected report is highlighted or marked as chosen.	As expected	Pass
4	Verify that the system retrieves the requested report from the database.	The system fetches the report data from the database based on the selected report.	As expected	Pass
5	Verify that the report is displayed on the screen.	The report content is shown in the user interface, formatted and structured properly.	As expected	Pass
6	Review the report to ensure that it contains the expected information.	The report includes the relevant data and provides the desired information.	As expected	Pass
7	If desired, choose the option to download the report in PDF format.	The system provides a download link or option to save the report as a PDF file.	As expected	Pass
8	Verify that the report is downloaded successfully and can be accessed as a PDF file.	The report is downloaded and saved in PDF format without any issues.	As expected	Pass
9	Close the report interface.	The report interface is closed without any errors or unexpected behavior.	As expected	Pass
10	Repeat the above steps with different reports or by specifying criteria for a specific report.	The system allows the administrator to view multiple reports by selecting different options or specifying criteria.	As expected	Pass

8.9 TC009: Test <Integration Module> Subsystem: <Notification Pop-up (UC012)>

This test contains the following test cases:

(a)   TC009_01: Test <Student Notification (SD011)>

(b)   TC009_02: Test <Lecturer Notification (SD012)>

(c)   TC009_03: Test <Administrator Notification (SD013)>

8.9.1 TC009_01: Test <Student Notification (SD011)>

Test Case ID	TC009_03		Test Case Description		Test the notification function for admin				
Created By	Soon Der		Reviewed By		Marcus		Version		1.0
QA Tester's Log	None								
Tester's Name	Soon Der		Date Tested		22/06/2023		Test Case (Pass/Fail/Not)	Pass	
S#	Prerequisites:				S#	Test Data			
1	Access to Internet Browser				1	username = soonder15			
2	Logged into system as Admin				2	password = hij789			
3					3	lectID = L001 (Han Wei)			
4					4	feedbackID = 001			
5					5	studentID = S001 (Jing yong)			
6					6	courseID = 001			
Test Scenario	Verify on checking a new notification from student or lecturer								
Step #	Step Details		Expected Results		Actual Results		Pass / Fail / Not Executed / Suspended		
1	Click on the bell icon		Redirect you to the notification page		As expected		Pass		
2	Select "Student" or "Lecturer"		Show a list of feedbacks submitted by students or a list of reports generated by lecturers		As expected		Pass		
3	Click on "View Feedback" of course 001 in student notification or "View Report" of course 001 in lecturer notification		Redirect you to the feedback interface or the report interface		As expected		Pass		

Test Case ID	TC009_01		Test Case Description		Test the notification function for user of type student				
Created By	Jing Yong		Reviewed By		Han Wei		Version		1.0
QA Tester's Log	None								
Tester's Name	Jing Yong		Date Tested		21/06/2023		Test Case (Pass/Fail/Not)	Pass	
S#	Prerequisites:				S#	Test Data			
1	Access to Internet Browser				1	username = Jingyong14			
2	Logged into system as Student				2	password = abc123			
3					3	lectID = L001 (Han Wei)			
4					4	courseID = 01			
5					5	feedbackID = 001			
Test Scenario	Verify on checking a new notification as a student								
Step #	Step Details		Expected Results		Actual Results		Pass / Fail / Not Executed / Suspended		
1	Click on the bell icon		Redirects you to the notification page		As expected		Pass		
2	Click on the "View Reply" by lecturer "Han Wei"		Redirects you to the feedback interface with the reply pop-out open		As expected		Pass		

8.9.2 TC009_02: Test <Lecturer Notification (SD012)>

Test Case ID	TC009_02		Test Case Description		Test the notification function for lecturer				
Created By	Han Wei		Reviewed By		Soon Der		Version		
QA Tester's Log	None								
Tester's Name	Han Wei		Date Tested		22/06/2023		Test Case (Oass/Fail/Not)		
S#	Prerequisites:				S#	Test Data			
1	Access to Internet Browser				1	username = hanwei20			
2	Logged into system as Lecturer				2	pw = def456			
3					3	feedbackID = 001			
4					4	courseID = 01			
Test Scenario	Verify on checking a new notification as lecturer								
Step #	Step Details		Expected Results		Actual Results		Pass / Fail / Not Executed / Suspended		
1	Click on the bell icon		Redirects you to the notification page		As expected		Pass		
2	Click on the "View Feedback" of course 001		Redirects you to the new feedback interface submitted by student		As expected		Pass		

8.9.3 TC009_03: Test <Administrator Notification (SD013)>

	S#						
Test Case ID		TC009_03	Test Case Description		Test the notification function for admin		
Created By		Soon Der	Reviewed By		Marcus	Version	1.0
QA Tester's Log		None					
Tester's Name		Soon Der	Date Tested		22/06/2023	Test Case (Pass/Fail/Not)	Pass
	S#	Prerequisites:			S#	Test Data	
	1	Access to Internet Browser			1	username = soonder15	
	2	Logged into system as Admin			2	password = hij789	
	3				3	lectID = L001 (Han Wei)	
	4				4	feedbackID = 001	
	5				5	studentID = S001 (Jing yong)	
	6				6	courseID = 001	
Test Scenario		Verify on checking a new notification from student or lecturer					

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Click on the bell icon	Redirect you to the notification page	As expected	Pass
2	Select "Student" or "Lecturer"	Show a list of feedbacks submitted by students or a list of reports generated by lecturers	As expected	Pass
3	Click on "View Feedback" of course 001 in student notification or "View Report" of course 001 in lecturer notification	Redirect you to the feedback interface or the report interface	As expected	Pass

8.10 TC0010: Test <Integration Module> Subsystem: <Notification Pop-up (UC012)>

This test contains the following test cases:

(a)   TC010_01: Test <Edit Student Data (SD014)>

(b)   TC010_02: Test <Delete Student Data (SD015)>

(c)   TC010_03: Test <Edit Lecturer Data (SD016)>

(d)   TC010_04: Test <Delete Lecturer Data (SD017)>

8.10.1 TC010_01: Test <Edit Student Data (SD014)>

	S#				S#		
Test Case ID		TC010_01	Test Case Description		Admin can modify the data of a specific student within the system.		
Created By		Soon Der	Reviewed By		Jing Yong	Version	1.0
QA Tester's Log		None					
Tester's Name		Soon Der	Date Tested		22/06/2023	Test Case (Pass/Fail/Not)	Pass
	S#	Prerequisites:			S#	Test Data	
	1	Accessed to the Internet Browser			1	admin_email = soonder15	
	2	The user is logged into the system as an admin.			2	admin_password = ghi789	
	3	The system displays the student information form.			3	studentID = S001 (Jing Yong)	
	4				4	old_courseID = 01	
	5				5	new_courseID = 02	
Test Scenario		Changing the Student's CourseID from 01 to 02.					

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Search for the student with ID "S001" in the system.	The student with ID "SD014" is found.	As expected	Pass
2	Select the student "S001" from the search results.	The student "SD014" is selected.	As expected	Pass
3	Update the student's information with the test data.	The student's information is updated.	As expected	Pass
4	Save the changes to the student's information.	The changes to the student's information are saved.	As expected	Pass
5	Verify that the student's information has been updated.	The student's information is displayed correctly with the updated data.	As expected	Pass

8.10.2 TC010_02: Test <Delete Student Data (SD015)>

Test Case ID	TC010_02	Test Case Description		Admin can delete the data of a specific student from the system.		
Created By	Soon Der	Reviewed By		Jing Yong	Version	1.0
QA Tester's Log	None					
Tester's Name	Soon Der	Date Tested		22/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Accessed to the Internet Browser			1	admin_email = soonder15	
2	The user is logged into the system as an admin.			2	admin_password = ghi789	
3	The system displays the student information form.			3	studentID = S001 (Jing Yong)	

Test Scenario	Deleting the Student ID (S001)

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Search for the student with ID "S001" in the system.	The student with ID "S001" is found.	As expected	Pass
2	Select the student "S001" from the search results.	The student "S001" is selected.	As expected	Pass
3	Click on the "Delete" button to remove the student's data.	A confirmation dialog is displayed.	As expected	Pass
4	Confirm the deletion of the student's data.	The student's data is deleted from the system.	As expected	Pass
5	Verify that the student's data has been deleted.	The student with ID "S001" is no longer found in the system.	As expected	Pass

8.10.3 TC010_03: Test <Edit Lecturer Data (SD016)>

Test Case ID	TC010_03	Test Case Description		Admin can modify the data of a specific lecturer within the system.		
Created By	Soon Der	Reviewed By		Jing Yong	Version	1.0
QA Tester's Log	None					
Tester's Name	Soon Der	Date Tested		22/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Accessed to the Internet Browser			1	admin_email = soonder15	
2	The user is logged into the system as an admin.			2	admin_password = ghi789	
3	The system displays the lecturer information form.			3	lecturerID = L001 (Han Wei)	
4				4	old_faculty = "Faculty of Engineering"	
5				5	new_faculty = "Faculty of Computing"	

Test Scenario	Changing the Lecturer's faculty from Engineering to Computing. (L001)

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Search for the lecturer with ID "S001" in the system.	The lecturer with ID "S001" is found.	As expected	Pass
2	Select the lecturer "S001" from the search results.	The lecturer "S001" is selected.	As expected	Pass
3	Update the lecturer's information with the test data.	The lecturer's information is updated.	As expected	Pass
4	Save the changes to the lecturer's information.	The changes to the lecturer's information are saved.	As expected	Pass
5	Verify that the lecturer's information has been updated.	The lecturer's information is displayed correctly with the updated data.	As expected	Pass

8.10.4 TC010_04: Test <Delete Lecturer Data (SD017)>

Test Case ID	TC010_03	Test Case Description		Admin can delete the data of a specific lecturer from the system.		
Created By	Soon Der	Reviewed By		Jing Yong	Version	1.0
QA Tester's Log	None					
Tester's Name	Soon Der	Date Tested		22/06/2023	Test Case (Pass/Fail/Not)	Pass

S#	Prerequisites:			S#	Test Data	
1	Accessed to the Internet Browser			1	admin_email = soonder15	
2	The user is logged into the system as an admin.			2	admin_password = ghi789	
3	The system displays the lecturer information form.			3	lecturerID = L001 (Han Wei)	

Test Scenario	Deleting the Lecturer's ID (L001)

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not Executed / Suspended
1	Search for the lecturer with ID "SD017" in the system.	The lecturer with ID "SD017" is found.		
2	Select the lecturer "SD017" from the search results.	The lecturer "SD017" is selected.		
3	Click on the "Delete" button to remove the lecturer's data.	A confirmation dialog is displayed.		
4	Confirm the deletion of the lecturer's data.	The lecturer's data is deleted from the system.		
5	Verify that the lecturer's data has been deleted.	The lecturer with ID "S001" is no longer found in the system.		

9. **Test Approach Analysis**

The test approach includes the use of a combination of manual testing techniques and automated testing tools. We employed a black box testing strategy, focusing on equivalence partitioning (EP) and boundary value analysis (BVA) techniques. This approach allowed us to ensure comprehensive test coverage by considering different input scenarios and boundary conditions.

For manual testing, we carefully designed test cases based on the software requirements and functionality. Each test case was assigned a unique test case ID, which can be found in the respective sub-sections of the STD documentation. We analyzed the test steps within the test cases to identify inputs that require equivalence partitioning and boundary value analysis.

Equivalence partitioning (EP) was applied at specific steps in the test cases where input data had different equivalence classes. By selecting representative values from each equivalence class, we aimed to cover a wide range of scenarios without exhaustively testing every possible input value. This technique helped us ensure adequate test coverage while optimizing testing efforts.

Boundary value analysis (BVA) was used to test the boundaries of input ranges, as they often represent critical points in the software. We applied BVA at steps in the test cases where input data had defined boundaries. By testing the values at the lower and upper boundaries, as well as just inside and outside those boundaries, we aimed to validate how the software handles boundary conditions.

To support our testing efforts, we utilized automated testing tools that enabled us to execute test scripts and perform repetitive or complex tests efficiently. These tools helped streamline the testing process, improve overall test coverage, and ensure consistency in test execution.

The success or failure of our test approach will be evaluated based on several factors, including the coverage of test cases, identification of defects, and the overall reliability and performance of the software. We will analyze the test results to assess whether the software meets the specified requirements and functions as expected. Any deviations or failures will be reported, and appropriate actions will be taken to address them in subsequent testing cycles.

By employing a combination of manual testing techniques, equivalence partitioning, and boundary value analysis, we aimed to provide a systematic and efficient test approach that helps uncover potential issues, ensure software quality, and deliver a reliable product to the end-users.

9.1 **TC001: Test <Authentication and Security Module> Subsystem: <Log In (UC001)>**

9.1.1 TC001_01: Test <Login (SD001)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC001_01	2	Username	less than 1 characters	jingyong14	more than 30 characters
TC001_01	2	Password	less than 1 characters	abc123	more than 30 characters

● **Boolean / Set Value**

NO INPUT DATA

9.2 **TC002: Test <Authentication and Security Module> Subsystem: <Reset Password (UC002)>**

9.2.1 TC002_01: Test <Reset Password (SD002)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC002_01	2	Username	less than 1 characters	jingyong14 hanwei04 soonder15	more than 30 characters
TC002_01	4	Email address	less than 1 characters	Jyng14@graduate.utm.my hanwei04@utm.my soonder15@utm.my	more than 30 characters
TC002_01	5	TAC code	less than 1 characters	012345	more than 6 characters

Test Case ID	Step#	Input data	Invalid		Valid	Invalid
TC002_01	6	Password	less than 1 characters	im@jingyong123 im@hanwei456 im@soonder789		more than 6 characters
TC002_01	7	Password	less than 1 characters	im@jingyong123 im@hanwei456 im@soonder789		more than 6 characters

- **Boolean / Set Value**

NO INPUT DATA

9.3 TC003: Test <Front-end Interface> Subsystem: <Give Feedback (UC004)>

9.3.1 TC003_01: Test <Give Feedback (SD004)>

- **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC003_01	Initial	lecturer_FirstName	less than 1 characters	Noorfa	more than 30 characters
TC003_01	Initial	courseID	less than 1 characters	01	more than 30 characters

- **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC003_01	3	comments	"The lecturer does a good job!".	more than 50 characters and the input must be string

9.3.2 TC003_02: Test <Edit Feedback (SD005)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC003_01	Initial	lecturer_FirstName	less than 1 characters	Noorfa	more than 30 characters
TC003_01	Initial	courseID	less than 1 characters	01	more than 30 characters

● **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC003_01	3	comments	"The lecturer does a good job!".	more than 50 characters and the input must be string

9.4 **TC004: Test <Front-end Interface> Subsystem: <View Feedback (UC005)>**

9.4.1 TC004_01: Test <View Feedback (SD005)>

Range Value Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC004_01	Initial	Username	less than 1 characters	hanwei04	more than 30 characters
TC004_01	Initial	Password	less than 1 characters	def456	more than 30 characters

● **Boolean / Set Value**

NO INPUT DATA

9.5 TC005: Test <Back-end Interface> Subsystem: <Manage Feedback (UC006)>

9.5.1 TC005_01: Test <Add Comment (SD006)>

Range Value Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC008_01	1	Username	less than 1 characters	soonder15	more than 30 characters
TC008_01	2	Password	less than 1 characters	ghi789	more than 30 characters
TC005_01	3	FeedbackID	less than 1 characters	#001	more than 30 characters

● **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC005_01	3	comments	"Noted, I will try to improve it!".	more than 50 characters and the input must be string

9.5.1 TC005_02: Test <Remove Feedback (SD007)>
● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC005_02	1	Username	less than 1 characters	soonder15	more than 30 characters
TC005_02	2	Password	less than 1 characters	ghi789	more than 30 characters
TC005_02	3	FeedbackID	less than 1 characters	#001	more than 30 characters

● **Boolean / Set Value**

NO INPUT DATA

9.6 **TC006: Test <Back-end Interface> Subsystem: <Download Excel Feedback (UC007)>**

9.6.1 TC006_01: Test <Download Excel Feedback (SD008)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC006_01	Initial	Username	less than 1 characters	hanwei04	more than 30 characters
TC006_01	Initial	Password	less than 1 characters	def456	more than 30 characters

● **Boolean / Set Value**

NO INPUT DATA

9.7 **TC007: Test <Reporting Module> Subsystem: <Generate Report (UC010)>**

9.7.1 TC007_01: Test <Generate Report (SD009)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC007_01	1	Username	less than 1 characters	hanwei04	more than 30 characters
TC007_01	2	Password	less than 1 characters	def456	more than 30 characters
TC007_01	3	Timerange	InvalidTimerange	June2023	InvalidTimerange

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC007_01	4	CourseID	00	01	99
TC007_01	5	QuestionID	00	01	99
TC007_01	6	reportID	000	001	999

● **Boolean / Set Value**

NO INPUT DATA

9.8 **TC008: Test <Reporting Module> Subsystem: <View Report (UC011)>**

9.8.1 TC008_01: Test <View Report (SD010)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC008_01	1	Username	less than 1 characters	soonder15	more than 30 characters
TC008_01	2	Password	less than 1 characters	ghi789	more than 30 characters
TC008_01	3	ReportID	000	001	999
TC008_01	4	CourseID	00	01	99
TC008_01	5	Timerange	InvalidTimeRange	June2023	InvalidTimeRange

● **Boolean / Set Value**

NO INPUT DATA

9.9 TC009: Test<Integration Module> Subsystem: <Notification Pop-up (UC012)>

9.9.1 TC009_01: Test <Student Notification (SD011)>

● **Range Value**

NO INPUT DATA

● **Boolean / Set Value**

NO INPUT DATA

9.9.2 TC009_02: Test <Lecturer Notification (SD012)>

● **Range Value**

NO INPUT DATA

● **Boolean / Set Value**

NO INPUT DATA

9.9.3 TC009_03: Test <Administrator Notification (SD013)>

● **Range Value**

NO INPUT DATA

● **Boolean / Set Value**

NO INPUT DATA

9.10 TC010: Test<Integration Module> Subsystem: <Manage User (UC013)>

9.10.1 TC010_01: Test <Edit Student Data (SD014)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC010_01	2	student_username	less than 2 characters	jyng14	more than 30 characters

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC010_01	2	student_First Name	less than 2 characters	Jing	more than 30 characters
TC010_01	2	student_LastName	less than 2 characters	Yong	more than 30 characters
TC010_01	2	StudentID	less than 2 characters	S001	more than 4 characters
TC010_01	2	student_email	less than 2 characters and not valid email format	jingyong14@graduate.utm.my	more than 50 characters

- **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC010_01	3	student_contact	01188347293	Input is not in numbers
TC010_01	3	Age	20	Input is not in numbers
TC010_01	3	CGPA	4.0	Input must be in decimal and below 4.0

9.10.2 TC010_02: Test <Delete Student Data (SD015)>

- **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC010_02	2	student_username	less than 2 characters	jyng14	more than 30 characters
TC010_02	2	student_First Name	less than 2 characters	Jing	more than 30 characters

Test Case ID	Step#	Input data			
TC010_02	2	student_LastName	less than 2 characters	Yong	more than 30 characters
TC010_02	2	student_email	less than 2 characters and not valid email format	jingyong14@graduate.utm.my	more than 50 characters

● **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC010_02	3	StudentID	S001	more than 4 characters

9.10.3 TC010_03: Test <Edit Lecturer Data (SD016)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC010_03	2	lecturer_username	less than 2 characters	hanwei04	more than 30 characters
TC010_03	2	lecturer_FirstName	less than 2 characters	Han	more than 30 characters
TC010_03	2	lecturer_LastName	less than 2 characters	Wei	more than 30 characters
TC010_03	2	LecturerID	less than 2 characters	L001	more than 4 characters
TC010_03	2	lecturer_email	less than 2 characters and not valid email format	hanwei04@graduate.utm.my	more than 50 characters

● **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC010_03	3	lecturer_contact	01188347293	Input is not in numbers
TC010_03	3	Age	28	Input is not in numbers
TC010_03	3	Faculty	Computing	Input must be in a word only (ec. Computing, Engineering)

9.10.4 TC0010_04: Test <Delete Lecturer Data (SD017)>

● **Range Value**

Test Case ID	Step#	Input data	Invalid	Valid	Invalid
TC010_04	2	lecturer_username	less than 2 characters	hanwei04	more than 30 characters
TC010_04	2	lecturer_FirstName	less than 2 characters	Han	more than 30 characters
TC010_04	2	lecturer_LastName	less than 2 characters	Wei	more than 30 characters
TC010_04	2	lecturer_email	less than 2 characters and not valid email format	hanwei04@graduate.utm.my	more than 50 characters

- **Boolean / Set Value**

Test Case ID	Step#	Input data	Valid	Invalid
TC010_04	3	LecturerID	L001	more than 4 characters

Appendix A: Traceability Matrix

Test Case ID	Use Case ID/ Sequence Diagram ID	Package ID
TC001 for <Authentication and Security Module> Subsystem ● TC001_01	UC001 ● SD001	P001
TC002 for <Authentication and Security Module> Subsystem ● TC002_01	UC002 ● SD002	P001
TC003 for <Front-end Interface> Subsystem ● TC003_01 ● TC003_02	UC004 ● SD003 ● SD004	P001
TC004 for <Front-end Interface> Subsystem ● TC004_01	UC005 ● SD005	P002
TC005 for <Back-end Interface> Subsystem ● TC005_01 ● TC005_02	UC006 ● SD006 ● SD007	P002
TC006 for <Back-end Interface> Subsystem ● TC006_01	UC007 ● SD008	P003

TC007 for <Reporting Module> Subsystem ● TC007_01	UC010 ● SD009	P004
TC008 for <Reporting Module> Subsystem ● TC008_01	UC011 ● SD010	P004
TC009 for <Integration Module> Subsystem ● TC009_01 ● TC009_02 ● TC009_03	UC012 ● SD011 ● SD012 ● SD013	P005
TC010 for <Integration Module> Subsystem ● TC010_01 ● TC010_02 ● TC010_03 ● TC010_04	UC013 ● SD014 ● SD015 ● SD016 ● SD017	P005