

## **OBLIGATORIO**

### **CARRERA ANALISTA DE SISTEMAS**

### **DISEÑO APLICACIONES I**

### **MAYO 2023**

## **GENERALIDADES**

- La entrega final deberá realizarse el día **lunes 14 de Agosto**.
- Se deberá realizar la entrega mediante correo electrónico, del software junto con toda la documentación exigida en la sección **Requerimientos de Entrega**.
  - **Asunto:** Entrega Obligatorio DI 2023
  - **Destinatarios:** segundoanalista@bios.edu.uy y docente del curso (OBLIGATORIO)
  - **Adjuntos:** documentación
  - **Cuerpo:** Nombre y Cedula de los integrantes del grupo que realiza la entrega
  - **Compartir la solución**
- De ser necesario un comprobante de entrega, **solo está habilitada Bedelía** a realizarlo.
- El valor de este obligatorio es de **50 por ciento** del total del curso.
- El obligatorio deberá realizarse en **grupos de 1 a 3 personas**.

## **DESCRIPCIÓN DE LA REALIDAD**

Se requiere de un sistema que permita gestionar los viajes, de una terminal de ómnibus de Montevideo; además de las compañías que los realizan, los destinos a los que se viaja y sus empleados.

De las Terminales (destinos por los que pasa /llegan los viajes) se conoce un código (compuesto por 6 letras) y el nombre de la ciudad en donde están ubicados. Existen dos tipos de Terminales: internacionales, de las que se conoce en qué país están; y las nacionales, de las que se sabe si cuentan con servicio de taxi (o no). Por ejemplo: Terminal Nacional (PayNor – Paysandú – No tiene servicio de taxis); Terminal Internacional (PalBra – Porto Alegre – Brasil)

De las compañías de ómnibus que realizan los viajes se sabe su nombre (que debe ser único en el sistema), la dirección de su casa matriz y un teléfono de contacto. Por ejemplo: Compañía (BusRapido – 18 de Julio 1456 – 099787878).

De los Viajes se sabe:

1. Código Interno (único en el sistema – autogenerado )
2. Fecha y Hora de partida
3. Fecha y Hora aproximada de llegada a destino final
4. Cantidad máxima de pasajeros (máximo permitido 50)
5. Precio del boleto
6. Anden del cual salen (la terminal cuenta con andenes numerados del 1 al 35)
7. Empleado que genero el viaje
8. Compañía que realiza el viaje
9. Conjunto de terminales por las que pasa el recorrido del viaje. Deben definirse en orden.



## **FUNCIONALIDADES DEL SITIO WEB**

**Página Web:** Pagina Inicial del sitio (Default)

**Actor Participante:** publico

**Resumen:** Esta página permite que un usuario se loguee y acceda a la página principal del sitio de la empresa en modo administración. Para ello se deberá solicitar nombre de usuario y contraseña. La autenticación del empleado se deberá realizar manualmente y mediante el repositorio de datos del propio sistema.

Además, se mostrarán todos los viajes que aún no partieron, es decir todos los viajes cuya fecha/hora de salida sean a futuro. Deberá desplegarse la siguiente información:

Numero de Viaje – FechaHora Salida – Anden de Salida – Destino Final

Este listado debe estar ordenado por fecha y hora (siendo el primero en listarse el viaje más próximo a partir)

**Aclaración 1:** *Destino final es la última terminal del recorrido del viaje*

**Aclaración 2:** *El listado debe desplegarse en un control GridView*

**Página Web:** Alta Empleado (usuario registrado del sistema)

**Actor Participante:** Usuario

**Resumen:** Esta página permite crear un usuario para el empleado en nuestro sistema. El nombre de usuario para loguearse debe ser único en el sistema (identificador) y contar con 8 caracteres exactamente. La contraseña debe ser de 6 de largo (3 letras y 3 números). También interesa saber su nombre completo. ***No se contempla la modificación ni eliminación en esta instancia***

**Página Web:** Alta Compañía

**Actor Participante:** Usuario

**Resumen:** Esta página permite crear una nueva compañía en nuestro sistema. Se deberán determinar los datos mencionados en el apartado “***Descripción de la Realidad***”. ***No se contempla la modificación ni eliminación en esta instancia***

**Página Web:** Alta Viajes

**Actor Participante:** Usuario

**Resumen:** Esta página permite crear un nuevo viaje. Se deberán determinar los datos mencionados en el apartado “***Descripción de la Realidad***”, considerando que se deberá tomar al usuario actualmente logueado como el empleado que genere el viaje. Considerar que un viaje puede recorrer varias terminales, y por lo tanto deberán guardarse en el orden de la recorrida que marco el usuario.

**Ejemplo 1:** el viaje parte de Montevideo, pasa por las terminales de Flores y Fray Bentos, para terminar el recorrido en la terminal de Paysandú. **Ejemplo 2:** el viaje parte de Montevideo, pasa por las terminales del Chuy y Porto Alegre, para terminar el recorrido en la terminal de Florianópolis. Como un viaje puede tener varias terminales en su recorrido, esta lista debe manejarse conjuntamente con el viaje; obligatorio manejo de transacción lógica. ***No pueden ni eliminarse ni modificarse los viajes bajo ningún concepto.***



**Página Web:** ABM Terminal Nacional

**Actor Participante:** Usuario

**Resumen:** Esta página permite mantener información de las terminales Nacionales que pueden ser destinos de los viajes. Se deberán determinar los datos mencionados en el apartado “*Descripción de la Realidad*”.

**Página Web:** ABM Terminal Internacional

**Actor Participante:** Usuario

**Resumen:** Esta página permite mantener información de las terminales Internacionales que pueden ser destinos de los viajes. Se deberán determinar los datos mencionados en el apartado “*Descripción de la Realidad*”.

**Página Web:** Listados de Viajes

**Actor Participante:** Usuario

**Resumen:** En esta página por defecto se muestran los viajes cuya fecha de partida este en el mes actual (mes y año de hoy). Dicha lista, se podrá filtrar por:

- compañía (**desplegadas en un control DropDownList en donde se muestre el nombre de la compañía**)
- destino final (**desplegados en un control DropDownList en donde se muestre el nombre de la ciudad de la terminal de destino final**)
- por fecha (**el usuario podrá ingresar manualmente este dato**)
- o una combinación de las anteriormente mencionadas

Además, Se deberá contar con un mecanismo para limpiar los filtros realizados, volviendo al listado original que se mostró por defecto.

*Estos tipos de filtros (todos) deben realizarse mediante Linq to Objects (obligatorio).*

\*\* Tomar en cuenta que deberá tenerse una Master Page (para páginas del sitio *no públicas*), con:

1. Nombre del usuario actualmente logueado
2. Acceso para desloguearse del sitio
3. Menú principal con todas las acciones que pueda realizar el usuario.



## REQUERIMIENTOS DE IMPLEMENTACIÓN

- Implementación completa del sistema con tecnologías .NET en lenguaje C#. Obligatorio la entrega en **.Net Framework 4.5.2 o 4.6.**
- La información deberá almacenarse obligatoriamente en una base de datos **SQL Server.**
- 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, e **Inserción** de datos de prueba. Las restricciones sobre datos deberán realizarse en la propia estructura de la tabla (uso obligatorio de los modificadores **unique**, **check**, o **default**). Tomar en cuenta que usuario se necesita para la ejecución del **Sitio publicado.**
- 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 relaciones dentro de los registros de la base de datos, se elimina físicamente; de lo contrario se hará una eliminación lógica. **Obligatorio de implementar**
- 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). Dichas clases deben contar con la lógica de dato necesaria y solo con un constructor completo.
- 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.**
- **Deberá publicarse el sitio, para su prueba final.**

## Requerimientos de Entrega

- Solución completa del Software
- Script de la base de datos
- Modelo Conceptual
- DER – RNE – Pasaje a Tablas
- Diagrama de Clases completo de la Arquitectura en capas (incluye a todos los componentes, clases, interfaces y relaciones entre ellos)

*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*