

**COMP 6030 : Software Engineering**

**Coursework – Individual Part**

**Group : 15**

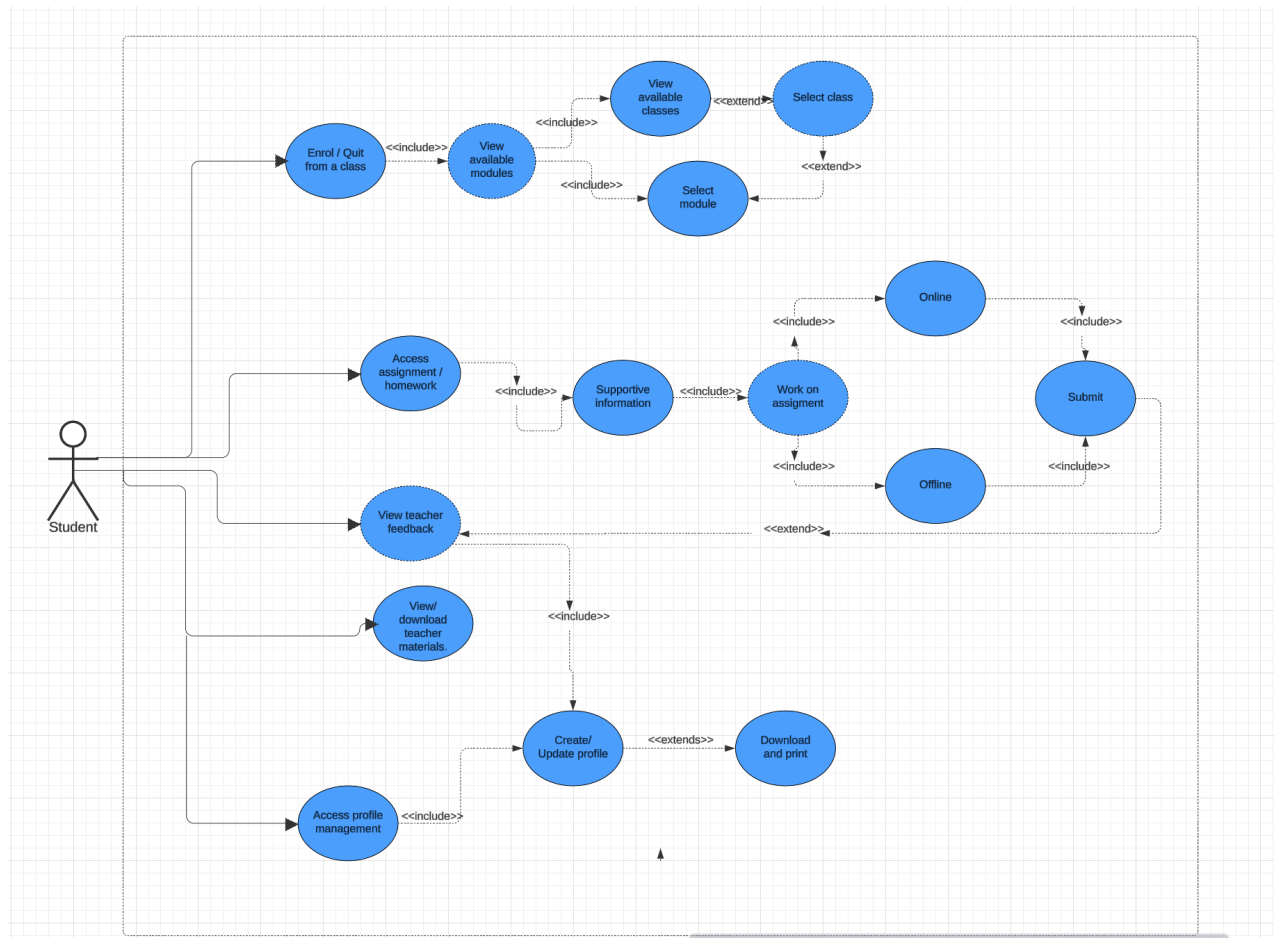
**Student Number : 19205073**

**Subsystem : Cla-Clo Student**

**Task1 - Specification and modelling software functional requirements (20 marks)**

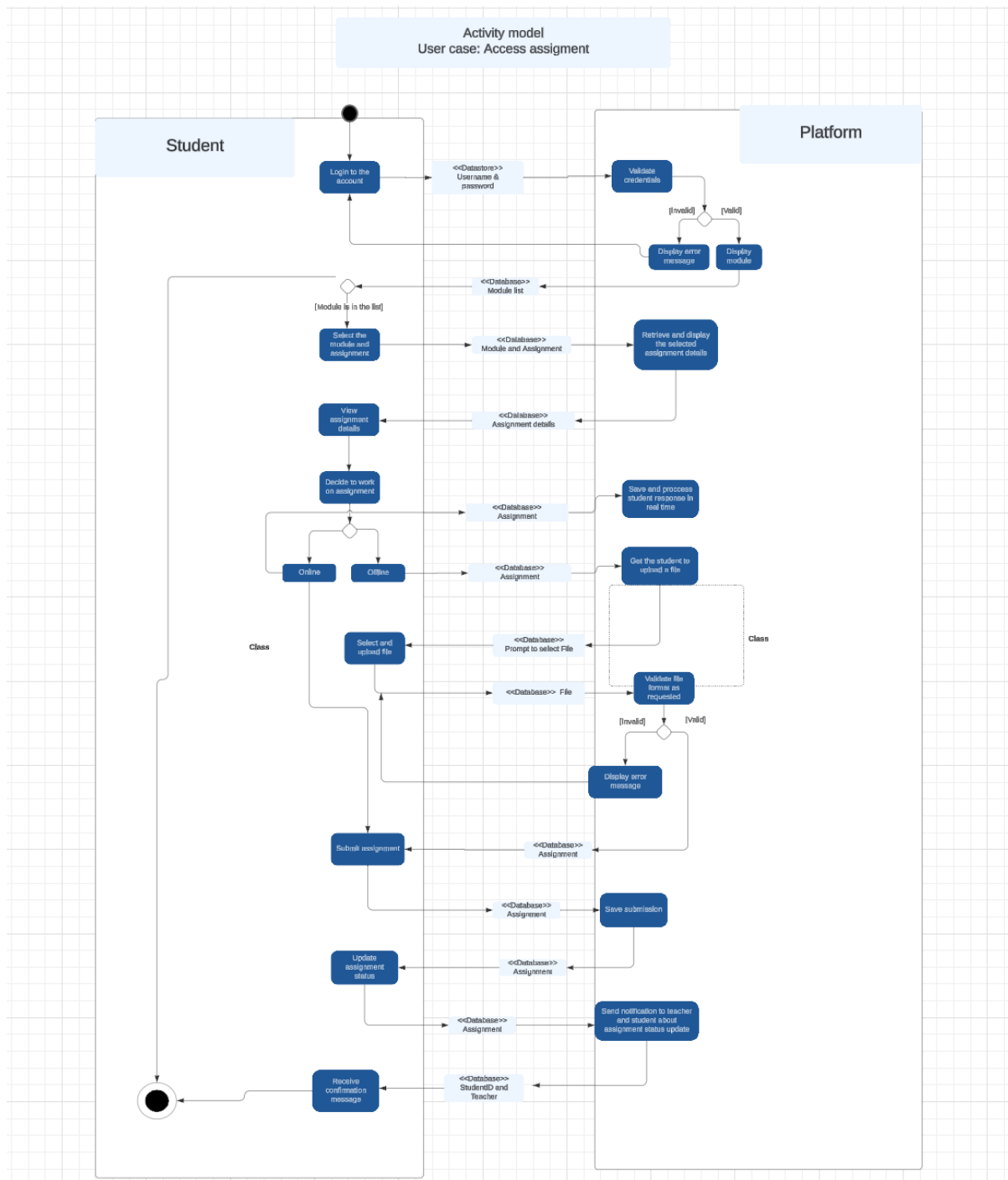
**A-Use Case mode**

Link: [Lucidchart](#)



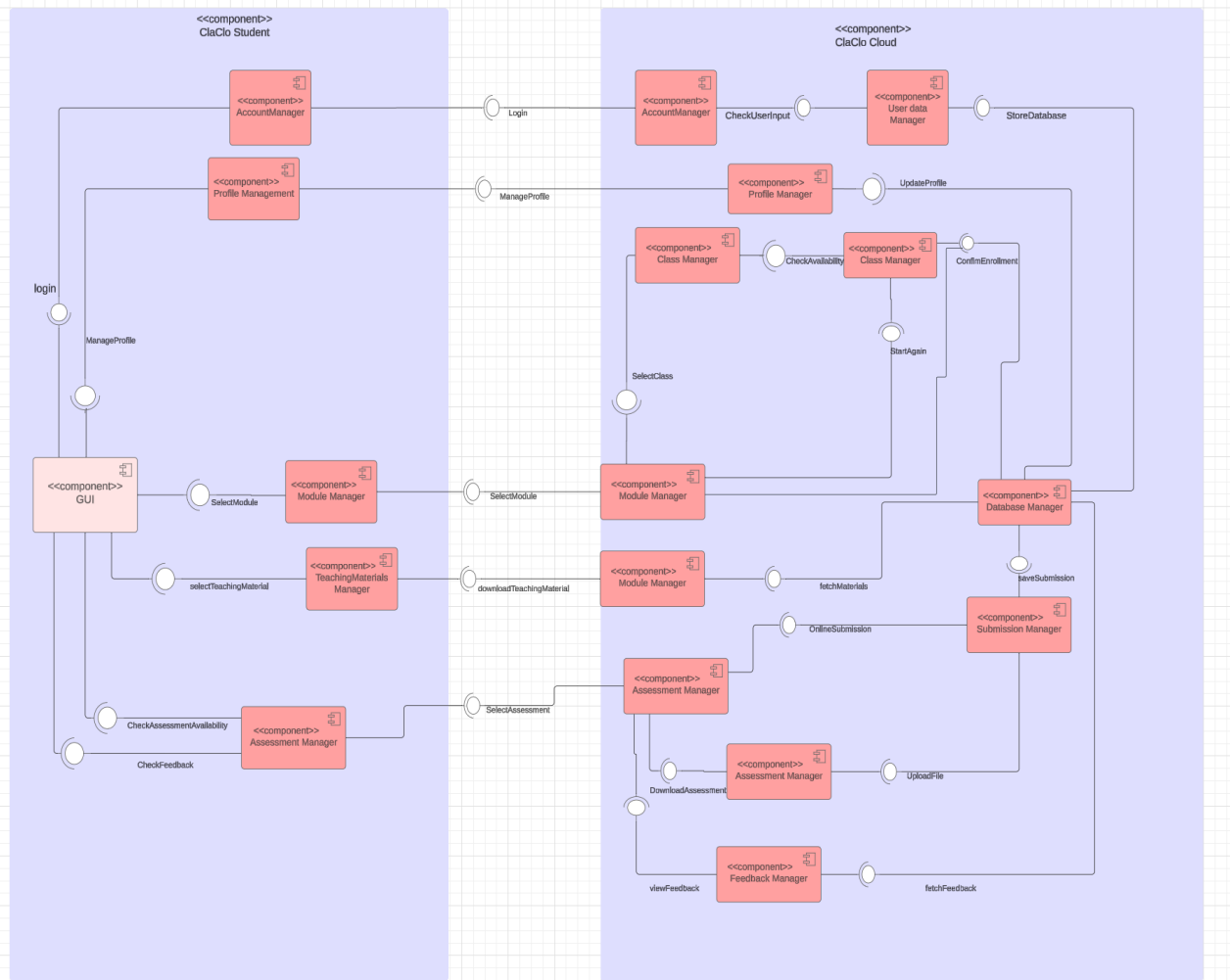
B-Activity model

Link: [Lucidchart](#)



## TASK 2- Architecture of the Student subsystem

Link: [Lucidchart](#)



## Specification of components

<b>Component</b>	AccountManager
<b>Description</b>	This microservice allows the student to login to their account
<b>Type</b>	microservice

<b>Required interface</b>	UserDataManager
<b>Provided interface</b>	CheckUserInput

<b>Component</b>	ProfileManager
<b>Description</b>	This microservice allows the students to view, update and download their academic profile for printing or sharing
<b>Type</b>	microservice
<b>Required interface</b>	databaseManager
<b>Provided interface</b>	manageProfile, updateProfile

<b>Component</b>	Module Manager
<b>Description</b>	This microservice allows the student to enroll to a module/class
<b>Type</b>	microservice
<b>Required interface</b>	DatabaseManager
<b>Provided interface</b>	CheckAvailability,ConfirmEnrollment

<b>Component</b>	Class Manager
<b>Description</b>	This microservice allows the student to enroll into a specific class within a module if many classes are available}, and access all of that class related activities
<b>Type</b>	microservice
<b>Required interface</b>	DatabaseManager
<b>Provided interface</b>	CheckAvailability,ConfirmEnrollment

<b>Component</b>	Teaching Materials Manager
------------------	----------------------------

<b>Description</b>	Allows the student to view or download or view the teaching materials that were uploaded by the teachers.
<b>Type</b>	microservice
<b>Required interface</b>	DatabaseManager
<b>Provided interface</b>	downloadTeachingMaterial, fetchMaterial

<b>Component</b>	Assessment Manager
<b>Description</b>	Allows the student to view or download the assignments and their feedback after grading
<b>Type</b>	microservice
<b>Required interface</b>	SubmissionManager, Feedback Manager
<b>Provided interface</b>	CheckAssessmentAvailability, CheckFeedback, DownloadAssignment

<b>Component</b>	Submission Manager
<b>Description</b>	Handles the submissions of the assignments
<b>Type</b>	microservice
<b>Required interface</b>	DatabaseManager
<b>Provided interface</b>	SaveSubmission

<b>Component</b>	Feedback Manager
<b>Description</b>	Handles the feedback of the submitted assignments
<b>Type</b>	microservice
<b>Required interface</b>	SubmissionManager
<b>Provided interface</b>	fetchFeedback viewFeedback

### Specifications of interfaces

Name	CheckUserInput
------	----------------

Provider	Account Manager	
Operation name: login	Signature	checkUserInput(userID: String, password: String): Boolean
	Function	Checks student credentials and returns true or false if they match with the database saved details
Name	updateProfile	
Provider	Profile Manager	
Operation name: updateProfile	Signature	updateProfile(studentID: String, profileData: JSON): Boolean
	Function	Allows the student to update their profile based on recent changes related to academic studies

Name	SelectModule	
Provider	Module Manager	
Operation name: selectClass	Signature	selectClass(studentID:String, classID:String)
	Function	Allows the student to select one of the available classes on that specific module
Operation name: chooseAssignment	Signature	chooseAssignment(classID:String, assignmentID:String)
	Function	Allows the student to select any of the available assignments
Name	SelectAssignment	
Provider	Assignment Manager	
Operation name: submitOnlineWork	Signature	submitOnlineWork(studentID:string, assignmentID:string, savedFiles:JSON)
	Function	Allows the student to submit their online work straight away
Operation name: getAssignment	Signature	getAssignment(classID: String): List<Assignment>
	Function	Gets all the available assignments for that specific class and displays them as a list for the student

Name	uploadFile	
Provider	FileManager	
Operation name: submitOfflineWork	Signature	submitOfflineWork(assignmentID: String, file: File): Boolean
	Function	Allows the student to submit their offline work

Operation name: validateSubmission	Signature	validateSubmission(file: File): Boolean
	Function	Checks if the submission format is right

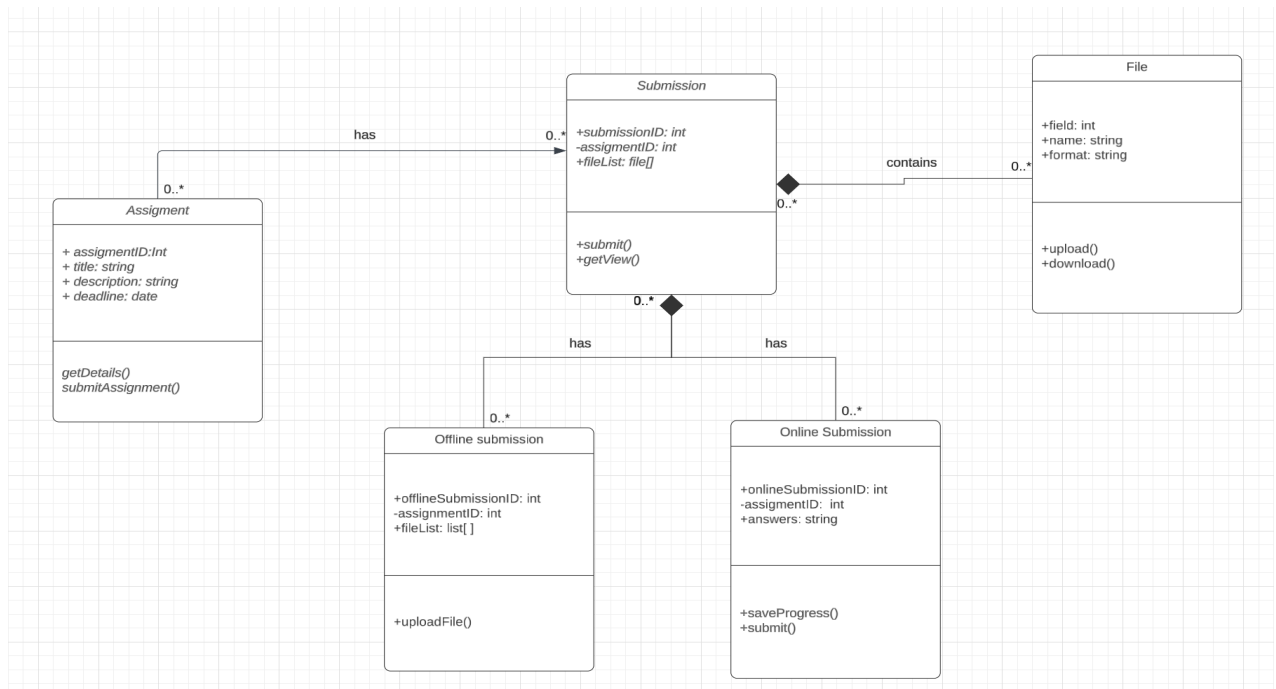
Name	selectTeachingMaterials	
Provider	TeachingMaterial Manager	
Operation name: fetchMaterials	Signature	fetchMaterials(moduleID: String): List<Material>
	Function	Allows the student to see the teaching materials
Operation name: downloadMaterials	Signature	downloadMaterials(materialID: String): File
	Function	Allows the student to download the available teaching materials

### Task3

#### A-Structural model

Model for submitting assessments

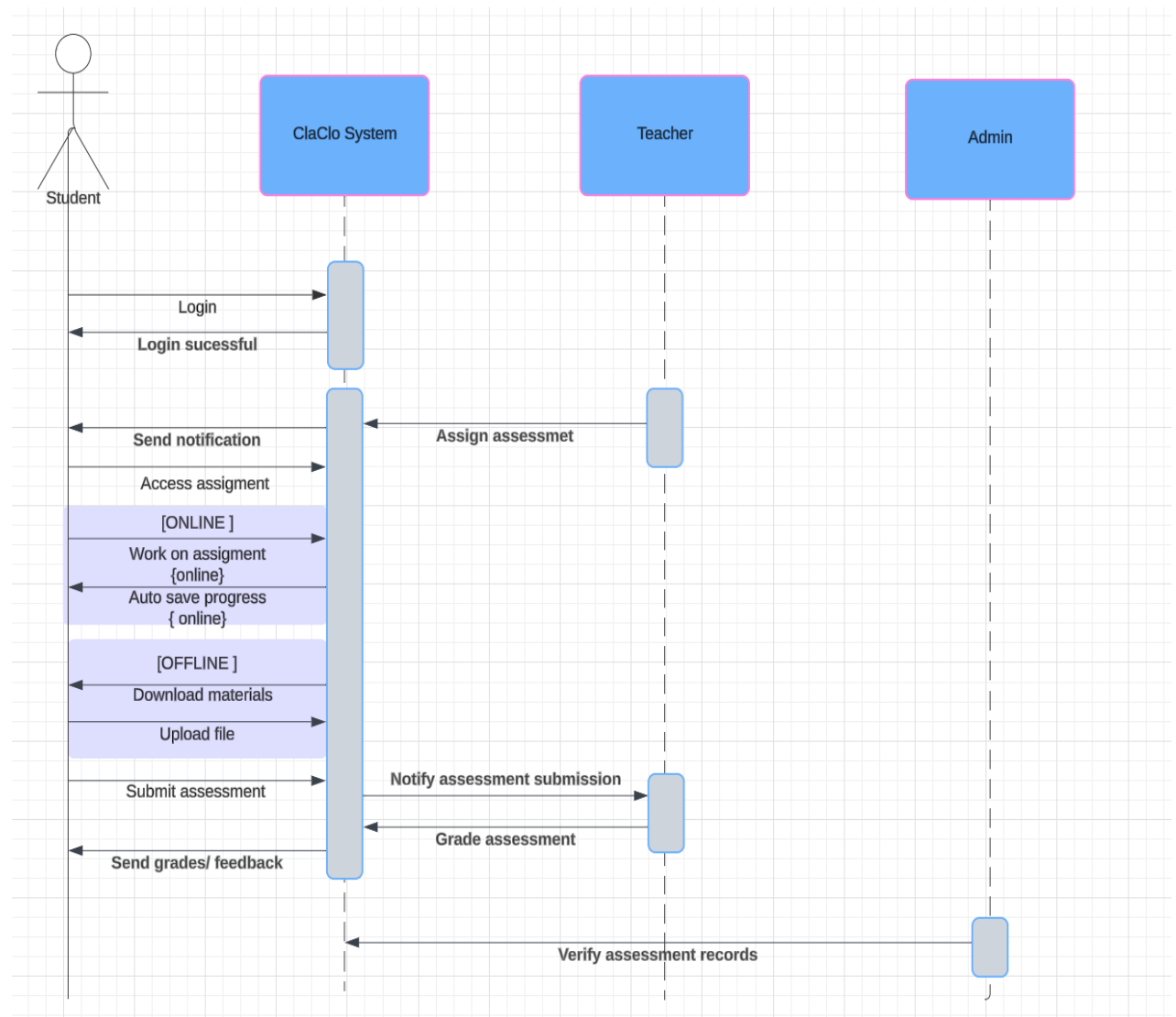
Link: [Lucidchart](#)





## B-Behaviour model

Link: [Lucidchart](#)



## Task4 - Software Testing

### A-Unit test plan

#### Class: Student

Test Case	Methods and Parameters	Expected Output
Access assignment (successfully)	accessAssignment(studentID, assignmentID)	Displays assignment with its contents successfully
Access assignment (unsuccessfully)	accessAssignment(studentID, assignmentID)	Assignment is not available to this student. Displays message saying "No new assignment assigned"
Submit online assignment (valid)	submitAssignment(studentID, assignmentID, Files)	Assignment submitted successfully.
Submit online assignment (invalid)	submitAssignment(studentID, assignmentID, Files)	Error message: "Invalid file! Please finish the whole assignment before submission online"
Save online work progress	saveProgress(studentID, assignmentID, progressData)	Progress saved successfully
Download assignment materials	downloadMaterials(studentID, assignmentID)	Materials downloaded successfully
Submit offline assignment (valid)	submitOfflineAssignment(studentID, assignmentID, uploadedFile)	Successful submission
Submit offline assignment (invalid)	submitOfflineAssignment(studentID, assignmentID, uploadedFile)	Unsuccessful submission, "Please upload files"

## B- System Test plan

**Use Case:** Decide to work on assignment    **Scenario:** Choose to work online

**Pre condition:**

- Student already logged in

Student	System
1.Select module	2.Redirect to dashboard
3.Select assignment	4.Display assignment details
5.Choose to work online	6. Load the online assignment interface
7.Start working on the assignment	8. Saves progress regularly and display "Saved" messages.

**Use Case:** Decide to work on assignment    **Scenario:** Choose to work offline

**Pre condition:**

- Student already logged in

Student	System
1.Select module	2.Redirect to dashboard
3.Select assignment	4.Display assignment details
5.Choose to work offline	6.Encourage student to download all needed materials
7.Downloads materials	8.Provide downloads links/ files
9.Complete assignment offline	10.Wait for student to upload assignment later

**Use Case:** Decide to work on assignment    **Scenario:** Choose to upload offline work

**Pre condition:**

- Student already logged in

Student	System
1.Select module	2.Redirect to dashboard
3.Select assignment	4.Display assignment details
5.Choose to upload completed offline work	6.Displays the upload link
7.Downloads materials	8.Provide downloads links/ files
9.Upload files	10.Validate file and save submission in the system.
11.Submit assignment	12.Display “ Assignment submitted successfully” and send notification to teacher

**Use Case:** Decide to work on assignment

**Scenario:** Choose to work online and start losing connectivity

**Pre condition:**

- Student already logged in

Student	System
1.Select module	2.Redirect to dashboard
3.Select assignment	4.Display assignment details
5.Choose to work online	6. Load the online assignment interface
7.Start working on the assignment	8. Saves progress regularly and display "Saved" messages.
9.Starts losing internet connection while working.	10. Work saved and then displays “Offline Mode activated! Save manually please!”

**Use Case:** Decide to work on assignment

**Scenario:** Invalid assignment at this time{ overdue assignment)

**Pre condition:**

- Student already logged in

<b>Student</b>	<b>System</b>
1.Select module	2.Redirect to dashboard
3.Select assignment	4.Display assignment details
5.Choose to work on assignment	6. Error message displayed “ Assignment not available anymore”