Hayder Nawaz – GeoQuiz Loreto Sixth Form College – practical coursework

**Contents**

[Analysis 1](#_Toc139383241)

[1.1 Problem Identification 1](#_Toc139383242)

[Background to the problem 1](#_Toc139383243)

[Computational approaches 2](#_Toc139383244)

[1.2 Stakeholders 3](#_Toc139383245)

[1.3 Research the Problem 5](#_Toc139383246)

[1.3a Existing Solutions and Stakeholders 5](#_Toc139383247)

[1.3b Essential features 8](#_Toc139383248)

[1.3c Limitations 8](#_Toc139383249)

[1.4 Proposed Solution 8](#_Toc139383250)

[1.4a Requirements 8](#_Toc139383251)

[1.4b Success Criteria 9](#_Toc139383252)

[2.1 Decompose the problem 9](#_Toc139383253)

[2.2 Describe the solution 9](#_Toc139383254)

[2.2a Structure of the solution 9](#_Toc139383255)

[2.2b Algorithm design 9](#_Toc139383256)

[2.2c Usability features 9](#_Toc139383257)

[2.2d Key variables and data structures 9](#_Toc139383258)

[2.2e Validation 9](#_Toc139383259)

[2.3a Testing iterative development 9](#_Toc139383260)

# Analysis

## 1.1 Problem Identification

### Background to the problem

The game I’m developing for my coursework is essentially a revision platform for GCSE geography due to geography having one of the largest specifications to revise from. This game will be called GeoQuiz. It is inspired by Kahoot which can be seen through some aspects of the game i.e., the quiz format, however kahoot has some issues which can be improved upon such as it not having enough educational benefits whereas GeoQuiz will have features where it tells you where you need to improve on and provide questions to do so. Other issues that Kahoot entails is that there is a timer, for a student to correctly answer a question he/she needs to take as much time as he/she needs now so then they can correctly answer when they are in an exam. Furthermore, we will improve on some functions of kahoot such as adding a sign up/login system for teachers and students where teachers can set quizzes and view student quizzes scores.

#CHANGE PROTOTYPE, THIS WILL BE NEEDEED FOR DEVELOPMENT SECTION

GeoQuiz will have 3 prototypes;

First: Its objective will be to have a start page, subject selector page and a quiz page. The main goal of this prototype is to have a working quiz where we call on questions and answers through a file handling operation.

Second: Now we focus on creating a score board where we can tell the users where they got their answers incorrect and where they need more practice on. This scoreboard will be displayed as a table with different columns such as percentage, incorrect answers, scores for each topic and past scores etc.

Third: Finally, we focus on creating a sign up and login system where there will be 2 login platforms; 1 for teachers and 1 for students. The student can only track his progress and answer the quizzes assigned to them by teachers, Furthermore the teacher would be able to see the students scored and assign tasks based on the student’s weak areas. Additionally, the third prototype will also aim to have adaptive questions. What I mean by this is the difficulty of the questions will change depending on how the player is doing. For example, if the player consecutively answers the question correct then the difficulty will increase; on the other-hand if the player is incorrectly answering the question then the difficulty will decrease.

### Computational approaches

Thinking abstractly – A technique in which excessive details are removed to arrive at a representation of a problem that consists of only the key features. Another definition is abstraction is the process of separating ideas from particular instances/reality.

Abstraction will be applied to GeoQuiz through Kahoot. What I mean by this is that I will abstract unnecessary details from kahoot and then apply the important details to GeoQuiz. An example can be that kahoot has many unnecessary details which I’m not going to implement such as the points system. Furthermore, for GeoQuiz itself, an unnecessary detail is the question box itself due to it not being a clickable button we don’t have to add a program to make the question box a choice for the correct answer.

Benefits of abstraction can include that it simplifies the idea, saves time, can include it being easier to remember syntax in high-level languages as it is closer to natural language and coding becomes accessible to beginners.

Thinking ahead – Taking into account what may happen in the future based on what’s happening now. It also means less time is spent waiting for instructions to be fetched. The purpose of thinking ahead is to make programs easy and intuitive for users to use.

With thinking ahead, we can take things from kahoot in order to save time and have a lower chance of error. For example, we can take the basis of having a multiple-choice quiz of 4 answers or we can even take the leaderboard concept but instead of having a leaderboard we can have a scoreboard.

Thinking ahead has a lot of benefits such as developers can build programs that are easy and intuitive to use. Furthermore, it can save a lot of time as a whole.

Thinking procedurally – A disciplined method of thinking in sequence, in order and logically (like a flowchart). It also makes the task of writing a program a lot simpler by breaking a problem down into smaller parts which are easier to understand and consequently, easier to design. A huge part of thinking procedurally is decomposition which is essentially the idea of breaking down a problem into smaller pieces.

Here Is an order of how my code should be written thinking procedurally:

Create a sign in/log in system > create a main page to pick subjects to revise from > create a quiz page to answer quiz questions from > create a review page which highlights the user’s weak areas. This is an overall simplified version of a flow chart to think procedurally. An OOP system will be used through the whole program but for 1 example let’s use the login system. The login system will have multiple stages. E.g., Creating a user interface for the user to input their name and password, then we use an if function to decide whether if the login was a teacher or student, then when they login it should output whatever, it’s supposed to output whether you’re a teacher or student.

The benefits of thinking procedurally can include it being much easier to think in OOP terms but when comes the time to code some little specific parts you are much more productive with just procedure programming. Furthermore, it makes the task of writing a program a lot simpler by breaking a problem down into smaller parts which are easier to understand.

Thinking logically – Thinking in a way which involves identifying where decisions need to be made by the user within the program, and planning out the outcomes of the decision made.

Thinking logically helps our program by making problem solving capabilities easier and makes complex parts of our program easier to understand. This will help when students need to look at the code to understand it. Moreover, due to making the problems to solve easier it requires less power/maintenance due to it being ran on school laptops.

There is a large range of benefits of thinking logically. Some include helping you to prepare for different scenarios and can help you think more critically.

Thinking concurrently – The process of completing more than one task at any given time.

It (thinking concurrently), helps your program by allowing you to spot patterns and parts of problems where concurrency can be applied, thus speeding up processes.

Benefits of thinking concurrently is that it makes more efficient use of processor time, increased program throughput and reduces time being wasted.

## 1.2 Stakeholders

Overall, my main stakeholders are GCSE geography students, these students are the prime beneficiaries to the program. This quiz serves as a tool for improving the student’s understandings of the geographical concepts, preparing them for GCSEs. It offers an engaging and interactive platform for students to test their knowledge and to gain new knowledge and ultimately achieve better results. Additionally, the performance tracking features of GeoQuiz are tailored to help students monitor their progress.

My 3 stakeholders are;

-Illyase/ A GCSE geography student – These are the people who are revising geography in order to achieve higher grades in their GCSEs. Illyase is a hardworking, high-achieving, ambitious student with good grades who’s looking for a new platform to study his geography. Illyase can help me understand if the questions are too difficult or easy or if the User Interface is too bland or too overwhelming.

-Maimoon/ A GCSE Geography head of department – These are not necessarily the same as geography teachers as they only need to see progress and scores so they know whether to make improvements to the staff or any other methods of teaching if the progress is poor. Maimoon is a head of geography department at the esteemed loreto college who needs an easily accessible database which holds his students results in geography HomeWorks and their exams results. Maimoon can tell me about the ease of reviewing the questions in case it’s too difficult to understand.

-Mr Shiels/ GCSE Geography teacher – These are the people who assign work for the students but don’t do the work themselves, this is so the students can work on their weak points which are reviewed by the geography teachers. These teachers actually teach the knowledge of geography so if the students get unfortunate grades, then the teachers will be looked down upon which is why it is their priority that their students attempt to get the highest grade that they can get. Mr shiels needs to set his students homework where he can customise the difficulty of the questions and set a timer for how long the students should take on the questions. Mr shiels can assist me in reviewing the questions to judge if they are fit for a GCSE standard of questions.

GCSE geography student (GS) conversation:

me; “Would you like to get your answer result after you answer the question or after the end of the quiz?”

GS; “Preferably at the end of the quiz, furthermore…”

GS; “…How is it going to help me with my exams?”

me; “It’s going to help you memorise content and exam technique for writing answers.”

GS; “Is there 6 or above, mark questions implemented into this quiz?”

me; “No because this game is to help students memorise their content not how to answer questions.”

GS; “But with exam strategy questions students are more likely to obtain higher marks in the exam.”

Me; “I will take that into account, thank you very much.”

GS; “Can you also make it so there is a timer and question number?”

Me; “Like Kahoot?”

GS; “Exactly.”

Me; “Yes I can envelop that feature.”

Geography head of departments (GH) conversation:

me; “What type of data do you want to see from a student?”

GH; “I want to see scores from the tests and which subjects they lost marks in.”

me; “In what format would you prefer your students work be displayed to you?”

GH; “In a table format.”

me; “But wouldn’t be too big thus leaving a lot of excess room from empty rows, etc.”

GH; “No because other rows can be filled with useful information such as the student’s weak areas.”

me; “So you prefer add tables apart from scores?”

GH; “Yes that would be very ideal.”

Geography teacher (GT) conversation:

me; “Should I consider including 6-mark questions in my large range of question types in the quiz for the students?”

GT; “No, the students need a platform where they can focus on memorising content and at most answer 1–3-mark questions.”

me; “Many of the students I asked seem to share the same opinion. Furthermore…”

me; “…Should there be a difficulty section in terms of the questions?”

GT; “No because this app is helped to retain memory from knowledge learnt in lesson, If a student were to get set a hard task by me then they will get flustered and forget.”

me; “So focus more on memory retaining information?”

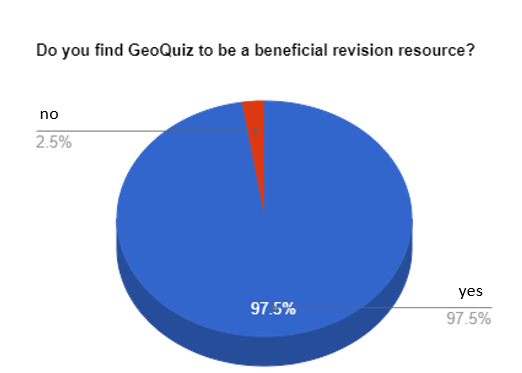
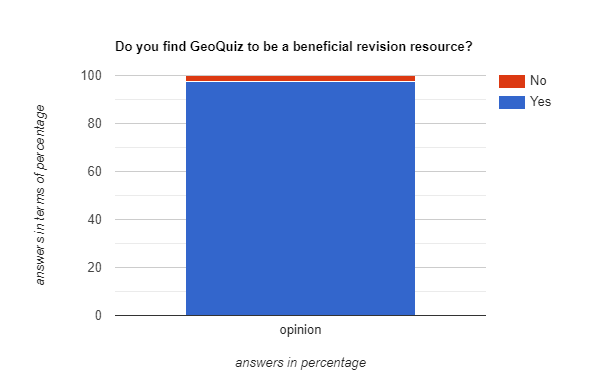
GT; “Yes it’s a key aspect if a student wants good grades.”

me; “Thank you for your time.”

After conversation with the stakeholders a couple aspects of the game will be altered. For example, before I only had content revision questions but GS has convinced me to add more exam answering style questions. Furthermore, GH has introduced the idea of adding more than 1 table of exam answers such as a weak area row or even a strong area row. Finally, GT has suggested not to go with difficulty of question feature. This is due to the main objective of GeoQuiz is to memorise and testing students is not an effective way to memorise.

Public feedback section:

We asked a multitude of geography students who used GeoQuiz if they find it beneficial or not and here are the results in 2 different formats;



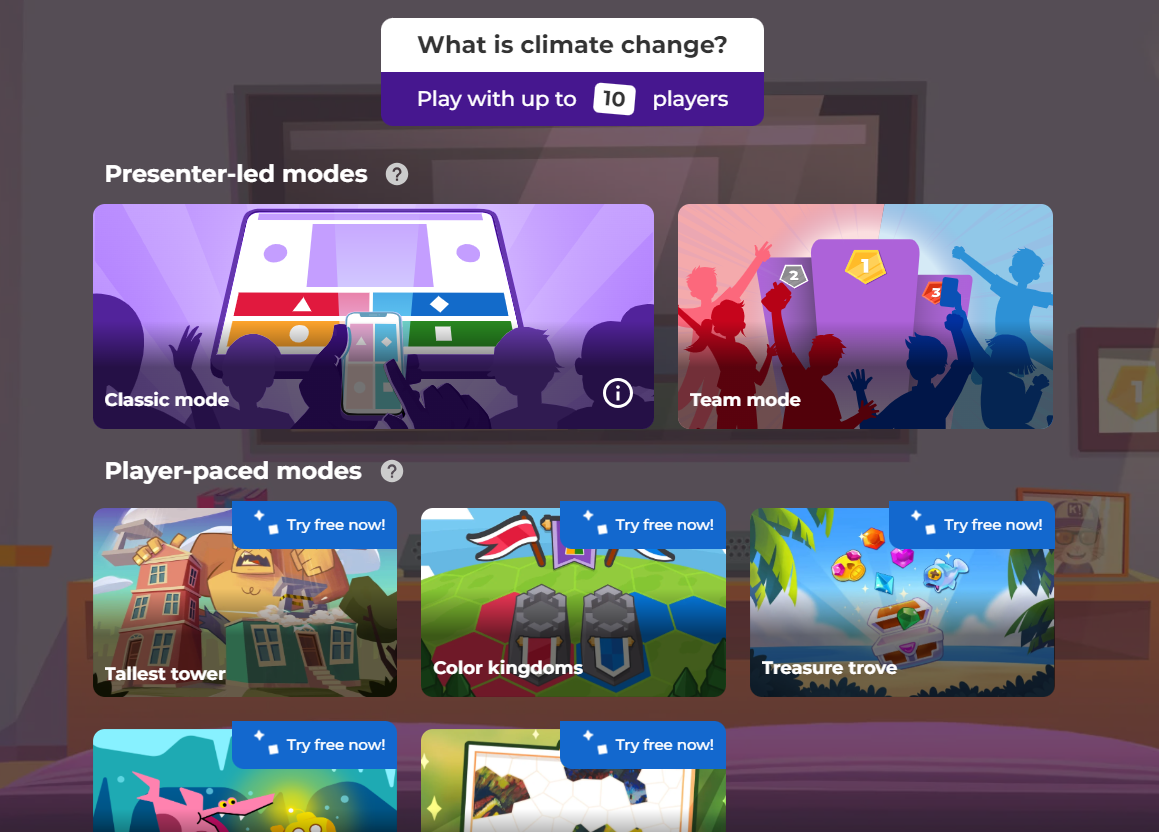
## 1.3 Research the Problem

### 1.3a Existing Solutions and Stakeholders

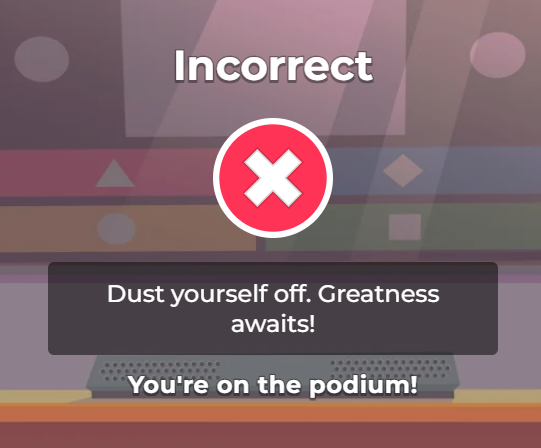
There are 2 existing solutions in where GeoQuiz takes a lot of inspiration from. These solutions are Kahoot and SmartRevise. There a lot of advantages and disadvantages of these learning platforms and unfortunately there are some features I cannot implement into my coding due to certain limitations such as not being able to play online etc.

All the advantages/disadvantages/key features:

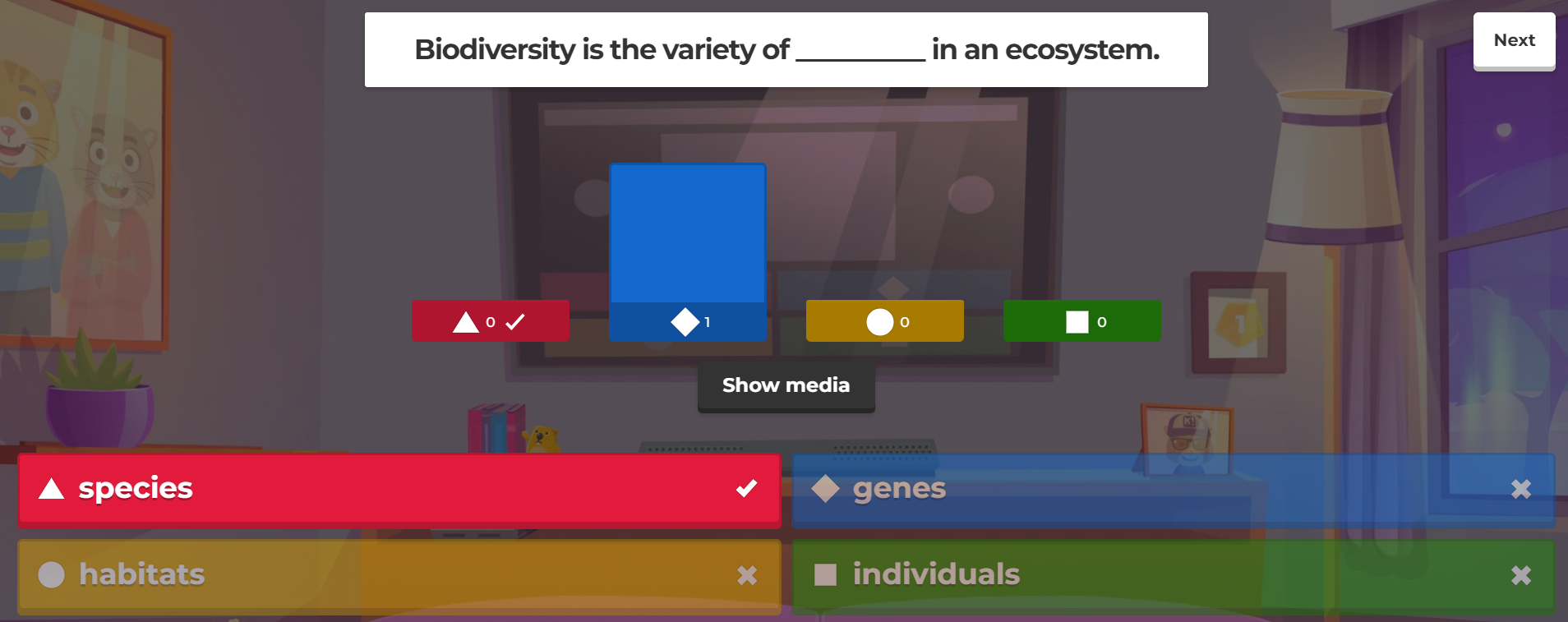
- Here we can see that there’s many game modes for the user to play from in kahoot. This can prove to make the user have a more enjoyable experience; However, there is a limitation which is that it’s with other players which is a feature we can’t integrate in our game.



- Here kahoot tells us if we are incorrect straight away, one of my stakeholders (the student), said they would prefer the total score being told at the end- due to this my program is going to display the results at the end of the program. However, we can envelop a positive message being shown whether the student answered correctly or incorrectly.



- Here kahoot shows a leaderboard of results just for 1 question which is something I can’t envelop into my game dues to the stakeholders requesting the results being shown when the quiz ends. Furthermore, kahoot really distinguishes the different players with each player having their own distinct colour however this is another limitation for our program as it won’t enable the feature of having 2 or more players. Moreover, it tells you what the correct answer is and what everyone else picked. GeoQuiz will learn from this by outputting the correct answer when a question is incorrectly answered but we will not display others answers simply because there is no other people.



- Here it displays the users name as well as their score. We are going a different route in where we are going to display the total points at the end of the quiz and we won’t have the name inputted displayed due to when creating the login system, the user already created their name which wont be allowed to change



- Here the timer and question number is displayed when answering the question which is a smart feature I will try to envelop into my quiz so the user will know how many questions are left and how long they will have to do it.

- Here is finally a feature from SmartRevise in where the student picks between multiple courses. Our program will unfortunately only be able to involve 1 due to the large amount of work being put into the geography content, it will be unlikely we can do the same for other courses.



- Here the user can see his homework, when it is due and the subject of choice. Our program will take the feature of displaying the deadline however it won’t be waiting for the teacher to mark as it will mark straight away. Furthermore, I will display the subject name and where its from in the specification.



### 1.3b Essential features

List of essential features in GeoQuiz;

-From what I gathered from the stakeholders I gathered a list of essential features for GeoQuiz and these are top priority. These are top priority to finish before the deadline;

+Settings page + This is so the user can change certain features of the game, e.g., sound.

+Contents page + This is so the user can choose the specific part of geography they want to do, using the specification.

+Answer page + We need a form where our users can actually answer questions in a multiple-choice format, furthermore the question should be displayed in the same form. Some other features in this form should include a timer and the question number {This was requested by a student}, etc.

+Login/sign up system + A login/sign up system will be necessary so the teacher can assign work to a student, furthermore multiple students.

+A efficient password recovery system + In case the user loses their password, it should be swift and easy to recover it.

+Scoreboard + At the end of each study session it should tell you the score you achieved.

+A database / file handling excel file. of questions and a way to query them + A list of questions should be available for the program to access and query so the user can answer them. This data bases has to have at least 30 questions for each topic.

+A excel file of the student’s past results + A list of results from the students attempts at quizzes the teacher set.

+A functional marking system + The system should be reliable when marking so it can efficiently output the score to the user later on.

+A functional score output system + The system should also be able to efficiently count the number of marks a student got wrong and the number he got right.

+A classroom code generator + This should display the teacher’s unique classroom code so a student can easily join the classroom.

### 1.3c Limitations

List of limitations in GeoQuiz;

-No multiplayer - due to this, there’s a lot of features we can’t include in our game such as a competitive 1v1 mode etc. Multiplayer games are more beneficial when it comes to studying as it makes it seem as its less studying and more of a actual game, furthermore it can create competitiveness between students driving them to achieve more.

-No leaderboard – due to there not being enough players because of no multiplayer; we can’t include a leaderboard so instead we created a score board for the students 5 best scores. This way the student can reflect on his scores and improve each time.

-No viewing individual questions if correct and incorrect after quiz is complete, however we can view the overall score to compensate for this issue. Another way we are planning to resolve this issue is by telling the user which specific subjects they would need to improve on.

-No multiple game modes- unlike kahoot we can’t incorporate multiple game modes due to the time it would take being too long to make. This would have been good because it would make GeoQuiz less repetitive in terms of the way you revise and answer questions.

-No animations- kahoot has animations for when you move from question to question and when you answer a question correct or incorrect. The reason we can’t implant this feature in our game is because it would need an artist or a team of artists. Even though the idea of animations seems small, it can completely change the tone of the game thus making it more light-hearted.

-No avatar- kahoot has a character that the user can pick from to display themselves when answering questions. Unfortunately, we can’t implement that due to the design of the characters would need an artist much like the same reason we can’t implement animations.

**Talk about your own limitations, time skill etc.**

## 1.4 Proposed Solution

### 1.4a Requirements

ADD EXTRA CODE CHANGES

Highlight based on different prototypes

* Login
  + Login as student
    - Taken to home screen for game
  + Login as teacher
    - Taken to teacher home screen
  + Forgot password
    - Enter email
      * Has to be a email that a user has already registered with
    - Send password to that email
      * Aswell as a message, e.g., “you dropped something, be careful not to lose it again!”
* Register
  + Enter username
  + Enter password
    - * Has to have a number and a upper case and lower case
  + Enter email
    - * Has to be a valid email
  + Choose whether you’re a teacher or student
  + Add new account to the data base
  + Then taken to their respective home screen
* Student Home screen
  + Start button
    - Given a list of every topic in GCSE specification for Geography
    - When they click on a topic, they get taken into the quiz page
  + Continue button
    - Continue the last quiz that they have attempted
  + Homework button
    - Displays in a table all the different homework that they got set.
  + Previous data
    - Displays previous scores and homework scores
      * In a table format
  + Join classroom button
    - Displays text box where student inputs the teacher’s unique classroom code to join their classroom
  + Settings button
    - Sound on/off
    - Change password
      * Asks previous password twice and to input new password twice.
* Teacher Home screen
  + Set homework button
    - Pick which students to set homework to
      * From specification presented to them
    - Pick a due date for the homework to be completed
      * Validation check so the date is not set before the day it is set
    - Given a list of every topic in GCSE specification for Geography
    - Picks a topic to set
    - Picks the number of questions
      * Range check between 15 to 100 questions
    - Pick the time set for all the questions to be answered
    - Set/Continue button which takes them back to the home screen
  + Student’s data button
    - Choose student to display data
    - Display the chosen student’s data in table format
  + Unique classroom code
    - Generates 5-character unique code which a student should be able to use to join the teacher’s classroom
    - Displays unique classroom code so student can join
  + Settings button
    - Sound
    - Change password
* Quiz page
  + - Display instructions with a 10 second timer that starts the quiz when hits 0
      * Instructions and rules of the quiz will be displayed
    - Question displayed at the top
    - Timer displayed at the top
    - Question number next to timer
    - 4 boxes to choose answer from
      * Will not display A B C or D but rather the answers for question being asked
    - Once box is clicked a continue button will be displayed
    - Once end is reached, display score with message based on the score with button below it which reads “Review Answers”
    - Store overall score in database

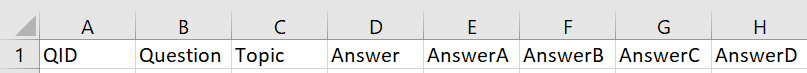
Software requirements;

ADD EXTRA REQUIREMENTS WE ADDED, E.G. NUGGET PACKAGES

C sharp > this software has to have the ability to read and execute the program written in C sharp. We are using C sharp as its easy to interpret and has an efficient debugging system the system finds errors in our code and fixes it.

Database > the software needs to have a suitable app to create the database needed for the geography questions and the different account, for instance excel.

For example, this is the format we need for the databases.



Libraries > libraries are chunks of preprogrammed code (E.g., “click” from a form because its already coded to click we don’t code it). This is an essential piece of software as it makes coding much simpler and easier and, in a way, more efficient.

Visual Studios > This is the IDE (integrated development environment) in where my program be programmed in. It’s basically the blueprint in where my program will be written on.

64-bit Windows (OS) operating system (or any after Windows 7 SP1) > Visual Studio runs on Windows so therefore runs on a Windows operating systems. A 64-bit operating system is recommended by their website but any higher would be even better.

Hardware requirements;

A storage device, hard disk, with a minimum of 25 Gigabytes of free space > This is so visual studios can be downloaded but the amount of space depends on the type of visual studios that can be downloaded.

A minimum of 3 Gigabytes of RAM > This is the minimum amount to run visual studios so our code can run efficiently.

A processor with a speed of 1.8 GHz or faster > This is so the program won’t take long to load and function and so it can run without crashing.

Battery > a battery is required to power the computer but it doesn’t have to be a large battery as the program is going to be run of school laptops due to mainly students utilising this app.

Mouse > to interact with the screen when, for example, clicking question answers.

Keyboard > to interact with the screen when, for example, registering your account or logging in.

Monitor > to view the questions being asked

An assembler > to translate the code to a language that the computer will understand

A CPU > to allow FDE cycles to be carried out multiple times.

### 1.4b Success Criteria

|  |  |  |
| --- | --- | --- |
| Objective | Justification | How evidence will be provided |
| An initial home page or menu with intuitive options to choose to login/signup or use the GeoQuiz program. | Student and teacher each require different programs and these must be easy to navigate. | Screenshot of home page – showing all options. Stakeholder opinion. |
| Allows teachers and students to make an account. | Accounts ensure progress is tracked and scores are saved between users. Their login details should also be saved in a database where it can be called upon when logging in. | Screenshot of signup and accounts created. |
| Provides an error message and makes the user repick if a username is taken. | If 2 users had the same username, scores could get combined and access may be granted to the wrong account. | Screenshot of trying to sign up with a username already in use with a message asking to input another username. |
| Must check that password is 8-20 characters long and contains a number and an uppercase letter. | Ensures a good level of security on the accounts, reduces chances of unauthorised access. | Screenshot of trying to sign up with a password too long, too short and in between, with and without numbers and with and without a uppercase letter. |
| Allows users to log in to their account, only granting access with the correct details. | Login means only authorised personnel have access to the program, and also identifies the user so that their scores, difficulty levels can be tracked and weak and strong areas can be tracked. | Screenshots of logging in with correct details, then with incorrect username and incorrect password. |
| Numbers for questions randomly generated and inserted into set format. | Saves teacher and me time, ensures infinite number of potential questions. This also ensures a repetitive element is NOT implanted in GeoQuiz. | Screenshot of this function coded. |
| Answers calculated by computer for multiple choice questions. | Saves time so the teacher would not have to mark the work, furthermore it helps give the student their scores straight away. | Screenshot of quiz highlighting the correct box in multiple choice green and the incorrect box red. |
| Teacher can delete student accounts. | This is necessary to ensure the records regarding quiz results are up to date, and so that the tutor can remove old students so less space will be taken to hold the user’s details in the database for users and their details. | Screenshot of students file before and after deletion. |
| Teachers should be able to set homework for students where they can choose a time limit and topic. | This is so students can do practice knowledge they learnt in an exam style format so they can more efficiently remember the knowledge they learnt. | Screenshot of the teacher’s page setting the homework as well as a screenshot of the student’s to-do section showing the homework the teacher set. |
| Questions begin easy and get harder depending on student ability, which the quiz will learn throughout. In other words, the quiz has to have adaptive questions. | Adaptive quizzes are proven to be highly effective for practicing skills. | Screenshots of varying scenarios and the questions the quiz outputs. Difficulty can be measured based on the type (negative, fraction) and range the numbers falls into. Exam questions are most difficult so should only show if user is doing well. |
| Easy to use, simple and bright user interface. | An easier, more effective interface can result in a more positive mindset to work. Furthermore, it would not seem so much as work rather than a game. Students typically respond better to this -as they can focus just on the geography. The “difficult to use” aspect was a critique of a revision resource a student had previously used in the survey | Screenshots of the interface background as well as fonts and buttons. We will also input screenshots of stakeholders opinions of the user interface. |
| Clear/understandable questions. | The questions must be suitable for a GCSE student and shouldn’t be displayed in a way that confuses the student. It should be simple that there is a question and 4 answers you can choose from. | Screenshots of how the questions are displayed. |
| Simple to use answer boxes for the questions. | The answers should have a premade template that are boxes which the user can select. This makes marking much easier as there are no variations in the format of the input, meaning only one answer is correct as there is only 1 box which is the correct answer. | Screenshot of answer boxes. |
| Questions automatically marked accurately. | This saves time and gives immediate feedback which students wanted according to the survey. | Screenshot of questions being marked by highlighting the correct box green and the incorrect box red. |
| Final feedback for students. | The tutor and students both highlighted this in the survey and stakeholder requirements, and I found it was slightly lacking in a current solution. This will also help benefit the student more as he will know where to work on. | Screenshots of different feedback depending on result and where to improve and where they are already strong. |
|  |  |  |

## 2.1 Decompose the problem

DECOMPOSITION – SAY HOW CODE IS BROKEN UP, SAY HOW WE BROKE UP INTO DIFFERENT CLASSES AND INTERFACES AND METHODS AND JUSTIFY

Decomposition;

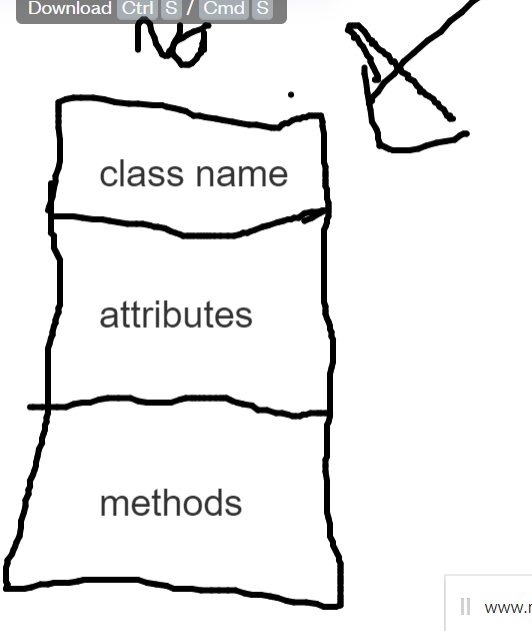
The breaking down of a complex problem into smaller, more manageable parts so that the solution can be more efficient and easier to find. I am using decomposition in a top-down diagram which displays each feature of the game and how to achieve that feature. A top-down design is the name given to breaking a problem down into increasingly smaller and smaller manageable parts (also known as decomposition).

This is my top-down diagram for geo quiz;

## 2.2 Describe the solution

#STATE EVERY CLASS I HAVE, WITHIN EACH CLASS STATE WHAT METHODS I HAVE- FOR EACH METHOD EXPLAIN WHAT IT DOES. STATE ANY ALGORITHMS IVE USED E.G. SORTING AND MERGING ALGORITHMS. STATE WHAT VARIABLES I HAVE AND WHAT DATA STRUCTURES IVE USED. STATE WHENEVER VALIDATION IS USED AND HOW IT WORKS. TALK ABOUT WHAT FEATURES I HAVE THAT MAKE THE FEATURES USABLE, E.G. BIG BUTTONS EASY TO CLICK.

### 2.2a Structure of the solution

Need to have class diagrams which have class name, attribute and methods. Looks like 

### 

### 2.2b Algorithm design

-Explain different parts of code like a flowchart, e.g.,

### 

### 2.2c Usability features

-Talk about/annotate only the forms we made that are in the program, not code.

### 

### 2.2d Key variables and data structures

-Write about all the fields using databases and data types and the purpose of each record. E.g., length check

-Talk about all the variables used in the program. Talk about every method we used.

### 

### 2.2e Validation

Where are we going to use validation and error checking. Talk about each error and validation and how we are going to fix it.

## 

## 2.3a Testing iterative development

How are we going to test all our inputs to make sure we have the correct outputs. Need to decide what type of data we are going to use to test. Test as you go along.

#SAY WHAT TEST DATA WE ARE GOING TO USE AT EACH STAGE OF ITERATION.JUSTIFY WHY WE USED THE TEST DATA CHOSEN. STORE RESULTS OF TESTING IN A TABLE

TABLE HEADINGS;

TEST NUMBER ][ TEST DATA ][ TEST DESCRIPTION, WHY WERE TESTING SOME CODE [] EXPECTED RESULTS []

#IN EACH PROTOTYPE YOU NEED TO SHOW SCREENSHOTS OF ALL THE CODE IVE ADDED IN THE PROTOTYPE AND WITH EACH SCREENSHOT YOU WILL NEED TO ANNOTATE/(UNDERNEATH IT MAKE A PARAGRAPH AND ITS FUNCTIONS) IT MAKING SURE TO JUSTIFY ANY DECISIONS YOU MAKE

# 3.1A.B. Development – Prototype 1

A IS ANNOTATING EVIDENCE OF CODE AND B IS JUSTIFYING DECISIONS IN THE CODE.

*Introduction, that describes the objective of this version and links back to decomposition.*

Version Objectives for Prototype 1:

* Allow user to select a type of quiz; custom or complex

If custom is picked:

* Allow user to select a sub category from the geography specification
* Allow user to select a time and number of Questions and Start Quiz
* Start timer
* Get questions from the file and display the questions depending on the sub category picked
* Check answer and update score
* Allow questions to be adaptable in difficulty
* Go on to next question and Quiz
* Display score

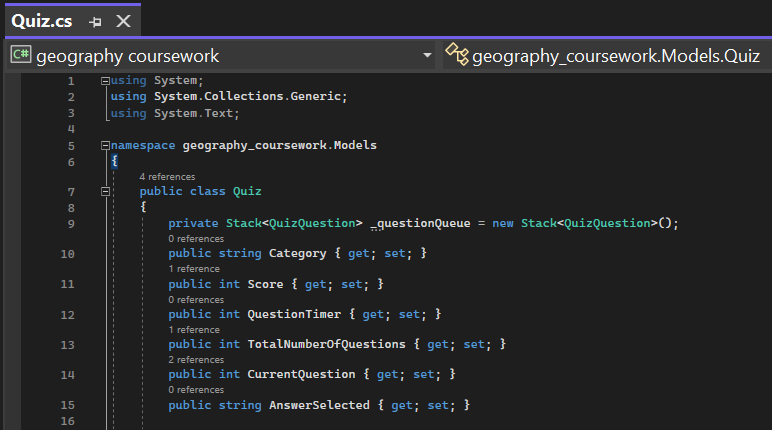
## Stage 1 - Give user a choice of doing a custom quiz or complex quiz

If Custom Quiz is picked:

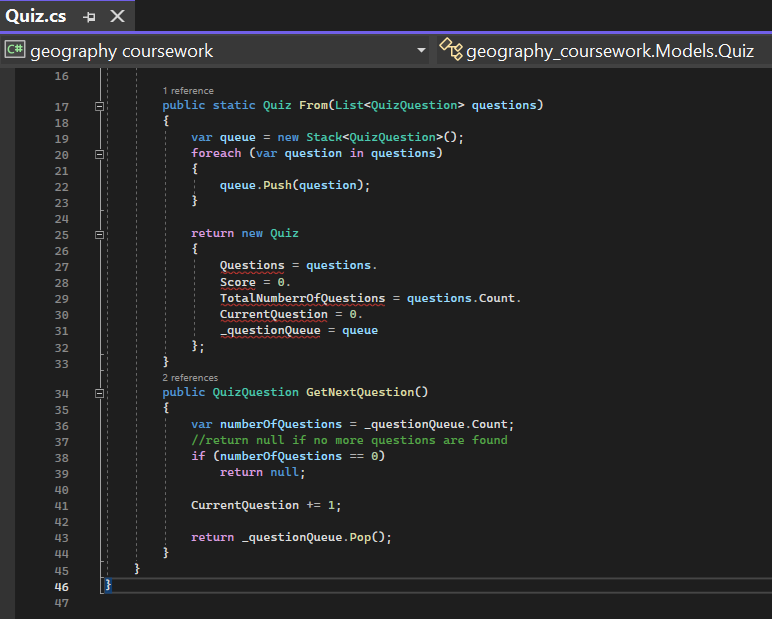
## Stage 1- Allow user to select a subject

## Stage 2 - Allow user to select a time and no of Questions and Start Quiz

## Stage 3 - Get questions from the file



## Stage 4 - Select and display question and answers

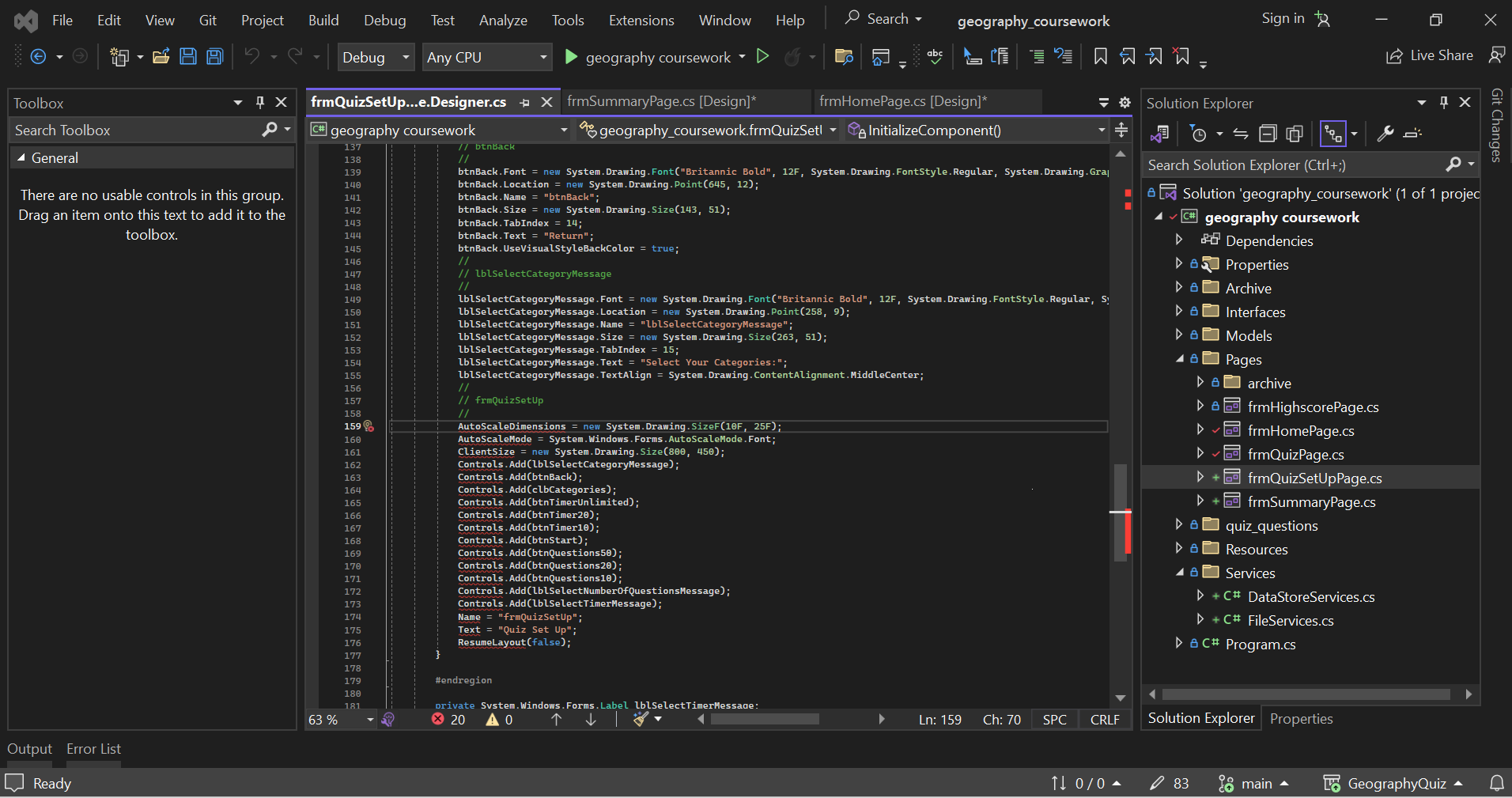


Here we use the push and pop method to output our geography questions from the csv file.

## Stage 5 - Check answer and update score

## Stage 6 - Go on to next question and Quiz

## Stage 7 - Display score and parts where improvement needed



3.2.A TESTING TABLE AGAIN AND USE TESTING NUMBER FOR 1 HEADING AND NOW INCLUDE ANOTHER HEADING FOR RESULTS.

3.2.B MAKE ANOTHER TABLE OF FAILED CODE, MAKE A FEW ERRORS AND FIX THEM 5 ROUGHLY

4.1.A QUICK PARAGRAPH/CONCLUSION – TALK ABOUT HOW WE TESTED THE SOLUTION AND HOW ROBUST THE CODE IS NOW. ALSO TALK ABOUT HOW WE FAILED SOME TESTS AND WENT BACK AND FIXED THEM AND HOW WE FIXED FAILED DESIGN.

4.1.B TALK ABOUT USER FEEDBACK AFTER THEY USED OUR FEEDBACK – LINK TO STAKEHOLDERS INCLUDE INTERVIEWS. GIVE STUDENTS AND TEACHERS APPLICATION TO TRY AND MENTIONS WHAT THEY LIKED AND DISLIKED ABOUT IT AND ADD GRAPHS MENTIONING HOW MANY PEOPLE FIND IT GOOD, ETC.

4.2 LOOK AT ANALYSIS AND COMPARE IT TO FINAL PRODUCT AND MENTION WHAT WE MANAGED TO DO AND WHAT WE HAVENT MANAGED TO DO AND IF WE HAVENT MANANGED THEN SAY WHY WE COULDN’T ACHIEVE THEM. ADD A LOT OF DETAILS THIS IS THE LARGEST SECTION.

4.3 TALK ABOUT FEATURES FROM HOW I MADE THE PROGRAM AND TALK ABOUT HOW EFFECTIVE IT IS FOR HOW WE ARE TRYING TO MAKE IT. WAFFLE A LOT

4.4.A TALK ABOUT HOW EASY IT IS TO MAINTAIN THE SOLUTION

4.4.B TALK ABOUT HOW FURTHER WE CAN IMPROVE THE SYSTEM, E.G. ADDING THE LOG IN SYSTEM