

Taller de Programación 2017

Trabajo Práctico Final

Introducción

El trabajo práctico final consta de la realización de una aplicación de cuestionarios.

La aplicación deberá ser capaz de presentar al usuario diferentes preguntas con opciones de múltiples respuesta o multiple choice, donde el usuario deberá seleccionar una respuesta para continuar con la próxima pregunta. Al final del cuestionario se le mostrará el puntaje obtenido en base a diferentes conceptos, por ejemplo cantidad de respuestas correctas, tiempo insumido, etc.

Requerimientos funcionales

Funcionalidad administrativa

El objetivo de la funcionalidad administrativa es persistir internamente en la aplicación los diferentes conjuntos de preguntas/respuestas a ser utilizados en la funcionalidad operativa. Un conjunto es una agrupación de preguntas/respuestas con el mismo origen. Las preguntas deberán pertenecer a una categoría (por ejemplo deportes, artes, etc.), tener niveles de dificultad (alto, medio, bajo) y múltiples respuestas asociadas.

En la aplicación podrían haber muchos conjuntos, pero el único requerido es el conjunto de preguntas del sitio **opentdb**

Opentdb (Open Trivia DataBase - <https://opentdb.com/>) es un sitio de internet el cual posee datos de diferentes trivias las que pueden ser inspeccionadas y también modificadas en forma on-line.

Este sitio permite además descargar las trivias mediante una API de fácil utilización devolviendo los datos en formato JSON (JavaScript Object Notation). A continuación se encuentra el link de acceso a la documentación de la API:

https://opentdb.com/api_config.php

Utilizaremos esta funcionalidad con el fin de obtener un conjunto de trivias para ser incorporadas a la aplicación. Un ejemplo de cómo acceder a los datos de éstas trivias desde una aplicación C# lo podemos encontrar en el siguiente repositorio de GitHub

<https://github.com/utn-frcu-isi-tdp/opentdb-test-client>

Un usuario administrativo de la aplicación deberá poder realizar importación de datos de las trivias disponibles, persistiéndolas internamente y asociándolas al conjunto de preguntas de Opentdb.

Como los datos de este sitio se encuentran en constante cambio, una vez realizada la importación se debe poder actualizar el set de preguntas cuando el usuario lo requiera. La única limitación a este proceso que debe implementarse es que una vez importada una pregunta no puede ser modificada, ni la pregunta en sí, ni sus respuestas.

Funcionalidad operativa

La funcionalidad operativa tiene como objetivo mostrar las preguntas al usuario e ir registrando las respuestas.

Un usuario operativo al iniciar la interacción con la aplicación deberá identificarse o seleccionarse para que de esa manera la sesión de respuestas esté relacionada con él mismo.

El usuario deberá poder seleccionar el conjunto de preguntas que querrá utilizar, si solamente existe un conjunto, se seleccionará automáticamente el existente.

A continuación el usuario deberá :

- Seleccionar la categoría de las preguntas.
- Elegir el nivel de dificultad de las preguntas.
- Seleccionar la cantidad de preguntas a contestar, que como mínimo deberán ser 10.

Una vez configurada la sesión el usuario podrá iniciar el cuestionario, al hacerlo se le presentarán secuencialmente preguntas a ser respondidas. Estas preguntas el sistema las seleccionará **aleatoriamente** de las persistidas internamente en base al conjunto de preguntas seleccionado, a la categoría elegida, el nivel y la cantidad de preguntas.

Al usuario al comenzar con el examen se le presentará un reloj indicativo de tiempo insumido actual.

Al finalizar el cuestionario, el sistema debe establecer un puntaje a la sesión el cual estará dado por la siguiente formula:

$$\text{Puntaje} = \frac{\text{cantidad respuestas correctas}}{\text{cantidad de preguntas} * \text{factor dificultad} * \text{factor tiempo.}}$$

Donde

- Factor dificultad:
 - alto: 5 puntos.
 - medio: 3 puntos.
 - bajo: 1 punto.
- Factor tiempo: si tiempo total insumido / cantidad de preguntas
 - menor a 5 segundos : 5 puntos.
 - entre 5 y 20 segundos : 3 puntos.
 - mayor a 20 segundos : 1 punto.

El usuario podrá visualizar un ranking que muestre los 20 mejores sesiones en cuanto a su puntaje mostrando la siguiente información:

- usuario
- puntaje
- tiempo
- fecha

Requerimientos no funcionales

- La aplicación deberá ser robusta ante cualquier tipo de errores.
- La aplicación deberá ser fácil de usar e intuitiva.
- La interfaz del usuario deberá ser consistente y no tendrá que tener errores de interacción ni de visualización de información.
- La aplicación deberá ser desarrollado sobre la plataforma .NET y en lenguaje C#.
- El programa deberá tener una interfaz gráfica, se sugiere el uso de WinForms, integrando los conocimientos y técnicas adquiridos durante la cátedra. La incorporación de conocimientos no adquiridos durante la cátedra serán también bienvenidos.
- El programa deberá persistir las configuraciones y otros datos en una Base de Datos relacional, utilizando el gestor es a elección del alumno. Se espera que en un futuro puedan configurarse persistencia en distintos gestores de Bases de Datos u otras formas de persistencia (como por ejemplo archivos, Bases de Datos No-SQL, entre otras), por lo que el software debe estar preparado para ello.
- La aplicación deberá contener una bitácora de monitoreo (archivo de log), que permita hacer diagnósticos ante la ocurrencia de errores.
- **Se espera que en un futuro existan mas conjuntos de preguntas disponibles en la aplicación. El sistema deberá estar diseñado para que los cambios a realizar para incorporar esas nuevos conjuntos, sea el menor posible.**
- El código fuente deberá estar correctamente comentado y documentado con los

formatos correspondientes.

Entregables

Los entregables del trabajo final son:

- Código ejecutable.
- Código fuente.
- Diagrama de clases y paquetes en formato digital de alguna herramienta UML conocida.
- Guía de instalación y uso del programa.

Importante: Se debe entregar todos los mencionados elementos en un CD cuando se defienda el trabajo.

Evaluación

Se evaluará el uso de buenas técnicas de desarrollo y documentación, legibilidad del código, utilización correcta del lenguaje y finalmente la resolución empleada para satisfacer los requerimientos.

Se deberá realizar una defensa grupal e individual del trabajo en el momento de la entrega, en donde los alumnos deberán explicar la solución empleada y responder a preguntas de forma individual del equipo docente.