OCR GCE A

COMPUTER SCIENCE PROJECT

H446-03

Name : <LLOYD NANA - AGYEMANG>

Candidate Number : < 3967 >

<Institution Name> : < 20780 >

Title of Project : < REVISION APP >

H446-03 – Project CONTENTS

Table of Contents

[A. Analysis 3](#_Toc452555018)

[B. Design 3](#_Toc452555019)

[Systems diagram 3](#_Toc452555020)

[C. Developing the coded solution (“The development story”) 3](#_Toc452555021)

[D. Evaluation 3](#_Toc452555022)

[Project Appendixes 4](#_Toc452555023)

# A. Analysis

Chart, treemap chart

Description automatically generatedGraphical user interface, application

Description automatically generatedRESEARCH AREA

**QUIZZIZ**

A screenshot of a computer

Description automatically generatedGraphical user interface

Description automatically generated

FIGURE 1

FIGURE 3

FIGURE 4

FIGURE 2

Here is an existing revision app, shown figures 1, 2 and 3 is the question-and-answer format of the question. From view it is more of a friendly, relaxing, and calm approach for users. Certain power ups are included within a game to make it more of a competitive and active. For the user you could say it could encourage them to visit this app but for A Level student I don’t think it would be revising appropriate because of how in depth the content will be. Quizizz allows the user to create a free account and allow him to participate in a quiz (joining via link or code) or to host a quiz. What I plan to do is to infuse both at the same time making sure content is covered at an interesting rate.

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

**KAHOOT**

Graphical user interface, application, Teams

Description automatically generatedGraphical user interface

Description automatically generated

FIGURE 7

FIGURE 5

FIGURE 8

FIGURE 6

Here is a similar application to Quizizz, kahoot. Just like quizizz it allows the users to create an account and create multiple choice questions for the user to share and play with others. In figure shows what it will look like when asked the question - 10 seconds will be given to think about the question so that when the four options are brought up, they can answer straight away. Although this does not mean they can't answer after answer options are presented, they have 30 seconds to do so but the issue is they risk losing points due to the speed of answering. This could be a con because it puts pressure towards the user to pick an answer without enough time to process the answer. This could also be a pro because it will be testing what the users pre knowledge. After all this however they cannot view their progress because they have joined via code. In my program the user whether staff or student will have to create a login so that progress can be checked and evaluated.

STAKEHOLDER

The program I would like to create is an ideal revision app for computer science students, mainly for first year’s students but second year students can also use it for a refresh of memory. My aim is to try intrigue, motivate and encourage fellow students to use this app as well other resources however this as the priority go to. This app should make it interesting to study along with self-notes making sure they tackle and identify areas of struggle.

This program can also be used as a simple application to creates quizzes for students as well as exam questions to help boost their revision.

I decided to interview computer science students and teachers on their opinions on the creation of this app and if they would want specific features to be added. Furthermore, if they in the first-place revise and if not what in the app could encourage them.

***YR12 STUDENT***

*Interview with Stakeholder A – Student* (Gary)

***How is revision going for you?***

“Honestly It is alright, but it could be better”

***From the previous question how could it be better?***

“I would like to feel engaged when doing my revision, I suffer in a lack of motivation so I would like to be encouraged into revising. Using not only my notes but other resources”

***What are your opinions on revision apps, and what would you like to see in them?***

“I would like to be able to access and complete tasks in the revision app anywhere. I do think revision apps are useful and practical.

***Would tracking your progress help?***

“Yes it would help for revision in the future”

***TEACHERS***

*Interview with Stakeholder C – Teacher*

***what would you like to do in this application?***

“I would like the app to allow students to build their knowledge for discuss question in certain topics and quick-fire questions. If they could then access their answers and feedback later or at any time that would be good”

***Would it be helpful if you could switch between student and class view when logged in?***

“That would be helpful I think”

***On teachers view what privileges would you like?***

“I would like to be able to add comments to students attempts to my own pre setup activities, which will be multiple choice and short answer questions. Tracking progress of students would be a shout as well”

***I have come to decide to create these options: A multiple-choice section, study and recap section and exam practice questions. Are these options enough? If so what else would you like to add?***

“This already is a great start and sounds great for students”

***Summary of interviews***

*Student interview:*

After questioning this student, I discovered that he lacks motivation in his college studies. I'm going to make sure that the app doesn't bore or demotivate him. I'll do my best to make it as relatable as possible so that he or she becomes interested in their rewrite.

*Teacher interview:*

From the teacher's viewpoint, she wants to be able to watch her pupils' development so that she can ask them more difficult questions or teach them more effectively. She wants to be able to provide feedback to each student who has completed each question. The teacher wants to be able to help his or her students as much as possible, and it is my task to want to do so within my skill set.

LIMITATIONS

A limitation of my project is that the information covered within it will not cover topics from other exam boards, therefore students taking AQA or Edexcel exams will only have access to a restricted amount of content because my project will only cover subject from the OCR exam board. This is primarily due to a lack of time, but it's also owing to the fact that I've never studied computer science from any other test boards, which would make incorporating it into the software extremely complex and time consuming.

Another of my project's limits is that it is definitely a sort of software with a multiplayer component. One of the things that makes revising applications so pleasurable is that they allow you to do just that. Unfortunately, I was hoping to include this into my project. This is a more complex aspect of programming that does not fit into the current structure of what I want the project to accomplish since I lack the networking code skills while still wanting to fulfil the program's main aim - a sufficient revision tool. With a feature like this in my programme, it would increase the app's popularity, meaning students would want to come back to it since doing things like revising with friends would be really beneficial.

ESSENTIAL FEATURES

USER PROGRESSION TRACKING SYSTEM — Behind the scenes, a tracking system must be in place to ensure that students' important problems, that they are having difficulties in with an answer. This would need the creation of a database that would store the user's performance for each question asked. It will also keep track of how much a user (student) does when a certain question is displayed. They will either require less or more editing according on how well they have answered. This means that this question will either be shown because of the pupils' performance or because it will aid in revision.

REMINDER SYSTEM – Students have the option of receiving regular reminders to take and finish various quizzes. Notifications will be sent out based on the users' specified reminders, whether on specific days or at specific times every day. By doing so, I can ensure that students perform frequent revision, which will benefit computer science students' knowledge.

LOGIN SYSTEM – If students attending the same school, they will need their own personal logins as the more than one computer will be used by more than one student. This computer will have to be enabled to carry out multiple logins, all of this will be stored in a database whilst being encrypted so each users’ details are kept between themselves. This should allow students to see their progress individually rather than measured with other students.

SAVE PROGRESS SYSTEM – The ability for users to save their results into a database is an important feature I've chosen to provide in my programmed. This means users can go to their profile and enter their results once they earn a score on one of the quizzes. The data will then be saved for them to view when they return to their account later. Kahoot does not have a feature that permits students who do not create an account but join via pin to store their scores over multiple sessions. This is something I've thought about, and I've made it a high priority for users to form an account right away so they can save data.

TEACHER TO STUDENT VIEW – this feature will allow the teacher to watch and see from a student’s point of view. This is because the teacher may want to add anything to the specification that they believe is not well covered in detail but is vital. It can be added to the system by the teacher for the student to see and benefiting from.

HARDWARE AND SOFTWARE REQUIREMENTS

***Mouse/Trackpad***

Some sort of mouse like device would be required to give a response to given questions in quizzes and navigate around the program’s menu. Essentially the left click will be needed to click buttons on program like selecting the topic or answering questions. It will not be hard to adjust as everyone has some sort of smart device.

***Operating system***

The recommended operating systems I would suggest are windows 10 or windows 11 as they are the most recent pieces of systems. They would be the smoothest as they are most recent allowing the program to run smoothly and quicker. Also, notifications that were missed can be viewed.

***Display and resolution***

1280x720 resolution would be required to fit the content on the screen e.g. images and diagrams, multiple – choice and text buttons. It will be enough for the user to see and to be visualise multiple things at the same time.

***Microsoft***

To track the user’s performance any version of Microsoft Access is required. Having access installed will allow the main function of the revision program to work.

***A Keyboard***

The user will need a regular QWERTY keyboard to get into their accounts and put out their responses to the longer questions.  It will continue to serve its purpose. As a result, the decision between the two is entirely reliant on the preferences of the USERs

***Access Database***

For the stakeholder to access the database that is connected to the programme and store their work, they must have the most recent version of Access database. A recent version will be preferable. This will help to avoid any issues and guarantee that the software remains user-friendly.

COMPUTATIONAL THINKING

For a variety of reasons, the software I'm developing will combine many different sorts of computational thinking.

***Abstraction***

The generation of the revision material is an example of abstraction employed in the software. Because the computer science specification contains a lot of information, it's crucial to remember that not all of it will be required to be included in the project. When you think abstractly, you'll notice that the stuff that's most important to you is more likely to be highlighted.

***Decomposition***

Decomposition is a technique involved in production of nearly every project you can imagine, including this one. Breaking down the development of a revision software into smaller, more manageable chunks. For example, the construction of buttons and icons, as well as the deployment of databases inside the login system, may be split down into smaller, more manageable parts. Concurrent thinking may be employed in this situation since similar tasks can be completed at the same time to save time.

***Divide and conquer***

This is a method that heavily relies on recursion. This involves breaking the problem down into smaller pieces and then putting them together.

***Problem recognition***

With the revision app I'm making, I hope to address the issue of weakness tracking, which will allow students who use the app to work on their faults, as well as the issue of students losing interest in an app much too quickly.

***Iteration***

The usage of iteration is evident throughout the project, and one example is the secure login method developed in the application.

The user must match a username and password to one that is already registered on the system; however, if the information don't match anything that is already registered, a loop occurs, forcing you to keep entering data until you've provided anything that is already recorded.

SUCCESS CRITERIA

MUST

***Simple log in system***

This is a straightforward system that allows people to log into an existing account or create a new one.

***Difficulty based questions to help those off different skill levels***

Because not all students will be on the same level, this is one of the issues I'm attempting to address with my project.

***Some kind of progress tracking to show whether or not something has improved or decreased.***

Knowing how much you're improving when editing has a negative impact on your ability to rewrite.

***An attractive Ul, similar to 'Quizziz, Kahoot, Seneca Learning’***

The purpose of this is to avoid boredom by making the app more engaging.

***Types of revision***

Exam questions and quizzes are examples of sorts of revision that can be used to assess your progress.

COULD

***Content for GCSEs***

This would be beneficial for younger students who may want help with certain basic concepts of computer science because they did not study it in year 11.

***Exam board material differs.***

Even though I'm mainly focused on OCR test material, if there is time left over at the conclusion of the project, I might consider include some AQA material.

***More challenges***

Because I'm attempting to include difficulty-based questions, adding even more difficulties would be ideal for accommodating students with varying degrees of understanding.

***Various methods of revision***

Adding additional sorts of revision, such as flashcards, if time allowed, would assist vary the numerous methods people may revise.

WOULD

***Aspects of multiplayer***

As demonstrated in my study for my project while looking at other revision apps, if I had the skills of coding multiplayer elements into this app, it would assist keep students more interested with the software.

***For written questions, special marking is possible.***

This would require some form of submission procedure that would go straight to the leader but would take too long.

***All 3 exam boards***

I could integrate information from OCR, Edexcel, and AQA if I had significantly more development time than was allocated.

***Checking friends' progression***

this would be a feature where a student can check the progression of

classmates to see what level they're performing at currently.

SHOULD

***Easy-to-navigate Ul for the user***

This is critical since a difficult-to-navigate application can be a huge turnoff.

***A significant amount of content***

This app has to have enough content to ensure that students who use it can revise well.

***A quiz system that works***

For quizzes to function properly, features such as a time restriction and a correct marking system must be included.

***There are enough quizzes with varying degrees of difficulty.***

This is critical to the programme because without it, I wouldn't be able to handle the reality that not all students in my course are on the same page.

***Enough feedback was provided.***

It's critical that enough feedback is supplied in the test evaluations to help increase the project's complexity while also ensuring that the kids running the app are making progress.

# B. Design

## Systems diagram

PROGRAM FUNCTIONS

***Generate Code***

When a teacher or student registers up, they are given a unique code that allows them to be identified. This will be seen on their account as a preview.

***Ratio/Percentage of Answers***

After a student completes a quiz, it will assign a grade out of a possible 100, then a percentage, and finally, based on the number of quizzes completed, a ratio of the success rate after a specific percentage.

***Start of Quiz***

When the user selects a topic, he will be given the option to revise, quiz, or do an exam-style question, which will display a large box, or multiple choice with a timer and questions. The timer will be used to help students prepare for a real exam or paper.

***Answer, Complete/Submit, and save***

The user will be answering questions and finishing them. Once the user completes/submits the form, it will be saved to their account immediately. The teacher will be able to review the quiz once it has been finished and saved in the user's account.

***List of Topics***

This will be displayed as a tab within the programme, from which the user can select.

***Add revision, quiz, and exam style question***

This feature is only accessible to teachers so that it is a reliable source to them to ensure accurate revision. If the teachers feel so, he or she can add more revision tasks, quizzes, or exam questions for a student. This will help in development for the student’s progress. Furthermore, they can make their custom questions.

PSEUDOCODE AND FLOWCHART

A screenshot of a computer

Description automatically generated

Initial problem plan

Graphical user interface, table

Description automatically generated

Quiz format visual

yes

no

A screenshot of a computer

Description automatically generated with medium confidence

Sign in and log in process

Diagram

Description automatically generated

Schematic

Description automatically generated with low confidence

Revision Format

Schematic

Description automatically generated with low confidence

Exam question Format

A screenshot of a computer

Description automatically generated with medium confidence

Different Views For Teachers

Graphical user interface, application, email

Description automatically generated

TxtUsername

- The user must input their username in this textbox in order to access this programme. The username is stored once completed on the computer system so it can be recognised later. At no time will the user be required to generate a new username, even if they shut down the computer and restart it.

txtUsername

* Again, this textbox like the username it will be the confidential to her access into his or her account. It will be saved and stored into the computer system. You will not be able to see password as it will be hidden in asterisks (\*) for confidential reasons.

btnLogIn

* If the user has created an account, once he or she has clicked the log in page it will take them to the main menu page where they will access what they would like to do. If not, they will have to make an account by clicking the hyperlink. Again, if the system does not recognise the credentials typed in it will pop up with “username or password credentials incorrect. Try again”.

lblSignUp

* Here is the link I have used to create an account if the user does not have one or if the user has forgotten the password

Graphical user interface, application

Description automatically generated

txtUsername

* This is where the user will create a username that will be saved onto the system so that when logged in it is recognised ever time when trying to log in

txtPassword

* Just like the username, the user will create a password that will be saved onto the system. Only difference is that it will be hidden in asterisks (\*) for confidential reasons.

btnCreate

* This button will create the log in after the username and password have Been filled in – so then this is then saved onto the system. A message is also popped up saying ‘Account created’

BtnClose

When the user is done creating an account, he or she will then click close and re enter it into the log in and get started on the revision app.

Graphical user interface, application

Description automatically generated

btnRevise

* This one of the revision app modes the user will be able to access. This button will then lead to another set options with a selection of topics the user will want to revise from. These buttons have been used to separate each of the options to show clear differences.

btnExamQ

* Like Revise this button will link to again a selection of topics. In the topic chosen by the user there will be an exam question on the topic – prior before this the user will have contextual knowledge from taught lessons or self-revision using the app.

btnQuiz

* This button will also lead to another page with a range topic - difference is with a range of multiple choice-based questions.

lblProfile

* By creating this link this will allow the user to access his or her account. They will be able to see their personal details, score, completed revision marks. The scores will unfortunately have to be typed in manually but will be saved.

lblLogOut

* This link allows the user to log out of their account. I used a link rather than a button as it would take less space. Also after reviewing similar type revision apps, I think the link would look more aesthetically pleasing

lblMainMenu

* This link will allow the user to go back to the menu to access their other options of revision within the app.

BtnNature

* All of the buttons displayed carry out the same process. There will be tab page linked each button which will present content. The information I will put in will follow the spec as we learnt. X

A screenshot of a computer

Description automatically generated

lblHome

This link will allow to go back to the previous page like the button home, logout, and my profile. This will be less hassle to code and make as they take less space and 2 lines of code.

**BtnNature**

Each of the buttons will have the same purpose in leading to a page of information/subject content. Each button has different content linked with the topic area name e.g. Components of a Computer will show content of the different hardware and software parts. To use less storage, I decided that use buttons to link to tab pages. As the user can simply navigate between the form pages to access the various modes, it also gives the impression that the system is easy to use.

# C. Developing the coded solution

I am going to be breaking this development down into different prototypes. This is to double check and mark off pillars within the development whilst building a better structured plan to go about the development. This should allow me to carefully assess whether some features will need to be added to the program or will not be needed to. Either it will fit and will not need improvements or require an alternative simpler and easier to understand.

1. Create all the forms – make sure forms are functioning properly
2. Create a quiz algorithm – set up a quiz algorithm to recognise a users weaknesses and strengths
3. Create a tracking system for student progress – create a database and algorithm to monitor students’ progress
4. Create an award system – create an algorithm to award points for certain questions and gathered points
5. Create a reminder system – create a programme to be set to allow students to remind themselves.

# D. Evaluation

# Project Appendixes

Insert as many project appendixes as you need for your project.

These might include, but are not limited to:

* Complete Code Listing (ESSENTIAL)
* Interview Transcripts
* Meeting notes
* Observation notes or questionnaires