# Data Model Information

## Quiz, Question and Answer models

For this application I have followed a relational database approach creating a PostgreSQL database. I have chosen this approach as a relational database gives me the ability to create meaningful information by joining tables and also eliminates data redundancy and easy to backup and disaster recovery.

I decided to use a SQL database as has its built-in language, Data Definition Language (DDL) that gives flexibility to make changes to the database even while the database is running and while queries are happening. I selected PostgreSQL as is open source and can reduce costs.

Due to the time frame to develop this project I decided to use autoincremented Integers to hold Id values for the Quiz, Question and Answers entities. I based my decision also on facilitating debugging and manual testing as Integer values will be easier to recognise at a glance than Guid values. For future development would be desired to modify these fields and assign them Guid values in order to give them a unique identifier to avoid collisions in the case that several databases needed to be merged.

|  |  |
| --- | --- |
| QuizId | Title |
| Integer – Primary Key | Varchar(100) |

* The Title has a maximum length of 100 characters as it is an optimum length for a title. When a user introduces a title for a quiz, this length attribute is subject of client-side and server-side validation before the title is inserted into the database.

|  |  |  |
| --- | --- | --- |
| QuestionId | QuestionText | QuizId |
| Integer – Primary Key | Varchar(150) | Integer – Foreign Key |

* The QuestionText has a maximum length of 150 characters as it is an optimum length for a question text. When a user introduces a text for a question, this length attribute is subject of client-side and server-side validation before the text is inserted into the database.
* The QuizId is used as a Foreign Key as a way of linking questions with the appropriate quiz.

|  |  |  |  |
| --- | --- | --- | --- |
| AnswerId | AnswerText | IsCorrect | QuestionId |
| Integer – Primary Key | Varchar(150) | Boolean(false) | Integer – Foreign Key |

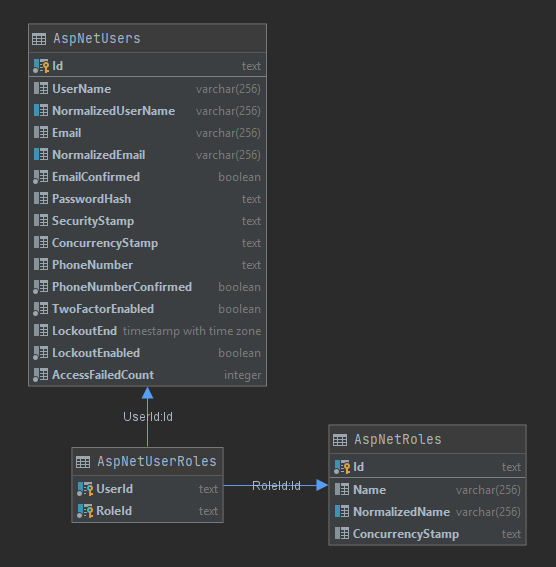
* The AnswerText has a maximum length of 150 characters as it is an optimum length for a answer text. When a user introduces a text for an answer, this length attribute is subject of client-side and server-side validation before the text is inserted into the database.
* The Is Correct property is a Boolean used to indicate which answer is correct. Its default value is set to false in order to avoid null values inside the database.
* The QuestionId is used as a Foreign Key as a way of linking answers with the appropriate question.

\*\* For all the tables, the Primary Key and the Foreign Key are indexes as a way to enhance database performance, finding and retrieving specific rows faster.

## ASP.NET Core Identity models

For user authentication and user management I have used the ASP.NET Core Identity framework as it gives classes and methods that facilitate the implementation of a login functionality, the storage of user data with security hashing for passwords and to manage user roles and claims.

For this project only few advantages of this framework have been used but could be fully implemented in future development for creating user claims or for holding more user data if it is decided so.

****As all the data models that ASP.NET Core Identity framework provides have not been used for this project, I follow to give details of the ones the application is making use of.