**Project Description**

The program I will be creating will consist of a quiz that is software-based integrating with a database. The quiz will have 3 levels and once the user completes one level, they will unlock the next. As I intend for my users to have accounts this will be important as I will store the user's progression in a database and retrieve account data upon request. When the user starts the program a main screen in which the user logs in or registers for an account or views the leader board will appear. There will also be extra functionality for a user to save their account and sign out. The quiz questions will consist of what the capitals of various countries are. The audience of my game will be any young adult regardless of what they do or have studied with common sense.

My Project is suitable to the specifications set by you as my main technology is software development, and my secondary technology will be database design and development. The software aspect will be processing the data such as account information and will deal with inputs and outputs. my database will integrate with the software to store this account information, my 2 advanced higher concepts are the following: Concept 1 will be procedural programming with an array of records and Concept 2 will be using bubble sort. For my software, I will be using Visual Basic as my coding language. I will be using and interacting with my database through the software. I will be using SQL to execute queries once I establish a connection to the database. The queries will be used to save the users level, read in the database table and register an account. I will use the bubble sort algorithm to filter through all the user’s accounts and find the highest-scoring users and if any users tie with the highest-scoring user they will be recognised. A user will not be able to play the game unless logged in, though they will be allowed to use the leaderboard feature.

**Scope**

**(while not in the specification I felt this was necessary)**

The scope of my project will be.

1. A completed design with pseudocode for the adv h concepts and the integration and a data dictionary along with the query design.

2. test plan with a persona tested

3. fully working program integrated with a database that will store user data

4. completed test data represented appropriately

5. detailed evaluation report focusing on concepts and functional requirements

**Constraints**

The constraints my project will have will be technical, time and economic, there will be no legal:

1. My knowledge and understanding of SQL will be limited so not having done it before I will need to adjust my time accordingly to learn it.
2. My final project will run on the Visual Studio 2022 test environment and will use Microsoft Access 2022 to run my database.
3. I will ensure that my project is done and handed into my Centre by the 28th of March.
4. As for technical constraints, I will have to use laptops funded by the school as a MacBook (the computer I have at home) does not support Microsoft access this will heavily impact development and will need to be accounted for in terms of time.
5. There will be no costs as all the software I will be using for the project will be free or licensed by the school.
6. There will be no legal constraints as my product is 100% original and does not use any produced material.

**Boundaries**

**(again while not in the specification I felt this was necessary)**

Once the project is complete it will contain:

1. An input validation system where the password the user enters in the sign-up phase is validated.
2. My quiz will consist of 3 levels which the user can select to start from after completion.
3. The user must not be able to play the quiz without logging in.
4. A sign-in system where the user enters their username and password, and it is checked if those are the same as any account info on the database.
5. A fully integrated project in which the database and the software work together to output the end game.
6. User account numbers must not exceed 10.

**UML Use Case Diagram**

A diagram of software application

Description automatically generated

**Requirements Specification**

Project purpose: The purpose of this application is for a user to simply play a game and be able to create an account to save the level the user has completed; this will be done by the user creating a username and password.

**Functional Requirements**

* The Software Application must establish a connection to the database before executing queries.
* The Software Application will need to read and write account information this will be done by using SQL to execute a query and import the data into the application upon login and registration and the leaderboard system.
* All User Inputs MUST be validated whether this is answers to questions or account information.
* The game must not be playable without logging in, login is necessary to play whether logging into an existing account or registration.
* Once a user inputs their account information this must be validated by checking the character counts for usernames and passwords are 8 characters.
* Upon registration of an account, the database connection must be opened, and the registration info must be written into the database.
* Once data is imported into the application a bubble sort algorithm will be used to find the highest-scoring users.
* A level must not be played unless that level or the previous level has been completed.
* When validated that the user information matches the login information the user must have the option to continue from the last level or play from the start.
* All user data must be stored in the database.
* When a user creates a new account, they must stay logged in with those registration details provided they have been validated.
* A user must be able to save an account once logged in.
* A user must only be able to sign out once logged in.

**End-User Requirements**

The end user requirements will be:

* Clutter-free and clear UI.
* An easy-to-navigate interface.
* Login systems requirements must be displayed.
* Simple fonts and sizes so that the user can interact easily.

**Inputs and outputs and processes**

The user inputs will be:

1. Username
2. Password
3. Whether they wish to log in or create an account or sign out or save account
4. Question Answers

The Software Applications processes will be:

1. Execute SQL queries.
2. Verify login information.
3. Save and create account information.
4. Connecting to the database
5. Validation of EVERY user input
6. Checking that the user’s answer to a level is the same as the answer
7. Retrieve the level after the user wishes to continue
8. Check a user’s level before allowing them to play it (this does not apply to level 1)

The Inputs for the application will be:

1. Username information when creating or saving an account.
2. Password information when creating or saving an account.
3. Level information when creating or saving an account.

The outputs for the application will be:

1. Consistent error messages throughout the whole program (whether a user is trying to do something they need to be logged in for or if the database connection fails)
2. A message saying the answer is correct or incorrect.
3. The question for the level they are at.
4. Login and create account prompts and confirmation of login.
5. The leaderboard.
6. Current logged-in username and level once logged in.

**Resources Required**

Creating my project will require me to have access to the following software:

* + Microsoft Word (latest) for documentation
  + Visual Studio (2022) for coding
  + Microsoft Access (2012) for the user database
  + Google Chrome (latest) for the UML use case diagram

**Feasibilities**

In terms of feasibilities, there will be little to no legal, economic, or technical feasibilities. As my project is an educational project and there will be no money made this should void legal issues. As my project will not be implemented on a large scale simply having a USB and a laptop is all I will need to carry this project out so there will be no economic feasibilities. As for time feasibility, I will be limited on how long I have for the project, and I will have to rigorously plan my time. I will show this by using the table below.

The table below goes off weekdays these days do not need to be during school hours as the whole project is self-study meaning no classes will be supplied by the school nor is there an advanced higher computing teacher. This allows me to work whenever (provided self-made deadlines are met) and I will be starting the project later and be more disciplined to complete it quicker because of this.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Main Task | Sub Task | Duration | Start of Task | End of task |
|  | Plan Project Idea and Read Specification | 1 day | 18th Jan | 18th Jan |
|  | Outline Problem and Identify Adv H Concepts and Integration | 0.5 days | 19th Jan | 19th Jan |
|  | Identify constraints | 0.5 days | 19th Jan | 19th Jan |
|  | Generate End -User Requirements Specification | 0.5 days | 22nd Jan | 22nd Jan |
| Analysis | Generate Functional Requirements  Specification | 0.5 days | 22nd Jan | Jan |
|  | Create UML Use Case Diagram | 1 day | 23rd Jan | 23rd Jan |
|  | Identify Resources and Timings and Identify Tasks | 1 day | 24th Jan | 24th Jan |
|  | Project Plan | 1 day | 25th Jan | 25th Jan |
|  | Design of Adv H Concepts Pseudocode | 5 days | 26th Jan | 1st Feb |
|  | Design of Integration (excluding queries) | 4 days | 2nd Feb | 7th Feb |
| Design | User interface design | 3 days | 8th Feb | 12th Feb |
|  | Learn about and design SQL queries | 3 days | 13th Feb | 15th Feb |
|  | Start Implementing leaderboard system and general methods along with record structure | 5 days | 16th Feb | 22nd Feb |
| Implementation | Implement database connections and create database table | 7 days | 23rd Feb | 4th Mar |
|  | Implemented UI | 3 days | 5th Mar | 7th Mar |
|  | Description of new Skills and or Knowledge | 1 days | 8th Mar | 8th Mar |
|  | Log of Ongoing Tests | This will be logged throughout implementation |  |  |
|  | Full detailed test plan | 3 days | 11th Mar | 13th Mar |
|  | Persona | 1 day | 14th Mar | 14th Mar |
| Testing | Evidence of test plan completed | 2 days | 15th Mar | 18th Mar |
|  | Document problems faced by testing | 3 days | 19th Mar | 21st Mar |
|  | Evaluation of fitness for purpose | 1 days | 22nd Mar | 22nd Mar |
| Evaluation | Evaluation of Robustness and Maintainability | 2 days | 25th Mar | 26th Mar |
| Gather Evidence | Simply print documents | 1 day | 27th Mar | 27th Mar |