Project description

Part#2: Black-box testing

I. Description

1. Part A

Choose one of the given applications to test (all functional and non-functional requirements).
https://github.com/amirhamza05/Student-Management-System
https://github.com/francoisjacquet/rosariosis

Make sure that:
- at least 02 functional requirement (as use-cases) that have at least two alternative or exception flows have been tested
- boundary value analysis, equivalent class partitioning, decision table and use-case testing methods have to be used
- and at least one non-functional requirement has been tested.

2. Part B

Choose one of the given applications (in III) and choose a list of its functionalities to test.
Make sure that, each member have to test:
- at least 02 functional requirement (as use-cases) that have at least two alternative or exception flows have been tested
- boundary value analysis, equivalent class partitioning, decision table and use-case testing methods have to be used
- and at least one non-functional requirement has been tested.

II. Report

- Report file type: .pdf.
- T**he application of (black-box testing) methods to generate** the test-case set.
- Test-case description (at least): purpose, pre-condition (if any), data (can be re-used), exection procedure/flow-of-step, expected results (may be presented for each step).
- The execution results and comments/analysis description and PASSED/FAILED confirmation for each test-case.
- For each FAILED, the step flow to re-produce the fault should be provided if it is very difficult to be concluded from the testing procedure and data.

III. Applications to be tested

1. Moodle E-learning system (https://moodle.org/demo)
2. Other application

Please register your application with its description and selected functionalities/non-functionalities to the Lecturer.

IV. Marking schema

	Max point	**Rubric**		
PART A	**4.0**			
1. App spec	**0.5**	Functional: 0.25	Non-functional: 0.25	
2. Testing techniques	**3.0**			
2.1. Boundary value analysis technique	**0.5**			No expected result: -0.25
2.2. Equivalence class partitioning technique	**0.5**			No expected result: -0.25
2.3. Decision table technique	**0.5**			No expected result: -0.25
2.4. Use-case testing technique	**1.5**	Activity graph + Test scenarios: 0.5		Test-cases: 1.0 (steps + expected)
3. Non-functional	**0.5**			
PART B	**5.0**			
1. App spec	**0.5**			
2. Testing techniques	**4.0**			
2.1. Boundary value analysis technique	**0.75**			No expected result: -0.25
2.2. Equivalence class partitioning technique	**0.75**			No expected result: -0.25
2.3. Decision table technique	**0.5**			No expected result: -0.25
2.4. Use-case testing technique	**2.0**	Activity graph + Test scenarios: 0.5		Test-cases: 1.5 (steps + expected)
3. Non-functional	**0.5**			
EXTRA				
Report	**1.0**	Team + report detail		
Missing functional OR simple functional	**-1.0**			
	10			