Windows Programming – PROG2120 Assignment #6
& Relational Databases Assignment #4

Submission in groups of up to 3

Introduction

Most applications today act as a suite of programs that meet a broad set of objectives. You will find that these suites often include a database, a service and some user interface components.

This is a survey tool with some gamified aspects. Two immediate applications could be a trivia game, or a way to prepare students for tests.

Requirements (Minimal)

User Experience

The user must provide a name. (No password is required.)

The user is presented with a set of questions (one at a time) and multiple choice answers, only one of which is correct.

Zero points are awarded for the wrong answer.

Points are awarded for the correct answer based on how fast the answer was provided. For example, the user may have 20 seconds to answer the question, and each second wasted drops the point value by 1. When 20 seconds is up, the question is considered incorrectly answered and zero points are awarded.

At the end of the questions, a score is provided with a leaderboard.

Correct answers are only provided when the game ends.

Many users can be running the application at one time.

Database

The database must support storage of the questions, potential answers and the correct answer for each question.

The database must store all answers to all questions. For the ones answered correctly, the length of time to answer is recorded.

The database must store the necessary information to generate a leaderboard.

Service

The service is a multi-threaded listener using an IPC protocol of your choice.

The service provides the questions, multiple-choice answers and correct answer to the client (user), as retrieved from the database.

It also collects the user’s answer (correct or not) and a score for any correct answers and stores this in the database.

Admin Experience

A single program provides the ability to:

- Edit questions and answers
- See the current status (either real time or updated at an interval of no more than 5 seconds) of all live participants
- View the leaderboard
- Using Office Automation, create an Excel spreadsheet with the question number, question text, average time to answer correctly and percentage of answering the question correctly.
  In addition, a histogram should be created to show the average length of time needed to answer each question correctly.

Other comments

You have a lot of flexibility in building this application. As such, it will be VERY easy to spend more time than necessary on this project. Please keep it simple, yet flexible. Here are some things to consider:

- Limit the number of questions to 10 for any game. You do not have to implement the possibility of multiple games. In other words, only 10 questions need to be stored in the database.
- Only support multiple-choice answers, and the choices for answers should always be 4.
- Some classes may be used in more than one program.
- You must use a MySql database
- You should use the .NET connector for MySql and write the applications in C#
- Variable connection information must NOT be hard coded. You may use command line parameters, prompts or a configuration file to get the necessary data.
- The system must work on multiple computers in the 2A213 lab. You will be demonstrating yoru solution.
- Use good object oriented programming principles.

Hand in

- All in one zip folder:
  - All solutions
  - A document with installation and usage instructions
- Submissions must be made to both Dropboxes – one for Windows Programming and one for Relational Databases.

NM