

## PROYECTO FINAL DE SEGUNDO AÑO

### CARRERA DE ANALISTA DE SISTEMAS

### ANALISTA PROGRAMADOR WEB .NET

2014

#### Generalidades

- La entrega final se hará durante la última clase de tutoría. Durante las clases de tutorías deberán realizarse entregas parciales de acuerdo a lo establecido en la sección **Entrega**.
- Se deberá entregar un sobre A4 cerrado con:
  - una copia impresa de toda la documentación solicitada
  - un CD o DVD con una copia digital de toda la documentación pedida y el proyecto completo (incluyendo el archivo .sln de la solución)
  - completar el formulario que se encuentra en la última página de la letra y pegarlo en el sobre de entrega. **No se aceptaran sobres escritos manualmente.**
- Se deberá **enviar todo el contenido digital** a la dirección [segundoanalista@bios.edu.uy](mailto:segundoanalista@bios.edu.uy)
- Se deberán formar grupos de 1 a 3 personas, **los cuales deberán inscribirse en bedelía desde el día 17 de noviembre hasta el día 27 de Noviembre**. Luego de esto, Bedelía informará la asignación de tutores así como de horarios.
- La asistencia a la última tutoría es obligatoria para **todos** los miembros del grupo ya que se realizara la defensa en maquina del proyecto.

#### Idea General

Se desea desarrollar un sistema que permita armar el juego del *Trivia*, para concursos en red.

#### Descripción de la Realidad

Los usuarios de la red se podrán registrar por si mismos a través del sitio web de la misma. Una vez obtenido el registro, podrán descargarse la aplicación del juego (aplicación escritorio).

El juego está compuesto por preguntas, las cuales abarcan tres áreas: **Deportes** (color rojo), **Ciencias** (color verde) y **Geografía** (color azul). Cada pregunta tiene asociadas 3 respuestas diferentes, de las cuales **solo una** será la respuesta correcta.

Cada juego de un usuario estará compuesto por 9 preguntas. Para el despliegue del juego en pantalla, se usara una cuadrícula de 3 x 3, compuesta por controles tipo **Button**. Cada uno deberá tener el color del área a la que pertenece la pregunta que representa. Por defecto cada botón desplegara el número de la pregunta (del 1 al 9); en caso de que la pregunta ya haya sido contestada correctamente, el número se cambiara por una **X**. Para moverse entre las preguntas, el usuario jugador deberá realizar una tirada de dados “virtual”, cuyo valor será el desplazamiento dentro de dicha cuadrícula. Para determinar que un juego está finalizado, el usuario jugador deberá haber contestado correctamente **todas** las preguntas

Los juegos serán creados por cada usuario jugador, asignándose las 9 preguntas de forma aleatoria (acción que hará automáticamente el sistema).

Todos los usuarios del sistema deberán tener nombre de usuario de logueo (**único** en el Sistema), contraseña, número de documento (**único** en el sistema e identificador del usuario) y nombre completo. De los usuarios de tipo jugador, también se deberá saber nombre público, el cual será desplegado en las estadísticas del sistema. De los usuarios administradores se deberá saber si pueden generar estadísticas, o no.

## Arquitectura Solicitada

Los jugadores podrán registrarse y acceder a la descarga del juego a través de un sitio web que será publicado en un servidor contratado para dicha funcionalidad. También a través de dicho sitio, tendrán acceso los administradores para realizar las tareas de: mantenimiento, autorización de juegos, y en caso de estar habilitados, ver las estadísticas del juego.

Los usuarios jugadores podrán jugar en aplicaciones de escritorio que se utilizarán en diferentes terminales.

La lógica de negocio del sistema estará ubicada dentro de un servidor de aplicaciones, y se podrá acceder a ella mediante un Servicio Web publicado en el mismo servidor.

La persistencia del sistema se instalará en un segundo servidor de aplicaciones, y la lógica de negocio deberá acceder a ella mediante **Remoting**.

La base de datos estará instalada en un servidor de datos. La persistencia se comunicará con dicho servidor mediante ADO.NET.

## Funcionalidades Mínimas del Sitio Web

**Formulario Web:** Principal (formulario por defecto del sitio)

**Actor Participante:** público

**Resumen:** En este formulario se desplegará el logo del juego, un hipervínculo a la página de registro de usuarios jugadores, y un control Login de ASP.NET para que un usuario registrado pueda loguearse (ingresará su nombre de usuario y contraseña). El logueo deberá ser realizado contra el repositorio de datos del sistema. Luego de realizada la autenticación, se determinará que tipo de usuario ingresó, refiriéndolo a la página principal correspondiente.

**Formulario Web:** Registro Jugador

**Actor Participante:** público

**Resumen:** En este formulario se solicitarán los datos necesarios para poder registrar a un jugador. Estos datos son: usuario logueo, contraseña, nombre completo, documento y nombre público que se mostrará en las estadísticas. Tomar en cuenta las restricciones mencionadas en la descripción de la realidad.

**Formulario Web:** Descarga Juego

**Actor Participante:** jugador

**Resumen:** En este formulario habrá un hipervínculo, el cual permitirá descargar un archivo ejecutable, el cual al ejecutarse en una pc, instalará la aplicación escritorio a través de la cual se podrá jugar al *Trivia*.

**Formulario Web:** Master Page

**Actor Participante:** administrador

**Resumen:** Esta MP, además de verificar si el usuario está logueado para poder ingresar a ella, desplegará:

1. Nombre completo del usuario logueado
2. Acción de deslogueo del sitio
3. Menú único (**obligatorio** uso de un control de navegación). Hay que tomar en cuenta si el usuario administrador tiene o no permisos para consultar estadísticas.

**Formulario Web:** ABM usuarios Administradores

**Actor Participante:** administrador

**Resumen:** En este formulario se solicitarán los datos necesarios para poder registrar a un nuevo administrador de juegos. Estos datos son: usuario logueo, contraseña, nombre completo, documento y si podrá o no ver estadísticas. Tomar en cuenta las restricciones mencionadas en la descripción de la realidad.

**Formulario Web:** ABM Preguntas

**Actor Participante:** administrador

**Resumen:** En este formulario se podrán agregar nuevas preguntas al sistema, para que estas puedan luego ser utilizadas en los juegos. Las preguntas tendrán un número (autogenerado por el sistema), serán de un tipo específico (geografía, ciencias o deportes) y tendrán un texto no superior a 200 caracteres que será el texto de la misma. Para cada pregunta se deberán ingresar tres respuestas, las cuales además del texto de las mismas (70 caracteres máximo), deberán saber si son la respuesta correcta o no, para la pregunta. Se deberán usar controles **RadioButton**, para determinar cuál respuesta es la correcta. Cuando se genera una pregunta, esta deberá crearse conjuntamente con sus tres respuestas, sí o sí. Las respuestas serán automáticamente numeradas del 1 al 3 para cada pregunta. Reglas de negocio y transacción lógica obligatorias.

**Formulario Web:** Estadísticas Juegos

**Actor Participante:** Administrador

**Resumen:** Este formulario permite obtener datos estadísticos de juegos *finalizados*. Se deberá partir de un documento con formato XML (alojado en la memoria, *no físicamente*), el cual contendrá un nodo por cada juego finalizado que forma parte de la red. Dicho documento se deberá generar en la operación correspondiente en el servicio Web del sistema. Mediante *Linq to XML*, se deberá poder filtrar dicho documento: todos los juegos, juegos ordenados por cantidad de movimientos, juegos ordenados por duración del juego. El despliegue de datos se realizará en una grilla.

**Formulario Web:** Estadísticas Juegos por Usuario Jugador

**Actor Participante:** Administrador

**Resumen:** Este formulario permite obtener datos estadísticos por jugador, solo de juegos *finalizados*. Se deberá partir de una colección de objetos, donde cada uno determinara a un usuario jugador con la cantidad de juegos finalizados que posee. Esta colección deberá ser desplegada dentro de un control *Repeater*. Si se selecciona uno de los registros de dicho control, en una grilla deberán desplegarse todos los juegos de dicho jugador (datos básicos únicamente).

**Formulario Web:** Usuarios de Base de Datos

**Actor Participante:** administrador

**Resumen:** Este formulario permite crear un usuario de logueo y de base de datos, en el Sql Server donde está alojada la base de datos del sistema. Para ello se deberá primero determinar nombre y contraseña del usuario a crear. Deberán desplegarse dos *DropDownList*, conteniendo los permisos que se pueden asignar a los usuarios a crear (uno para *roles* de servidor y otro con *permisos* sobre la base de datos). La creación de usuarios y la asignación de permisos deberán realizarse por medio de un procedimiento almacenado, que recibe el nombre del usuario a crear, su contraseña, rol de servidor y un único permiso sobre base de datos. Se deberá verificar previamente que dicho usuario no exista.

## Funcionalidades Mínimas de la Aplicación de Escritorio

**Formulario Windows:** Logueo

**Actor Participante:** jugador

**Resumen:** Este formulario permite que un usuario jugador se loguee y acceda al formulario principal de esta aplicación. Para ello se deberá solicitar usuario y contraseña mediante un **User Control**, que permita capturar su evento de registro, desde el lugar que se lo utilice. Dicho control deberá ser generado por el grupo de estudiantes.

**Formulario Windows:** Crear Juego

**Actor Participante:** jugador

**Resumen:** Este formulario solicita los datos necesarios para la generación de un nuevo juego. El usuario jugador podrá crear un nuevo juego, **únicamente** si no tiene juegos pendientes de finalización (fecha/hora finalización nula). Un juego está identificado por un número generado por el sistema, tiene una fecha y hora de inicio (generada automáticamente cuando se crea el juego) y pertenece a un solo jugador (en nuestro caso el usuario actualmente logueado). El sistema deberá aleatoriamente determinar las 9 preguntas que formaran parte del juego, sin desplegarlas en pantalla. Tomar en cuenta que cuando se genera dicho juego, la fecha y hora de finalización del juego debe ser nula y la cantidad de movimientos de este, deberá ser 0.

**Formulario Windows:** Jugar al Trivia

**Actor Participante:** jugador

**Resumen:** Este formulario automáticamente desplegará el juego sin finalizar del usuario jugador actual. Si no existe ninguno, dará mensaje de error. El despliegue se realizará de acuerdo a lo estipulado en la descripción de la realidad. El desarrollador deberá implementar la tirada de dados “virtual” dentro de un **Composite Control** (tomar en cuenta que los valores permitidos dentro de un dado son del 1 al 6). No importa si se contesta correctamente o no una pregunta, se deberá realizar una nueva tirada para moverse dentro del juego. No deberán tomarse en cuenta las preguntas contestadas correctamente para el movimiento dentro de la cuadrícula del juego. El usuario podrá guardar el juego en cualquier momento. Recordar que esta actualización del juego, deberá contemplar: cantidad de jugadas realizadas (acumuladas) y contestación correctas de preguntas (acumuladas). En caso que se contesten correctamente todas las preguntas, automáticamente el formulario deberá desplegar un mensaje de “**Éxito**”, y guardar el juego como finalizado, cerrando el formulario actual.

**Formulario Windows:** Histórico de Juegos

**Actor Participante:** jugador

**Resumen:** Este formulario permite desplegar datos estadísticos del jugador actual. Se deberá partir de un documento con formato XML (alojado en la memoria, **no físicamente**), el cual contendrá un nodo por cada juego del usuario jugador actualmente logueado. Dicho documento se deberá generar en la operación correspondiente en el servicio Web del sistema. Mediante **XPath**, se deberá poder filtrar dicho documento: todos los juegos o juegos cuya cantidad de movimientos sea superior a una ingresada por el usuario. El despliegue de datos se realizará en una grilla.





## Requerimientos de Implementación

- Implementación completa del sistema con tecnologías .NET en lenguaje C#. Obligatorio la entrega en **Visual Studio 2010**.
- La información deberá almacenarse obligatoriamente en una base de datos **SQL Server 2008**
- El script de la base de datos debe generarse *manualmente*, sin la ayuda de un asistente. Deberá contener el *Esquema de creación de la base de datos*, *Stored Procedures* necesarios para realizar todas las tareas solicitadas, *Inserción* de datos de prueba y *Creación de usuarios y permisos* necesarios para la correcta instalación y funcionamiento del sistema.
- Las eliminaciones a nivel de la base de datos deberán ser físicas o lógicas en función de las posibilidades. Es decir, si un elemento a eliminar no tiene elementos dependientes dentro de los registros de la base de datos, se elimina físicamente; de lo contrario se hará una eliminación lógica.
- Para el desarrollo del sistema utilizar la arquitectura en 3 capas vista en el curso, mediante la utilización de bibliotecas de clases.
- Obligatorio el uso de clases definidas por el usuario para la comunicación de datos entre componentes (tanto para invocación como respuesta).
- Los componentes de Lógica y Persistencia deberán ser generados en base a los patrones de Fábrica y Singleton vistos en clase. No deberá implementarse ningún otro patrón.
- Siempre se deberá de trabajar en forma conectada (ADO.NET) para el manejo de la información de la base de datos.
- Los componentes deberán lanzar excepciones en caso de error. No se contempla otra forma de comunicación de errores entre componentes
- La comunicación bajo Remoting deberá ser generada con sistema Singleton.
- Deberán usarse *Master Pages* y *Themes/CSS* para generar una interfaz gráfica uniforme en la web.

## Requerimientos de Entrega

- Solución completa del Software
- Script de la base de datos
- Modelo Conceptual.
- MER
- Diagrama de Clases completo de la Arquitectura en capas (incluye a todos los componentes, clases, interfaces y relaciones entre ellos)
- Diagrama de la Arquitectura del sistema
- Diagrama de Comunicación entre Componentes.

*Nota: todos los diagramas deberán ser generados con una herramienta para lenguaje UML. Estos deberán ser entregados en forma digital: una copia del archivo original del diagrama y una copia en formato PDF o JPG*

## Tutorías y Entregas

A continuación se detallan las entregas sugeridas:

- Primera Sesión (*semana del 26/01 al 30/01*):
  - Modelo Conceptual
  - Diagrama de Arquitectura
  - Diagrama de Comunicación de Paquetes
  - MER
  - Script BD
  - Interfaz gráfica web
- Segunda Sesión (*semana del 09/02 al 13/02*):
  - Sitio web totalmente funcional (sin Remoting ni Servicio Web)
  - Interfaz gráfica aplicación escritorio
- Tercera Sesión (*semana del 23/02 al 27/03*):
  - Servicio Web terminado
- Cuarta Sesión (*semana del 16/03 al 20/03*):
  - Entrega de todo lo solicitado en el punto **Requerimientos de Entrega**.
  - **Defensa obligatoria del proyecto**. Para ello el grupo deberá concurrir antes de su horario de tutoría, para realizar la instalación del sistema. Dicha instalación deberá respetar la arquitectura propuesta.

**CARRERA DE ANALISTA DE SISTEMAS**  
**PROYECTO FINAL DE SEGUNDO AÑO**  
**MARZO 2015**

NOMBRE DEL PROFESOR:

DATOS DE LOS INTEGRANTES DEL GRUPO

CI	NOMBRE	HORARIO

-----  
Comprobante de entrega (se corta y sella al entregar)

CI:

NOMBRE DEL ALUMNO:

HORARIO:

.....  
CI:

NOMBRE DEL ALUMNO:

HORARIO:

.....  
CI:

NOMBRE DEL ALUMNO:

HORARIO: