

## Programación Web 3

### UNLaM - Tecnicatura 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 aprobar a fin de estar en condición de aprobar/promocionar la materia.

### 2. Equipo

El equipo para realizar el trabajo práctico deberá ser de **4 alumnos**

## 3. Requisitos Técnicos

### 3.1 Proyecto .NET

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

### 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. Bulma (<http://bulma.io/>)
  - e. 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. **Se deberá utilizar Bundle and minification**  
 (<https://geeks.ms/etomas/2012/07/30/bundles-en-asp-net-mvc4/> )

#### 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 para organizar el contenido de una página en columnas, los tags table solo están permitidos para representar una grilla/listado de información.
2. Todo el contenido debe ser SEO friendly.
3. Se requiere el uso de Layouts para estructura los formularios web de la aplicación. Dentro del BaseLayout 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 utilizando DataAnnotations.
2. Se puede utilizar una lista que detalle todos los campos que no cumplieron con las validaciones.

### 3.6 Arquitectura y Consideraciones de Desarrollo

1. La capa de acceso a datos deberá ser realizada con Entity Framework. Este componente de .NET será explicado en clases a fin de que los alumnos comprendan cómo utilizarlo.
2. La capa Web deberá ser realizada utilizando MVC.
3. Deberán crear un repositorio privado en Github que deberá ser compartido con los profesores y donde irán subiendo los avances
4. Utilizar la menor cantidad posible de código en los /Controllers/[Entidad]Controller, e intentar que en los mismos haya llamadas a métodos dentro de otro proyecto que contenga las reglas de negocio.
5. Compatibilidad con exploradores.
  - a. Google Chrome (la última versión para Windows).

## 4. Objetivo del Proyecto

El proyecto a realizar consiste en realizar un sitio de una agencia de turismo. La misma servirá para que un administrador pueda administrar diferentes paquetes turísticos y que los usuarios puedan acceder al sitio, mirar los paquetes disponibles y reservar aquellos que les interese.

## 5. Especificación Funcional

Resumen de funcionalidad requerida:

**Ambos** (no es necesario estar logueado)

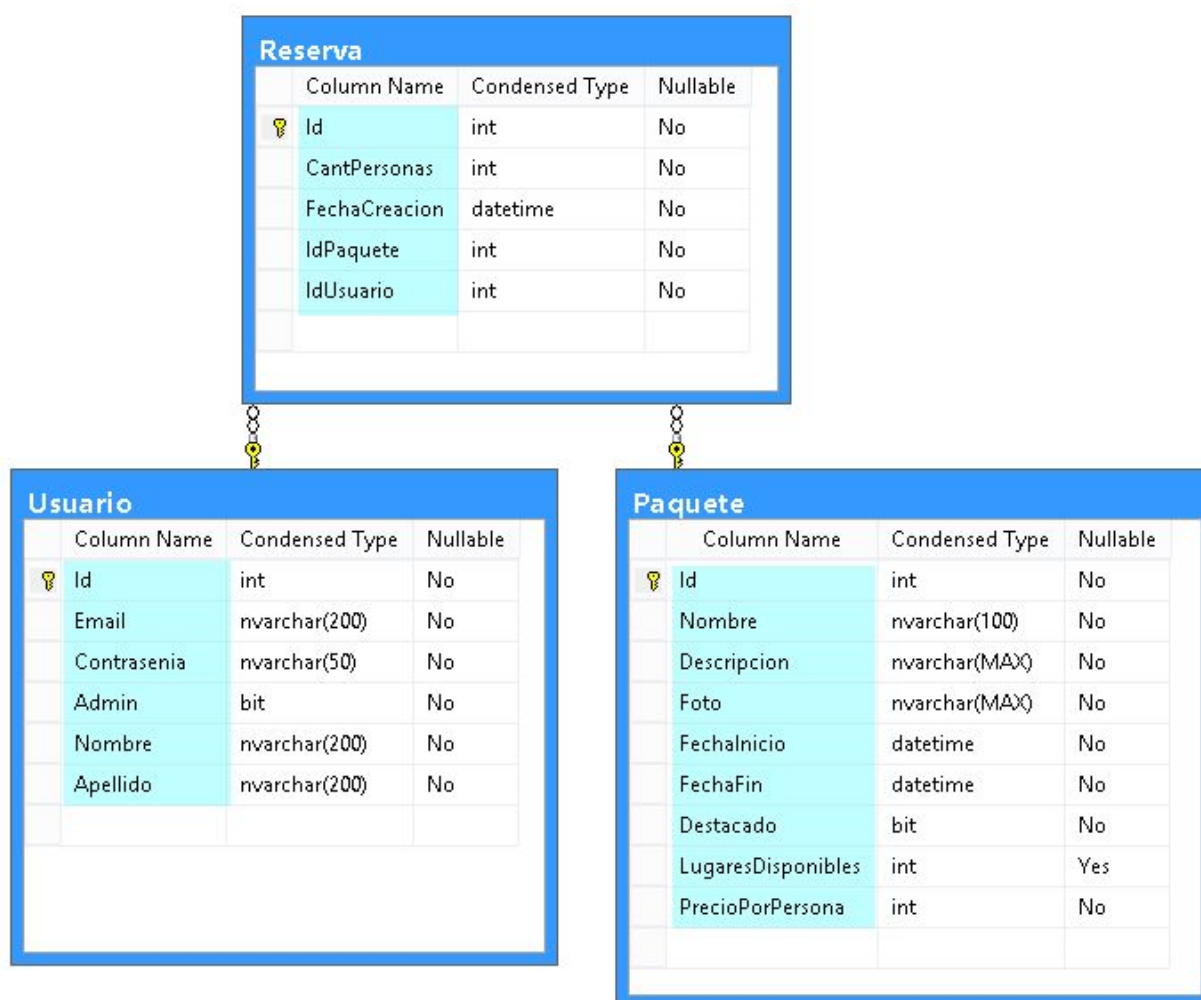
1. Página de inicio (**/home/index y /**)
2. Menú específico para cada grupo
3. Login/Logout de usuario (para login **/Home/login** y **/Home/logout** logout, puede ser solo un link el logout que te redirija la home.
4. Ver detalle de Paquete (**/paquetes/detalle**)

**Grupo Administración** (es necesario estar logueado)

5. Listado de Paquetes (**/admin/paquetes/index**)
6. Creación de Paquetes (**/admin/paquetes/nuevo**)
7. Eliminación de Paquetes (**/admin/paquetes/eliminar**)

**Grupo Usuario** (es necesario estar logueado)

8. Reservar Paquete (**/paquetes/reservar**)
9. Ver mi historial de Paquetes reservados (**/paquetes/historial**)



## 5.1 Pantalla Principal

Al ingresar al sitio (página /home/inicio), se deberán ver todos los paquetes **Destacados**, donde la fecha de inicio sea mayor a la fecha actual y ordenados por fecha de inicio ascendentemente.

Cada paquete visualizará la foto (dejar fijo el tamaño), el nombre, el precio y fecha de inicio. Por ejemplo:



Al hacer click en el nombre se debe redirigir a la página de ver detalle del paquete pasando por querystring el id del paquete (por ejemplo /detalle.aspx?id=123)

## 5.2 Menú, Partial Views y Layouts

Siempre habrá un menú visible.

### Menú Administrador

- Inicio (/home/inicio)
- Listado de Paquetes (/admin/paquetes/index)
- Creación de Paquetes (/admin/paquetes/nuevo)
- Logout (visible en caso de estar logueado, /home/logout)

**Menú Usuario**

- Inicio (/home/index)
- Historial de Paquetes reservados (visible en caso de estar logueado, /paquetes/historial)
- Logout (visible en caso de estar logueado, /home/logou)

**Menú para Anónimos** (sin estar logueado)

- Inicio (/home/inicio)
- Login (/home/login)

Para cada menú crear una **partial view** diferente (\_pvMenuAdmin, \_pvMenuUsuario, \_pvMenuAnonimo).

**Crear 4 Layouts, BaseLayout, AnonimoLayout, UsuarioLayout, AdminLayout.**

En **BaseLayout.cshtml** se debe poner todo el html compartido (links a css, js, etc, **usando Bundle**) y agregar las 3 partial views de menú, y en el code behind se hace la verificación de si que menú corresponde mostrar.

**AnonimoLayout.cshtml** debe implementar BaseLayout.cshtml y será implementada por la página de Inicio y Login.

**UsuarioLayout.cshtml** debe implementar BaseLayout.cshtml, será implementada por las páginas de Reservar Paquete, Historial de Paquetes reservados y en el code behind debe validar que el usuario esté logueado y no sea administrador, caso contrario redirige a la página de Inicio.

**AdminLayout.cshtml** debe implementar BaseLayout.cshtml y será implementada por la página de Listado de Paquetes, Creación de Paquetes y Eliminación de Paquetes y en el code behind debe validar que el usuario esté logueado y sea administrador, caso contrario redirige a la página de Inicio.

### 5.3 Login/Logout.

La pantalla contendrá un campo email y contraseña. Al ingresar en el menú del sitio estará visible un linkbutton de logout que al clickearlo borrará los datos de sesión y redirigirá a la página de Inicio.

**Validación:**

- **Email** - Requerido y Formato de email. (**html id y name=email**)

- **Contraseña** - Requerido. (**html id y name=contrasenia**)
- **Botón “Ingresar”**, (**html id y name=btnIngresar**) en caso de que se haya accedido a la página de login a través de otra página automáticamente, luego de hacer click en “Ingresar”
- **Usuario y/o Contraseña inválidos** (en caso de que no se encuentre la combinación de usuario y contraseña ingresada) (**html id y name=errorGeneral**)

## 5.4 Ver detalle de Paquete.

En esta pantalla se visualizará la información de un paquete. En el code behind modificar el Page.Title (esto modifica el tag title de html) para que sea el nombre del paquete.

La pantalla contendrá:

- **Nombre** (**html id y name=nombre**)
- **Foto** (**html id y name=foto**)
- **Descripción** (**html id y name=descripcion**)
- **Fecha de Inicio** (**html id y name=fechaInicio**)
- **Fecha de Fin** (**html id y name=fechaFin**)
- **Un Botón “Reservar”** (**html id y name=btnReservar**) Este botón estará visible solo en los siguientes casos: la Fecha de Inicio sea mayor a la fecha actual, y Lugares Disponibles sea mayor a cero o null (no tiene límite).

**Al hacer click en Reservar**, este botón debe redirigir a la página de reserva (/paquetes/reservar/[id]) enviando el id de paquete en la url.

## 5.5 Listado de Paquetes (Administrador) - /admin/paquetes/index

Esta página será visible solo para los usuarios administradores. Debe visualizarse un link/boton de Nuevo (que lleve a [5.6 Creación de Paquetes \(Administrador\)](#)) y listado de todos los paquetes, mostrando **solo** las columnas Nombre, Inicio, Fin, Precio, Lugares, Destacado, Editar, Eliminar. **Editar y Eliminar deberán ser links**.

La lista debe estar **ordenada descendientemente por Fecha de Inicio y debe poseer paginación**.

Si el usuario no está logueado, se debe redirigir a la página de Login y luego de loguearse se deberá redirigir automáticamente a esta página (tip, al redirigir /home/login, enviar por querystring la url destino, por ejemplo /home/login?redirigir=/admin/paquetes/index).

Al hacer click en Eliminar se debe redirigir a **/admin/paquetes/eliminar/[id]**



Aquellos paquetes cuya fecha de inicio es menor a la fecha actual, deben estar con otro color (de letra o de fondo)

## 5.6 Creación/Edición de Paquetes (Administrador) - /admin/paquetes/nuevo y /admin/paquetes/editar

La pantalla de creación de paquetes tendrá los siguientes campos:

- **Nombre** - TextBox - Requerido. (**html id y name=nombre**)
- **Descripción** - TextBox - Requerido - No usar TextBox multilinea porque es complejo para validar.(**html id y name=descripcion**)
- **Foto (1)** - FileUpload - Requerido (**html id y name=foto**) (ejemplo de referencia de FileUpload <https://github.com/unlam-pw3/subir-imagen>)
- **Fecha Inicio** - TextBox - Requerido - Solo fechas válidas (**html id y name=fechaInicio**)
- **Fecha Fin** - TextBox - Requerido - Solo fechas válidas - Fecha Inicio debe ser menor a Fecha Fin.(**html id y name=fechaFin**)
- **Lugares Disponibles** - TextBox - Solo enteros positivos (**html id y name=lugares**)
- **Precio por Persona** - TextBox - Requerido - Solo enteros positivos (**html id y name=precioPorPersona**)
- **Destacado** - Checkbox (**html id y name=destacado**)
- **Botón “Crear”**, debe validar y guardar toda la información en la base de datos y dirigirse automáticamente a la página de listado de paquetes. (**html id y name=btnCrear y btnEditar**)
- Luego de Crear, y ser redirigido a la página de listado de paquetes, mostrar el mensaje “Paquete <nombre> ha sido creado exitosamente”, para ello pueden usar una variable en sesión (que luego de usarla deben borrar) o querystring.
- Luego de Editar, y ser redirigido a la página de listado de paquetes, mostrar el mensaje “Paquete <nombre> ha sido modificado exitosamente”, para ello pueden usar una variable en sesión (que luego de usarla deben borrar) o querystring.
- **Botón “Cancelar”**, debe redirigir a la página anterior (**html id y name=btnCancelar**)

Para todos los campos validar que la cantidad de caracteres ingresados no exceda el límite impuesto en la base de datos.

## 5.7 Eliminación de Paquetes (Administrador) - /admin/paquetes/eliminar

- Mensaje diciendo “¿Está seguro que desea eliminar el paquete<nombre paquete>? Actualmente posee <cantidad de reservas>.”

- Botón de “Eliminar”, debe eliminar todas las reservas y eliminar el paquete. Luego se debe redirigir automáticamente a la página de listado de paquetes. (**html id y name=btnEliminar**)
- Luego de Eliminar, y ser redirigido a la página de listado de paquetes, mostrar el mensaje “Paquete <nombre> ha sido eliminado exitosamente”, para ello pueden usar una variable en sesión (que luego de usarla deben borrar) o querystring.
- Botón de “Cancelar”, debe volver a la página de listado de paquetes. (**html id y name=btnCancelar**)

Si el usuario no está logueado, se debe redirigir a la página de Login y luego de loguearse se deberá redirigir automáticamente a Listado de Paquetes /admin/paquetes/default

## 5.8 Reservar Paquete (Usuario) - /paquetes/reservar

- Mensaje diciendo “Usted está a punto de reservar el paquete <Nombre Paquete>.”
- **Cantidad de Pasajeros** - Viene prepopulado con el valor 1 - Requerido (**html id y name=pasajeros**)
- **Botón de “Confirmar”**, al hacer click debe guardarse en la base de datos la información y redirigir a la página de inicio. Validar que la Fecha de Inicio sea mayor a la fecha actual, y Lugares Disponibles sea mayor a cero o null (no tiene límite). (**html id y name=btnConfirmar**)
- Botón de “Cancelar”, debe volver a la página anterior. (**html id y name=btnCancelar**)

Si el usuario no está logueado, se debe redirigir a la página de Login y luego de loguearse se deberá redirigir automáticamente a Listado de Paquetes /admin/paquetes/index.

## 5.9 Ver mi historial de Paquetes reservados (Usuario) - /historial

Se debe visualizar todas los paquetes reservados por el usuario ordenados por fecha de inicio.

Las columnas a visualizar son:

- **Nombre:** nombre del paquete (**html id y name=nombre**)
- **Fecha Inicio:** fecha de inicio del paquete (**html id y name=fechaInicio**)
- **Fecha Fin:** fecha de fin del paquete (**html id y name=fechaFin**)
- **Cantidad:** la cantidad de personas que incluye mi reserva (**html id y name=cantidad**)
- **Precio:** precio (precio por persona) (**html id y name=precioPorPersona**)
- **Precio Total:** precio total (precio por persona X cantidad) (**html id y name=precioTotal**)
- **Botón “Cancelar Reserva”**, Al hacer click se debe ejecutar una llamada de **AJAX** utilizando jquery que llame a un servicio web (asmx) o Web Api de reservas /Api/Reservas/Cancelar donde Cancelar es el nombre del metodo y enviandole el id de la reserva y (ejemplo de referencia web api

<https://www.codeproject.com/Articles/424461/Implementing-Consuming-ASP-NET-WEB-API-from-JQuery> ). En el método se debe validar que el usuario está logueado, que la reserva es suya, y que la fecha de inicio sea mayor a la fecha actual, en el caso de que alguna de estas cosas no sea válida, mostrar un mensaje por pantalla (escriban un span o similar pero que posea al menos la clase "error").

Si el usuario no está logueado, se debe redirigir a la página de Login y luego de loguearse se deberá redirigir automáticamente a esta página (tip, al redirigir al login.aspx, enviar por querystring la url destino, por ejemplo /home/login?redirigir=/paquetes/historial).

## 6. Forma de Entrega

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

- a) mpazwasiuchnik@unlam.edu.ar
- b) [pablokuko@gmail.com](mailto:pablokuko@gmail.com) ; pnsanchez@unlam.edu.ar
- c) el resto de los integrantes que participaron del tp (como CC)

El email deberá contener los siguientes puntos:

- a. El título del email será 2017-2C-TP-[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 (renombrar extension a .txt)** con toda la solución para que pueda ser compilada y ejecutada en cualquier computadora.

## 7. 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.