|  |  |  |
| --- | --- | --- |
| The system should allow the customer the ability to create short assessments in the form of tests | **FUNCTIONAL** | **Must Have** |
| A test should contain no more than 10 questions | **NON - FUNCTIONAL** | **Must Have** |
| Customer should be able to choose whether they want the assessment to be formative or summative. | **FUNCTIONAL** | **Must Have** |
| The software should record how many times a student answered a question incorrectly/correctly, at the first attempt when taking a formative test | **FUNCTIONAL** | **Must Have** |
| The student should get feedback after each attempt but only be provided with answers following final attempt. | **FUNCTIONAL** | **Must Have** |
| The system should be adequately protected and secured against unauthorised users. | **NON-FUNCTIONAL** | **Must Have** |
| The system should allow the lecturers to obtain statistics about the tests such as average class score and questions most often answered incorrectly | **FUNCTIONAL** | **Must Have** |
| The lecturer should be able to view results for any one test taken by a particular student | **FUNCTIONAL** | **Must Have** |
| The local program must be linked to an online database used to manage students scores | **NON - FUNCTIONAL** | **Should Have** |
| There should be a teacher login and a student login to allow for students to answer, teachers to create tests etc…. | **FUNCTIONAL** | **Should Have** |
| Each student much have a unique ID number and login to be used as a unique identifier (key) within the database. | **FUNCTIONAL** | **Should Have** |
| The system should be stored on a server, to make sure that students cannot edit any source code, making the tests easier or finding answers. | **FUNCTIONAL** | **Should Have** |
| When taking answers from students allow spelling mistakes or variations of the answer to be correct | **NON - FUNCTIONAL** | **Could Have** |
| The main body of code will be written in a program called python. | **NON - FUNCTIONAL** | **Should Have** |
| Users should be taken to different home screens based on whether they are students or teachers | **NON - FUNCTIONAL** | **Should Have** |
| Students should only be allowed access to their personal results, and the tests they are able to take | **FUNCTIONAL** | **Must Have** |
| Teachers should be able to have access to a higher level of features, including creating new tests and monitoring student progress | **FUNCTIONAL** | **Should Have** |
| The system should work reliably and without fail 99% of the time | **NON - FUNCTIONAL** | **Should Have** |
| 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. | **NON - FUNCTIONAL** | **Should Have** |
| An advanced package should allow teachers to view student results in a variety of orders (such as ordered alphabetically). | **FUNCTIONAL** | **Should Have** |
| There should be two types of tests able to be created, Formative tests and summative tests | **FUNCTIONAL** | **Must Have** |
| Formative tests should not record the individual student’s performance. | **FUNCTIONAL** | **Must Have** |
| Formative tests – students should be given feedback after each attempt at a question | **FUNCTIONAL** | **Must Have** |
| Each student’s password should have a minimum complexity | **NON - FUNCTIONAL** | **Could Have** |
| Login for both students and teachers require an ID, and the password | **NON - FUNCTIONAL** | **Should Have** |
| Summative Tests – Should record student progress and individual results | **FUNCTIONAL** | **Must Have** |
| Summative Tests – Allow students only a single attempt at any one test | **FUNCTIONAL** | **Must Have** |
| Summative Tests – The mark for the test taken should be provided immediately after the test is finished | **FUNCTIONAL** | **Must Have** |
| Summative Tests – lecturers should be able to view and retrieve results for all students after the test deadline | **FUNCTIONAL** | **Must Have** |
| Should be able to run on Windows 10 PC. | **NON - FUNCTIONAL** | **Must Have** |

**Requirements**

Functional Requirements

**Current Idea’s for system layout:**

Python based local program that is linked to an online database containing the exams made by the teacher.

Website setup with a login page that allows the exams to be answered online. Both code and database stored online.