Non functional requirement

* + Requirement 18: (Should Have)
    - Requirement: The system should be simple and easy to use, but each component should be distinct from any other. For example, 99% of users should be unable to start a test without meaning to.
    - Acceptance criteria
      * Demonstration of
        + The system being used without flaws in 99 out of 100 tests.
        + The system being correctly used by someone with no prior knowledge of the system.
        + Each section (component) of the system e.g. test page vs login page being noticeably different to the other sections of the system

Functional requirement

* Requirement 19: (Should have)
  + Requirement: An advanced package should allow teachers to view student results in a variety of orders (such as ordered alphabetically).
* Acceptance criteria
  + Demonstration of
    - The results for summative tests and formative tests should allow for the order of the results to be changed in a variety of different orders such as alphabetical, chronological, mean, mode.

Functional requirement

* Requirement 20: (Must have)
  + Requirement: There should be two types of tests able to be created, Formative tests and summative tests
* Acceptance criteria
  + Demonstration of
    - The system should only allow the lecturer to create two different types of tests, formative tests and summative tests.

Functional requirement

* Requirement 21: (Must have)
  + Requirement: Formative tests should not record the individual student’s performance.
* Acceptance criteria
  + Demonstration of
    - Once a formative test has been submitted by a student the individual students performance should not be recorded

Functional Requirement

* Requirement 22: (Must have)
  + Requirement: Formative tests – students should be given feedback after each attempt at a question
* Acceptance criteria
  + Demonstration of
    - When completing a formative test students should receive feedback containing advice on where to look for additional help with the question

Non-Functional requirement

* Requirement 23: (Could have)
  + Requirement: Each student’s password should have a minimum complexity
* Acceptance criteria
  + Demonstration of
    - When creating an account the password for a students account should necessitate the password to contain at least 8 characters, 1 upper case letter and 1 number.

Non- functional requirement

* Requirement 24: (Should have)
  + Requirement: Login for both students and teachers require an ID, and the password
* Acceptance criteria
  + Demonstration of
    - The system should require a student to enter both a ID and a password before being allowed to access the system
    - The system should require a teacher to enter both a ID and a password before being allowed to access the system

Functional requirement

* Requirement 25: (Should have)
  + Requirement: Summative Tests – Should record student progress and individual results
* Acceptance criteria
  + Demonstration of
    - For each summative test completed the system should record that tests results
    - For each summative test completed the system should record the individual students results

Functional requirement

* Requirement 26: (Must have)
  + Requirement: Summative Tests – Allow students only a single attempt at any one test
* Acceptance criteria
  + Demonstration of
    - Only a single attempt should be allowed from a student account

Functional Requirement

* Requirement 27: (Must have)
  + Requirement: Summative Tests – The mark for the test taken should be provided immediately after the test is finished
* Acceptance criteria
  + Demonstration of
    - Once a student’s account has completed a summative test the marks for the test should be presented to the student within 5 minutes.