

Programación Web 3

UNLaM - Tecnatura en Desarrollo Web

Trabajo Práctico

1. Objetivo

Este documento describe el alcance funcional y los requisitos técnicos del trabajo práctico que los alumnos de la materia Programación Web III deberán realizar a fin de estar en condición de rendir ambos parciales de la materia.

2. Equipo

El equipo para realizar el trabajo práctico deberá ser de 4 alumnos. No se aceptará grupos de mayor cantidad de alumnos.

3. Requisitos Técnicos

3.1 Proyecto .NET

1. El trabajo práctico deberá ser realizado utilizando ASP.NET. El tipo de proyecto a utilizar es una aplicación web.

3.2 Estilos

1. No se permitirán que se utilicen los estilos ya provistos por Microsoft en la aplicación de ejemplo que provee Visual Studio.
2. Todos los archivos .css deberán estar dentro de una carpeta.
3. No utilizar estilos inline (atributo style="") ni definir estilos dentro de una pagina (tags <style>).
4. Debe de utilizarse algún framework/biblioteca de hojas de estilo. Algunos ejemplos:
 - a. Twitter Bootstrap (<http://getbootstrap.com/> , temas <http://bootswatch.com/>)
 - b. Foundation (<http://foundation.zurb.com/docs/>)
 - c. KickStart (<http://www.99lime.com/elements/>)
 - d. Otro definido por los alumnos y validado con el cuerpo docente.

3.3 JavaScript

1. No utilizar JavaScript inline dentro de una página, se deberá referenciar a archivos js.
2. Todos los archivos .js deberán estar dentro de una carpeta.
 - a. Si se decide utilizar algún js que no es propio, el mismo deberá estar dentro de una subcarpeta.
3. Debe utilizarse jquery. Las librerías plugins son opcionales. Para el caso de los plugins, queda como responsabilidad de los alumnos asegurar el correcto funcionamiento del mismo para garantizar el funcionamiento del trabajo práctico.

Recomendaciones

Las funciones específicas de una página, deberían estar en un archivo .js con el mismo

nombre de la página

.
Aquellas funciones utilizadas en más de una página, deberían de estar dentro de otro archivo .js de uso común.

3.4 HTML

1. No utilizar tags table.
2. Todo el contenido debe ser SEO friendly.
3. Se requiere el uso de master page para estructura los formularios web de la aplicación. Dentro de la master page deberán referenciarse las hojas de estilo y archivos de javascript de uso común por toda la aplicación.
4. Debe utilizarse xHtml.

3.5 Validación

1. Utilizar validaciones tanto del lado del cliente (JavaScript) como del lado del servidor.
2. Se puede utilizar una lista que detalle todos los campos que no cumplieron con las validaciones.
3. Se requerirá que las validaciones en los formularios webs serán realizadas mediante controles de validación de ASP.NET

3.6 Arquitectura y Consideraciones de Desarrollo

1. El proyecto deberá estar organizado en capas. Cabe aclarar, que el no conocimiento de este concepto, no implica que no puedan iniciar el desarrollo del trabajo en cuestión.
2. La capa de acceso a datos deberá ser realizada con ADO.NET. Este componente de .NET será explicado en clases a fin de que los alumnos comprendan cómo utilizarlo.
3. Utilizar la menor cantidad posible de código en los archivos aspx.cs, ascx.cs, master.cs, etc. e intentar que en los mismos haya llamadas a métodos dentro de otro proyecto que contenga las reglas de negocio

4. Encriptar información sensible, como la contraseña de los usuarios.
5. Compatibilidad con exploradores.
 - a. Internet Explorer 9,10
 - b. Firefox (la última versión para Windows).
 - c. Google Chrome (la última versión para Windows).

4. Objetivo del Proyecto

El proyecto a realizar consiste en un aplicación para organización y registrar torneos con sus equipos y jugadores que lo componen. Estará agrupados en dos partes: administración y resumen.

Las principales funcionalidades son:

- Menú y pantalla principal.
- Pantalla de Contacto.
- Creación de Torneo.
- Creación de Equipo.
- Creación de Jugador.

5. Especificación Funcional

5.1 Menú y Pantalla Principal

Al ingresar a la aplicación, se deberá ver un menú para acceder a la administración o al resumen de equipos. Esta pantalla podrá contener dos botones con diseño dirigiendo a una será una pantalla individual por defecto.

Cada grupo de pantallas deberán tener una master page propia con el acceso a cada una de las pantallas que componen el menú de cada uno de los grupos.

1- Grupo Resumen de Equipos:

- Pantalla de Listado de Equipos (pantalla por defecto).
- Contacto.

2- Grupo Administración:

- Creación equipos
- Creación jugadores
- Creación torneos (pantalla por defecto).

Cada master page deberá tener una estructura de menú, contenido y footer.

El footer deberá ser un user control llamado ucFooter y reutilizado en cada una de las master pages. El contenido del mismo será:

- Universidad Nacional de La Matanza.
- Dpto. de Ingeniería e Investigaciones Tecnológicas.
- Tecnicatura en Desarrollo Web - Programación Web III
- Integrantes : (nombres completos de los integrantes separados por “-”).

5.2 Pantalla de listado de equipos

Esta pantalla deberá contener un user control llamado ucListadoEquipos donde se listarán todos los equipos registrados en el sistema indicando el nombre del torneo en el participa. Cada equipo sólo puede participar en un torneo a la vez.

Además, habrá un filtro para ver sólo aquellos equipos de torneos activos o de inactivos.

5.3 Contacto

La pantalla de contacto permitirá registrar un contacto con los siguientes campos:

- nombre completo (string, requerido)
- email (string, requerido, con formato válido de email)
- comentario (string, requerido).

5.4 Creación equipos

La pantalla de creación de equipos tendrá los siguientes campos:

- nombre (string, requerido).
- torneo (combo, opcional: la primer opción debe ser “Elija Torneo”)
- Monto abonado (int, validar formato).

5.5 Creación jugadores

La pantalla de creación de jugadores tendrá los siguientes campos:

- nombre (string, requerido).
- apellido (string, requerido).
- edad (int, validar formato sólo enteros).
- equipo (combo, requerido)

5.6 Creación torneos

La pantalla de creación de torneos tendrá los siguientes campos:

- nombre (string, requerido)
- activo (bool, por defecto en verdadero).

6. Condiciones de Aprobación

1. El TP deberá cumplir con todos los requisitos técnicos y funcionales definidos. No se aceptará que alguna funcionalidad o requisito técnicos no se encuentren plasmados en el trabajo práctico.
2. Al momento de la entrega del trabajo práctico, todos los alumnos de cada equipo deberán estar presentes para la defensa del trabajo práctico.
3. En la defensa del trabajo, se evaluará el grado de conocimiento y participación en el desarrollo del trabajo práctico de cada alumno..

7. Forma de Entrega

El TP deberá ser enviado a los siguientes destinatarios, copiando a todos los integrantes del equipo en el email enviado.

- a) matias.paz.w@gmail.com
- b) psanchez@ing.unlam.edu.ar
- c) Copien al resto de los integrantes (como CC)

El email deberá contener los siguientes puntos:

- a. El título del email será 1C2015-TP1-[Dia]-[Integrantes]
- b. Respuestas a las siguientes preguntas:
 - i. ¿Qué nota creen que deberían sacar en el tp? (1-10, donde para 7 debe estar toda la funcionalidad pedida) y ¿por qué?
 - ii. ¿Qué cosas creen que podrían mejorarse?
 - iii. ¿Qué les resultó más complicado?
- c. Archivo .zip con toda la solución para que pueda ser corrida en cualquier computadora.