Analysis

Problem identification

Creating and evaluating paper tests is a timely process. Many teachers are not able to dedicate time to supporting students due to the time needed to mark and give feedback on a test. As a school year progresses, the amount of knowledge that a teacher needs to test increases, taking even more time from providing support or teaching new knowledge.

I have inquired on the use of tests in many schools. They all stated that paper tests were the only type of examination in their schools, and when teachers were asked if they believed electronic examination would be efficient in terms of time, they all agreed. Furthermore, the use of electronic examinations would save schools' money on printer ink and paper, allowing more budget to be dedicated to new facilities. Also, in reducing the time it takes to create, do, and evaluate tests, there is more time to create personalised tests for specific students and more tests for a class overall, increasing the schools' average GCSE and A-Level results. The use of electronic tests allows for tests to easily be used outside of school, saving time in lessons to studying new topics.

My client, Mr Williams, is a history teacher at Pine Hill secondary school. Pine Hill secondary school is a Catholic secondary school consisting of 720 students. In the past the school has been acclaimed as one of the most prestigious in its area, however in the recent years the school has seen its educational performance decay primarily linked to financial problems. Mr Williams believes that due to the decrease in resources, teachers have not been able to distribute an adequate number of tests, exams and quizzes to classes, leading to decline success in students' final exams.

My client has requested the creation of a program to help reduce his constantly increasing workload due to the creation, collection and evaluation tests as the school year progresses. He exclusively uses paper tests for his students but is looking for an electronic alternative that can help him save time. His problem is amenable to a computational approach as the use of an electronic program can reduce his workload in distributing tests, automatically collecting student's results, and calculating a class's average score.

My solution is to create a program that allows a user to login to the program, this login will be able to differentiate a teacher and student and give permissions to the user appropriately. My client will be able to create a test, which is accessible by all students using the program to complete. Once completed, a student's results will be displayed to them and will be automatically collected for the client to evaluate, and feedback can be personalised feedback can be given to a student via the program.

Computational methods

Abstraction: Unnecessary features of creating a test will be removed. When running the program, the contents of all .csv files will be shown through the program instead of through an external window such as Excel. Manual marking and evaluation of a class's test is removed as the program automatically calculates the score for both a student and the class. Only necessary features of registering or logging in an account will be removed, only a username and password will be needed to register an account, information such as: date of birth, email, address will not be

required. Only accessible features based on the user's account permission will be visible on the secondary menu following login, for example, an account with admin status will be able to view all options on the menu (including creating a test and giving feedback), whereas an account without admin status will not be able to view all options on the menu as they will not have the permissions to access all options (will not have access to creating a test and giving feedback). When receiving feedback, only relevant feedback will be displayed to the user (either feedback directed to the user or directed to all users who have completed a test).

Decomposition: The program is broken into account registration and login, test creating and taking, and sending and receiving feedback.

Pattern Recognition: Every user will need to register an account once, and login every time they use the program. Teachers will repeatedly enter questions for a test. Students will repeatedly enter answers to a test.

Algorithms: Algorithms will be used in every process to make the program as efficient and simple as possible. Each function or process will be a separate algorithm e.g: the main menu, registration menu, login menu. Flowcharts and psuedocode will also be used when designing the program to prevent me from encountering any problems while developing the program. This will also prove as a visual aid to anyone trying to understand the program.

Stakeholders

My client, Mr Williams, has an interest in the program as it will reduce his workload and increase his class's exam results. He will need the program to able to make tests quick, therefore the program will need to be simple and easy to understand. As he teaches History, many tests he will create will contain essay style questions, therefore the program will require a feedback system as these questions will not have a definitive answer.

Other teachers at Pine Hill secondary school, will also benefit in saving time and resources by using this program. These teachers will also try the program; therefore, the program will need to be simple and easy to understand and contain a guide/help menu.

Students will be taking tests using the program and will also benefit by being able to access tests and review feedback on tests from both school and home. Furthermore, text and multiple-choice based tests will automatically provide a result, saving a student time in waiting for a teacher to collect, mark and return tests.

Governors have an interest in the program because the program will increase the school's overall educational performance by consolidating students' knowledge for exams and increase financial performance by saving money on paper and ink.

Headteachers will be interested in the program because it will increase the school's overall exam grades, making the school an appealing option for parents and students when selecting a secondary school to attend.

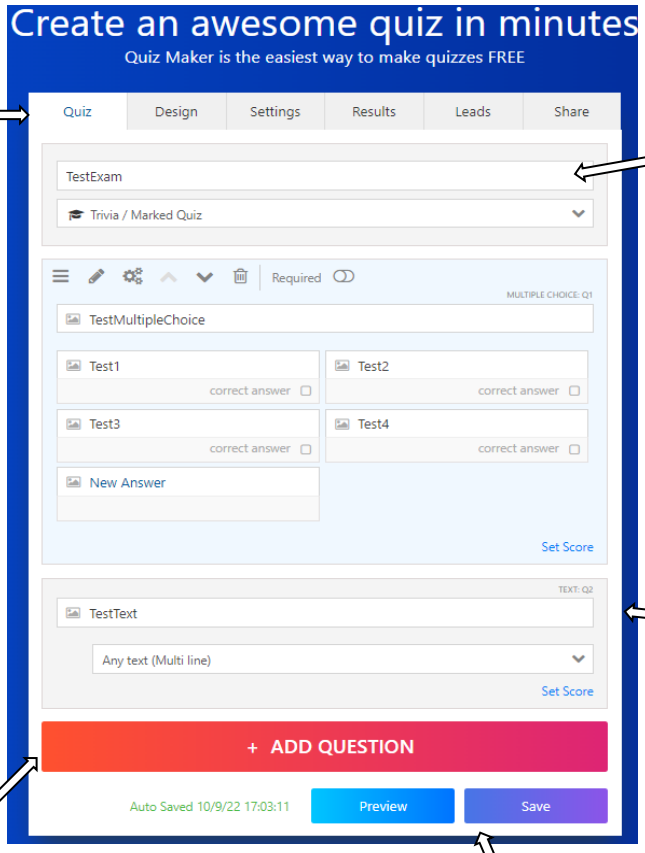
Parents have an interest in the program as it will increase their child's academic performance leading to parents being more satisfied with the school's teaching.

Research

In order to determine what my criteria will be for this program I will use a range of research methods. First, I will analyse an existing and successful test creator product to determine some features that may have contributed to its success. **Otherwise, I will not know what features are present in a proven model and therefore I will find it difficult to determine essential features of a test creator**. Next, I will conduct an interview with my client in order to assess what their problem is and determine the best solution to their problem. **Otherwise, I will not be able to get a**

detailed description of my client's requirements, without this it would be difficult in ensuring and determining how successful my solution is to my client's problem. Finally, I will use two questionnaires, one for teachers and one for students at Pine Hill secondary school to understand my users' experience with electronic tests and determine what features they would like to see present. I am using a questionnaire as opposed to interviews as I will to gather data from 63 people combined. **Without a questionnaire, it would be difficult to analyse the data received and conclude how the data will influence my solution, furthermore it would take a long time to conduct all interviews.**

Similar product



Multiple easy to access tabs. I will keep this feature, but instead dividing parts of the program into windows. **Otherwise, the program would not be easy to use**

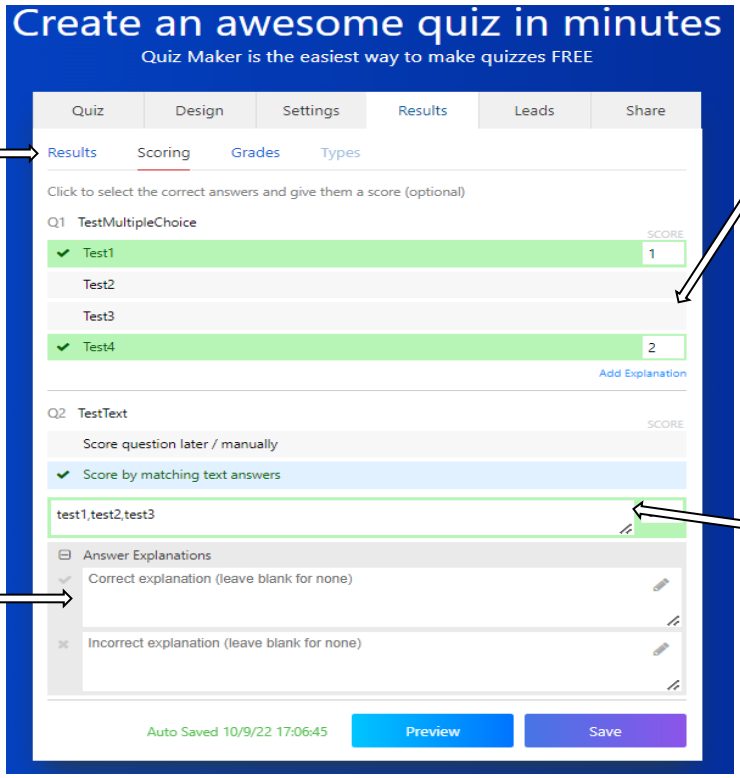
Easy to access input boxes. I will keep this feature, using as few input boxes as possible to make it easy to use and using appropriate labels to make it easy to understand. **Otherwise, the program would not be easy to use or easy to understand**

Simple design and colour scheme. I will keep this feature to have a straightforward design. **Otherwise, the program would not be easy to use or easy to understand**

Add question button. I will keep this feature so that the user can save the question when they are ready. **Otherwise, the user will not be able to choose when to save their current question**

Preview and save buttons. I will keep this feature to make creating a test as easy as possible by being able to view the test during creation. **Without this, the test creation feature would not be easy to use**

Use of results, scores and grades. I will keep this feature to be able to view results and influence feedback. **Otherwise, a user will find it hard to evaluate their results on a test**



Allows both text and multiple choice questions. **I will not implement this due to time constraints**

Allows multiple answers for text questions. I will keep this feature to make answering questions as easy as possible. **Otherwise, users will find it difficult to answer questions as the program would only accept one answer**

Allows explanations to answers. **I will not use this feature as the feedback giving and recieving process covers this**

www.quiz-maker.com is a popular quiz making website. There are many features I can implement into my program such as: a simple design and colour scheme, the option for both text and multiple-choice style questions, explanations to answers, a preview button when creating the test and the use of results, scores, and grades.

Client interview

To understand my client's requirements, I have conducted an interview to know the features he would like in my program. **Otherwise, I would not know how my client can benefit from my program and what they would like to see as a solution to their problem.**

1. What is your current problem?

I've started creating and distributing tests for my students to help them in their upcoming exams, however marking these tests and giving feedback is a timely process that takes up the majority of my time, when I should be providing additional support and teaching the next topics.

2. How can I solve this?

I need a program that allows me to create tests, review results and provide feedback for students efficiently. It would help me by automatically collecting test scores so I can focus more on supporting my students that need it the most.

3. What experience do you have in using electronic tests?

I have created tests in Microsoft Word, I have found it to be extremely simple to use, however I still rely on manually counting marks, calculating a grade, and writing feedback on tests.

4. Do you have any specific requirements for the program?

Yes, it should be easy for my students to use and have a straightforward design. It should also show how long it takes for my students to complete, so I can make sure my students aren't cheating if they are taking a test at home.

The first and second questions establish that the main issue my client is facing, is the time it takes for him to mark and provide feedback on these tests. From this I know that my program will have to be able to help make tests quickly, therefore the program will need to be easy to understand and easy to use. Furthermore, the program would need to allow him to provide feedback on a student's test results, therefore I will have to add feedback sending and receiving system.

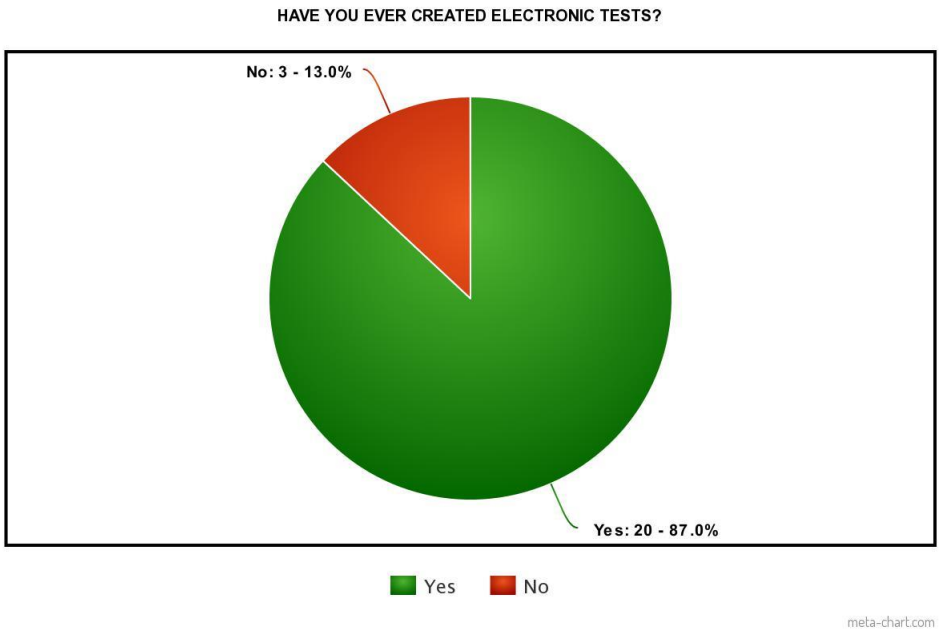
The third question establishes that my client has no experience in handling results and feedback electronically, therefore the program will have to be easy to understand.

The fourth question shows that a timer will have to be added to the program and linked to each result. Furthermore, it reiterates the need for an easy-to-use program and simple design.

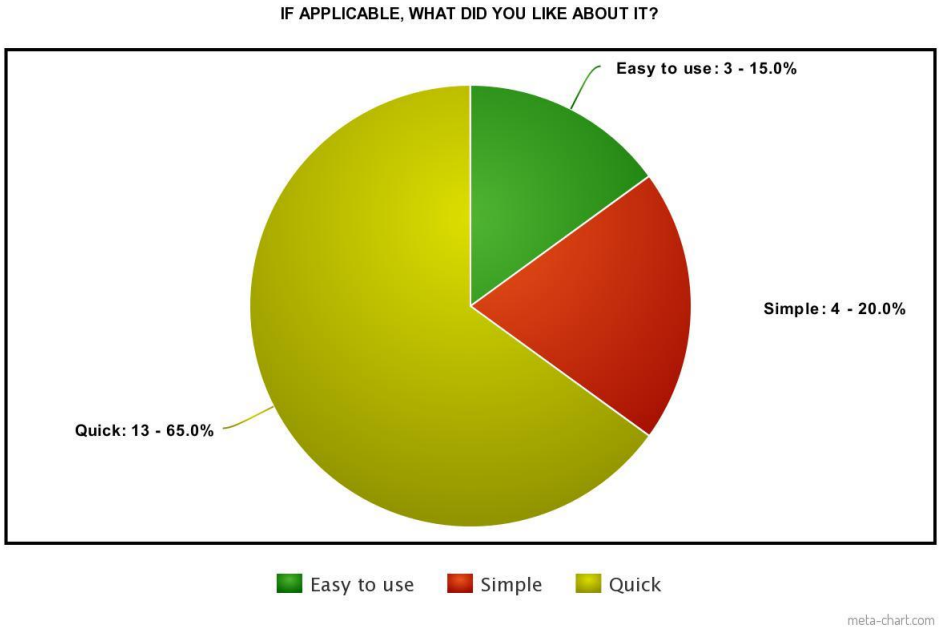
<u>First questionnaire (teachers)</u>

To gain a better understanding of how teachers would use my program, I created a questionnaire for the teachers at Pine Hill secondary school. **Otherwise, I would not know how other teachers can benefit from my program.**

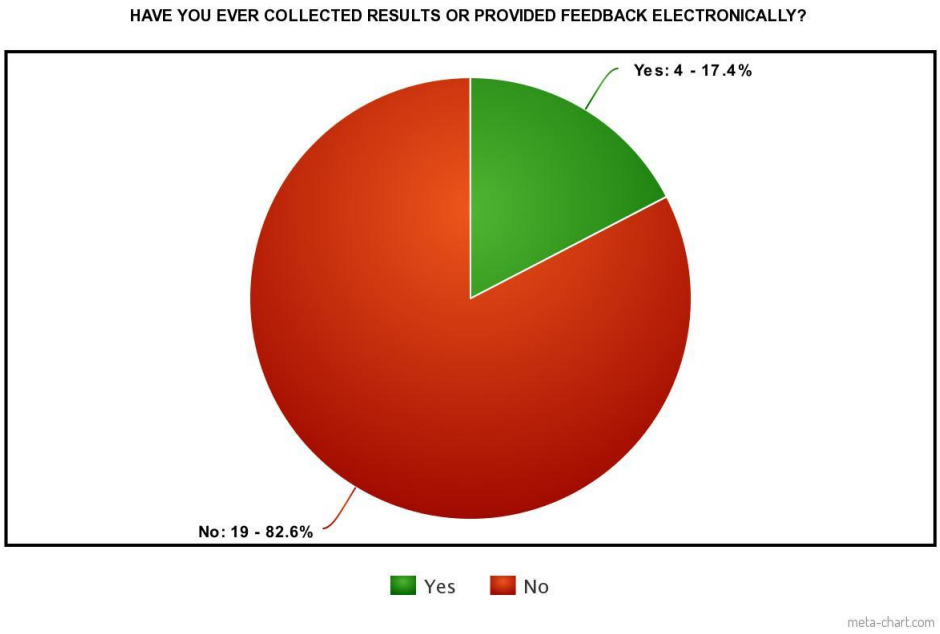
1. Have you ever created electronic tests?
   - Yes [20]
   - No [3]



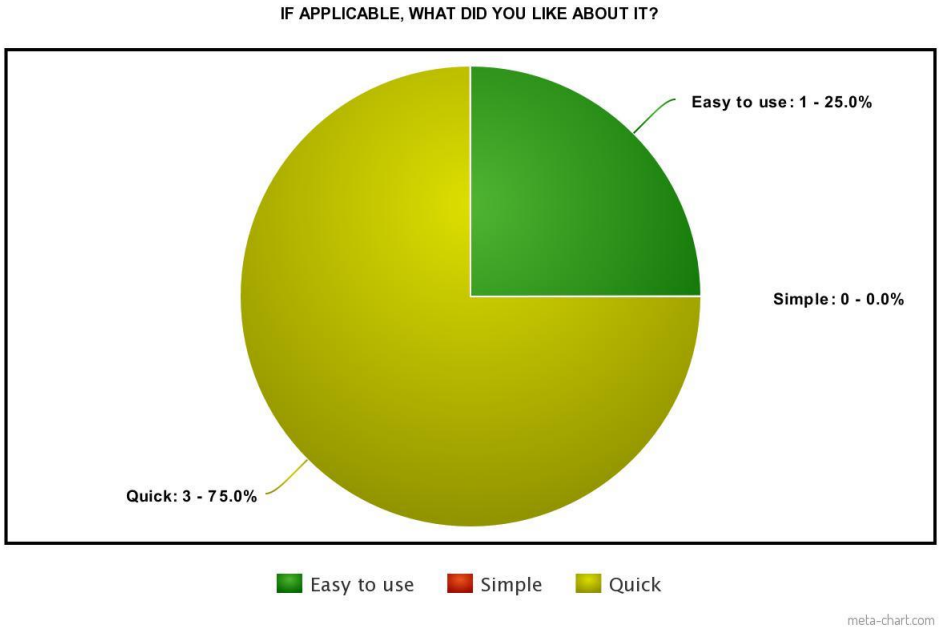
2. If applicable, what did you like about it?
   - Easy to use [3]
   - Simple [4]
   - Quick [13]



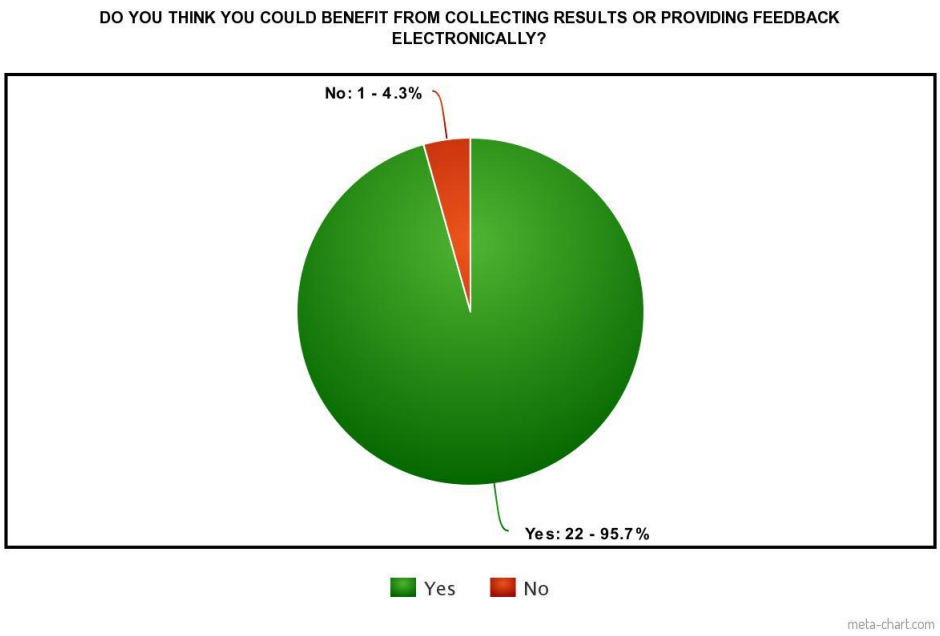
3. Have you ever collected results or provided feedback electronically?
   - Yes [4]
   - No [19]



4. If applicable, what did you like about it?
    - Easy to use [1]
    - Simple [0]
    - Quick [3]



5. Do you think you could benefit from collecting results or providing feedback electronically?
    - Yes [22]
    - No [1]

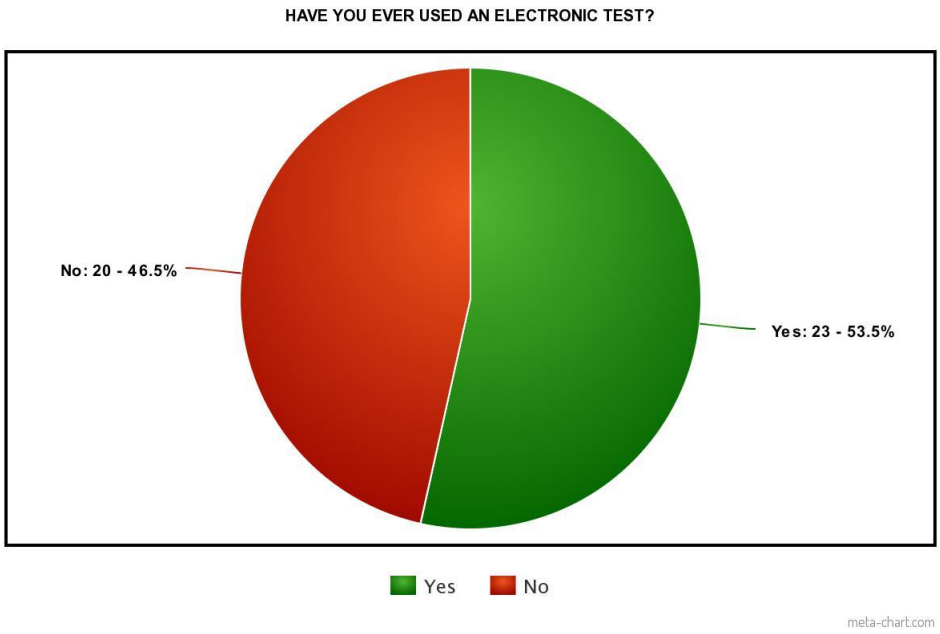


This questionnaire's answers similarly reflect the answers of my client. As these teachers, may not understand how to use the program as well as my client, I will have to ensure that my program remains simple and easy to use, especially in the reviewing of results and sending of feedback.

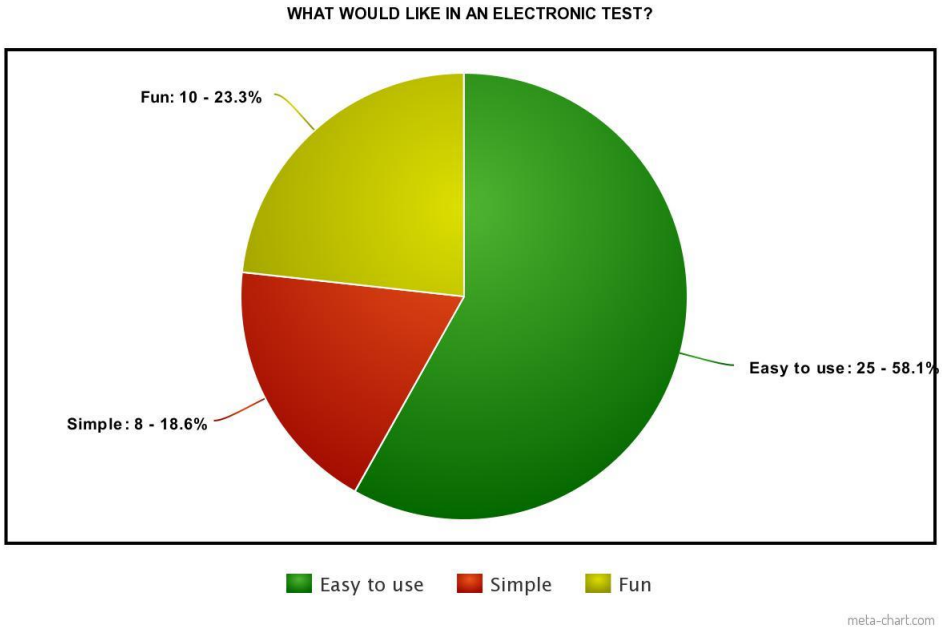
<u>Second questionnaire (students)</u>

To gain a better understanding of how students would like to use electronic tests, I created a questionnaire for a sample of students at Pine Hill secondary school. **Otherwise, I would not know how students can benefit from my program.**

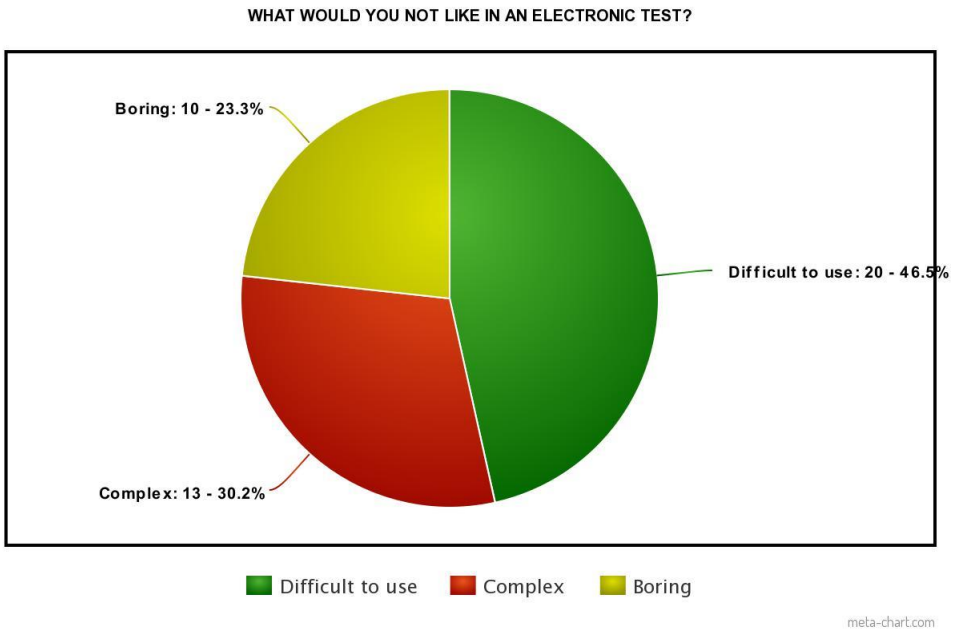
1. Have you ever used an electronic test?
   - Yes [23]
   - No [20]



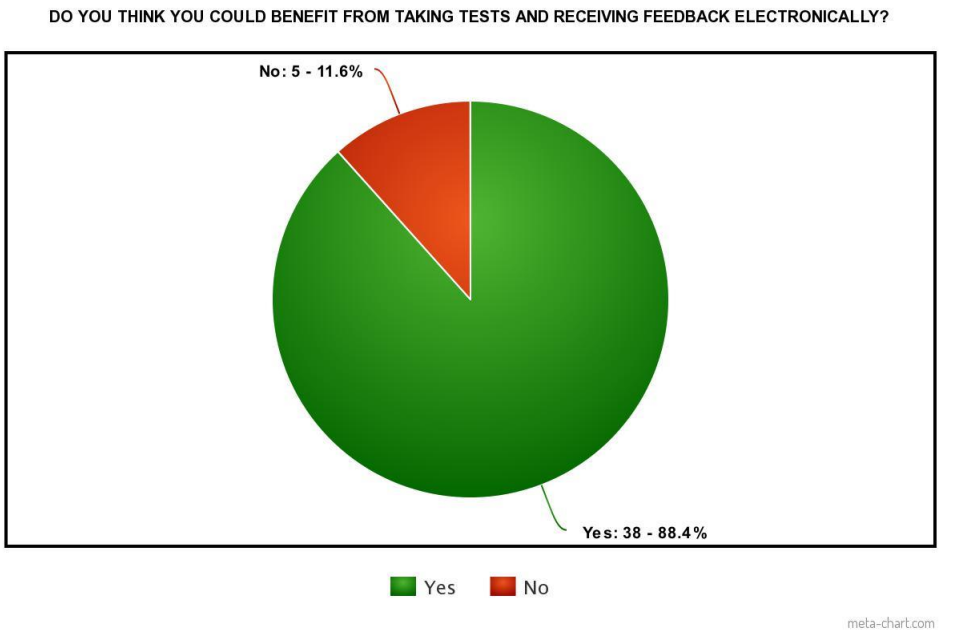
2. What would like in an electronic test?
    - Easy to use [25]
    - Simple [8]
    - Fun [10]



3. What would you not like in an electronic test?
    - Difficult to use [20]
    - Complex [13]
    - Boring [10]



4. Do you think you could benefit from taking tests and receiving feedback electronically?
   - Yes [38]
   - No [5]



The first and second questions establish that most students have used an electronic test before, however a significant percentage have not, this further reiterates the requirement for an easy-to-use program.

The third question shows that a large group of students find electronic tests boring and complex, therefore I will ensure that tests can remain simple and be varied in question types.

<u>Client requirements</u>

From the research above, I can conclude that my client's requires the programme to:

- Allow the user to create tests

Otherwise, the client is not able to use the program as an electronic alternative to paper-based tests

- Allow the user to review results

Otherwise, the client is not able to use the program as a quick way to assess results from his students

- Allow the user to provide feedback

Otherwise, the client is not able to use the program as a quick way to give feedback to his students

- Be easy to use

Otherwise, the client is not able to use the program as a more efficient and faster alternative to creating, distributing, collecting, evaluating and giving feedback using his current system. Furthermore, the program must be easy to use for his students and other teachers, who will not be as knowledgeable in how the program works compared to my client.

- Have a simple design

Otherwise, the program will be complicated, and users may find it difficult to use

Solution requirements

The program will start with a registration/login system to differentiate and authorise different permissions based on whether the user is a student or teacher. **Otherwise, the program does not know if a user is a teacher or student, and therefore cannot give admin privileges where necessary**. A teacher will be able to create a new test, which will be stored as a .csv file, before creation the test can be previewed. **Otherwise, the user cannot see what their progress is like when creating the test, making the program harder to use.** This test can include both text or multiple-choice questions that can be selected to vary the questions or if one option is more appropriate for a test. A teacher will be able to create grade boundaries for a test. **Otherwise, the teacher must assess the results by using marks, which is a harder process, making the program harder to use**. A teacher can modify a test, through stated inputs in the program, which will modify the .csv file's contents. **Otherwise, a teacher cannot add to a test using the program**. Both a teacher and student can take a test where, questions will be outputted one by one and an answer can be submitted, through an input box. **Otherwise, a student cannot complete a test as they cannot input an answer**. At the end of a test, the program will calculate the overall score and output it as both a score out of the total and as a percentage, as well as output the grade achieved. **Otherwise, a student cannot view their results and assess how well they did**. A teacher will be able to send feedback to a student's account and feedback can be received by students. **Otherwise, a teacher cannot help a student improve through the program**. The feedback will be stored on a .csv file, if a student has received feedback, upon login the program will search for feedback linked to their account and display it. **Otherwise, a student is not able to view any feedback given to them by a teacher**. To ensure the program is easy to use a 'help' menu will be accessible at any point. **Without this, a user may find the program hard to use or difficult to understand.**

Success criteria

Number	Requirement	Evidence of success	Justification

1	Each system is windowed	Screenshot of multiple windows open for each feature	To allow multiple parts of the program to be used at once. **Otherwise, the program will be hard to use as you would only be able to use one part of the program at a time**
2	Each input is provided using a keyboard	Screenshot of input box using text	To make the program easy to use. **Otherwise, the user will find it difficult to enter an input and the program would be unable to access a user's input**
3	Main menu	Screenshot of the main menu	To access various parts of the program. **Otherwise, the registration, login and guide features would be inaccessible to a user**
4	Register button	Screenshot of register button	To open a separate registration GUI for easy account registration. **Otherwise, the user would not be able select the registration option.**
5	Registration system	Screenshot of .csv file updating with details of an added registration	To differentiate and authorise different permissions to the program based on the status of the account as either a teacher or student. **Otherwise, the user would not be able to create an account, which is needed to access the program.**
6	Login button	Screenshot of login button	To open a separate login GUI for easy logins. **Otherwise, the user would not be able select the login option.**
7	Login system	Screenshot of access to program given after successful login, and no access to program given after unsuccessful login	To allow access to the program to authorised users. **Otherwise, the user would not be able to verify themselves, without this verification they would be unable to access the program.**
8	Account permissions	Screenshot of admin menu and default menu, based on the account logged in	To differentiate and authorise different permissions to the program based on the status of the account as either a teacher or student. **Otherwise, all users would be able to access the special features of the program, this is a security issue.**
9	Guide button	Screenshot of guide button that opens the guide	Having a guide in a separate window makes it easily accessible while using any feature of the program. **Otherwise, the user would not be able select the guide option.**
10	Guide	Screenshot of window, with information on how to use various parts of the program	To allow any user to understand the features of the program with no external help required. **Otherwise, a user would not be able to get help if they did not understand the program**
11	Exit button	Screenshot of 'exit' button	To allow a user to easily close the program
12	Create a test button	Screenshot of create a test button	To start the create a test process. **Otherwise, the user would not be able to select the create a test option.**
13	Create a test system	Screenshot of a newly created .csv file	To be able to create and store multiple tests. **Otherwise, a user cannot create tests to test their students' knowledge.**
14	Preview button	Screenshot of visual representation of .csv file	To be able to view the test during creation. **Otherwise, the user cannot easily view their progress on the test's creation.**

15	Select text or multiple-choice questions	Screenshot of created .csv file for both types of tests	To be able to select appropriate form of test based on questions
16	Create grade boundaries	Screenshot of .csv file with grade boundaries	To easily represent how good a score is.
17	Modify a test	Screenshot of .csv file updating with questions and answers	To be able to add questions and answers to a test. **Otherwise, questions and answers cannot be added to the test, therefore there would be no knowledge to test.**
18	Complete a test button	Screenshot of complete a test button	To start the complete a test system. **Otherwise, the user would not be able to select the complete a test option.**
19	Complete a test system	Screenshot of questions being outputted from a .csv file onto the program and an input box for the answers	To be able to use the test, to assess knowledge. **Otherwise, the user cannot complete created tests and will not be able to evaluate their knowledge on the topics in a test.**
20	Display user's score and time	Screenshot of outputted user score out of total score, and percentage score, and the time taken to complete the test	To be able to know how well you did on the test and how long you took. **Otherwise, a user cannot evaluate how well they did on a test.**
21	Feedback sender button	Screenshot of feedback sender button	To start the feedback sender process. **Otherwise, the user would not be able to select the give feedback option.**
22	Feedback sender system	Screenshot of message being sent to a specific account	For a teacher to be able to submit feedback via the program regarding a quiz. **Otherwise, the user would not be able to give feedback to users on a test.**
23	Display all users' scores, grades, and times	Screenshot of outputted list of students' scores, grades and times, and average score, grade, and time on the test	To be able to evaluate how well a student or a class understands the topic given the score and time taken. **Without, this a user would not be able to evaluate the results on a test and give appropriate feedback**
24	Feedback receiver button	Screenshot of feedback receiver button	To start the feedback receiving system. **Otherwise, the user would not be able to select the receive feedback option.**
25	Feedback receiver system	Screenshot of message being received by a specific account	For a student to be able to receive feedback via the program regarding a quiz. **Otherwise, a student would not be able to receive feedback on their test results.**
26	Easy to use	Screenshot of clear titles and labels for all windows and input boxes	To ensure both teachers and students understand how the program functions. **Otherwise, the program would not be efficient as a solution to the client's problem.**

<u>Limitations of the system</u>

The create a test system will only be able to award 1 mark per question, therefore long and essay-style questions cannot be automatically marked and will require manual marking. While this could be fixed by setting marks to be

awarded based on keywords used in the answer, this would be time-consuming for a teacher to enter all related keywords and would still likely require manual review of answers. **Without this problem, longer and more complicated questions could be automatically marked by the program saving a teacher even more time**. **Otherwise, the program will still need manual marking making the program less efficient.** The program runs on Python, which is not compatible with older versions of the Windows OS. **Without this problem, a user would be able to run my program on any Windows OS or even any OS making the program accessible to all users, otherwise a user would need to install a valid OS in order to use the program.**

Software and hardware requirements

Software requirements	Justification
Windows 7 or above	Otherwise, the program will not function properly or may not function at all with an older OS
Access to a web browser	The program will need to be downloaded through a web browser using an internet connection. Otherwise, the program cannot be installed on a computer.
Python 3.2.3 interpreter	The program is written in Python 3.2.3 and will require an interpreter to run. Otherwise, the program cannot be translated into an executable format.

Hardware requirements	Justification
Intel® Core™ i5 or equivalent	The minimum CPU requirement to run Python with no performance issues. Otherwise, the user may find the program slow to use, which will make it less efficient.
52kb storage space	All required files take up 52kb of space however, more will be needed when installing tests or creating tests. Otherwise, the full program cannot be installed on a computer and without extra space tests cannot be installed or created.