

Universidad Tecnológica Nacional
Facultad Regional Avellaneda



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

Materia: Laboratorio III

Apellido:		Fecha:	13/05/2022
Nombre:		Docente ⁽²⁾ :	Horacio Serrano
División:		Nota ⁽²⁾ :	
Legajo:		Firma ⁽²⁾ :	
Instancia ⁽¹⁾ :	PP	RPP	SP x RSP FIN

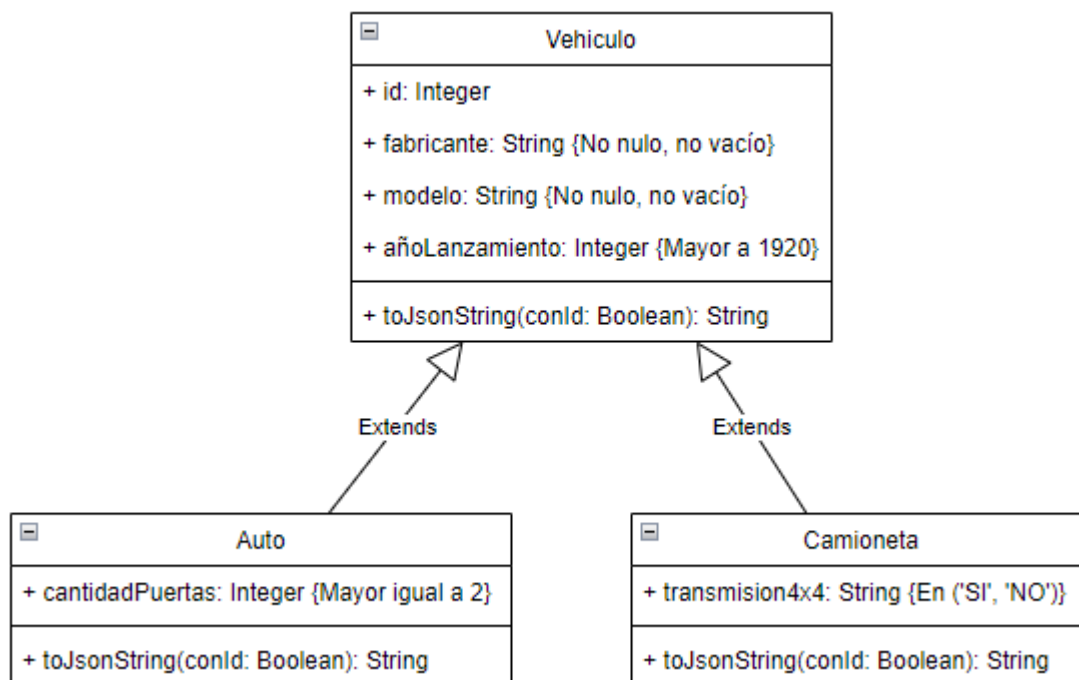
(1) Las instancias válidas son: 1^{er} Parcial (**PP**), Recuperatorio 1^{er} Parcial (**RPP**), 2^{do} Parcial (**SP**), Recuperatorio 2^{do} Parcial (**RSP**), Final (**FIN**). Marque con una cruz.

(2) Campos a ser completados por el docente.

Crear ABM de Personas.

1- Implementar la siguiente jerarquía de Clases. Las restricciones deben aplicarse en los formularios de ABM mediante Validaciones, no en las clases.

La operación `toJsonString`, recibe por parámetro un Booleano y debe generar una string la cual pueda ser parseada para reconstruir un objeto desde un formato JSON, con los mismos atributos y valores que el objeto original. En caso de que el parámetro `conId` sea Verdadero, la string Json debe incluir el Id, en cualquier otro caso NO debe incluir.



2- Implementar los Sigüientes Elementos de Interfaz.

a- **Formulario Lista:** Un formulario que contenga una lista de todos los elementos recuperados en el punto 3. Cada Fila debe Tener una columna para cada Atributo de todas las clases implementadas en el punto 1, en caso de que el atributo no aplique al elemento presentado en esa fila rellenar con N/A.

Cada fila debe tener además 2 columnas adicionales con un botón cada una, el primero con el texto "Modificar" el segundo con el texto "Eliminar"

Luego de la Tabla con los Elementos debe haber 1 botón con texto "Agregar Elemento".

b- **Formulario ABM:** Un formulario que tenga campos para todos los atributos de las clases implementadas en el punto 1, mostrando los que correspondan según la acción a realizar. Un Encabezado de texto que diga "Alta", "Baja", "Modificación", según corresponda y 2 botones con textos "Aceptar" y "Cancelar".

c- **Spinner:** Un contenedor transparente de Pantalla completa con un "Spinner".

Utilizar Estilos, se incluyen imágenes orientativas.

Formulario Lista

ID	Encabezado1	Encabezado1	Encabezado1	Encabezado1	Modificar	Eliminar
id1	atributo1	atributo2	atributo3	N/A	Modificar	Eliminar
id2	atributo1	atributo2	atributo3	N/A	Modificar	Eliminar
id3	atributo1	atributo2	N/A	atributo4	Modificar	Eliminar
id4	atributo1	atributo2	atributo3	N/A	Modificar	Eliminar

Formulario ABM

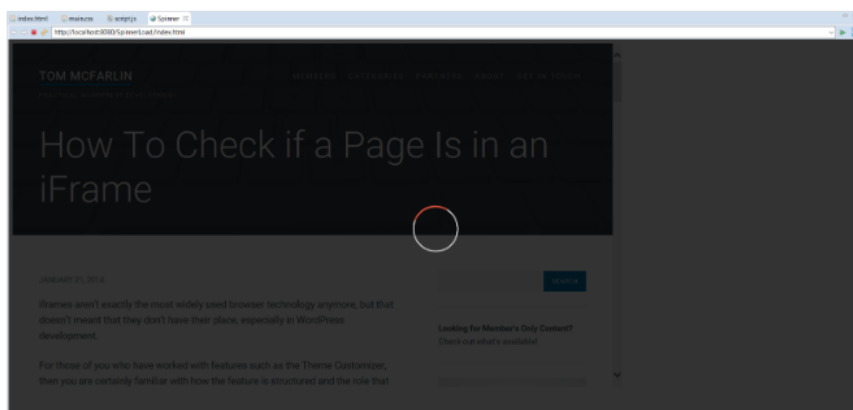
Id:

Atributo1:

Atributo2:

Tipo: ▼

Atributo3:



3- Generar una lista en memoria de la jerarquía de clases implementada en el punto 1 a partir de una cadena de caracteres de objetos Json. Esta lista se generará por única vez. Las acciones subsecuentes deberán actualizar la lista acorde al resultado de la acción realizada.

Obtener la cadena desde la api provista con el endpoint Vehiculos/Vehiculos, con el Verbo GET y sin Parámetros. En caso de tener una respuesta con código 200 Generar la lista en memoria y mostrar el Formulario Lista, de lo contrario Lanzar una advertencia por pantalla.

NO Utilizar la clase XMLHttpRequest, **Utilizar** Fetch, **NO utilizar** función Asíncrona, **Garantizar Secuencialidad**.

4- Implementar Funcionalidad de "Alta". Aplicar Validaciones Acorde a las Restricciones descritas en el diagrama UML.

Al hacer Click en el botón "Agregar Elemento", debe ocultar el "Formulario Lista" y mostrar el "Formulario ABM"(centrado en la pantalla). Este debe indicar en el encabezado la acción a realizar, bloquear el ingreso de Id (que serán generados por el API) y mostrar únicamente los campos correspondientes al tipo de elemento que se está insertando.

Al hacer Click en "Aceptar", bloquear la pantalla con el contenedor "Spinner", y realizar la solicitud al API sobre el endpoint /Vehiculos/InsertarAuto si es un Auto o /Vehiculos/InsertarCamioneta si es una Camioneta, con el Verbo PUT, con encabezado Content-Type de valor application/json y cuerpo un string de objeto JSON que representa los atributos del elemento a insertar SIN el ID.

En caso de recibir un resultado con código 200, actualizar el ID con el provisto en la respuesta, insertar el nuevo elemento en la lista, ocultar el contenedor "Spinner", ocultar "Formulario ABM" y finalmente mostrar "Formulario Lista" el cual debe reflejar los cambios.

En caso de NO recibir un código 200 ocultar el contenedor "Spinner", ocultar "Formulario ABM", mostrar "Formulario Lista" y realizar una advertencia de que no se pudo realizar la operación.

En caso de Hacer click en Cancelar, ocultar "Formulario ABM" y mostrar "Formulario Lista".

Utilizar la clase XMLHttpRequest, **NO utilizar** Fetch, **NO utilizar** Promesas, **Garantizar Secuencialidad**.

5- Implementar Funcionalidad de "Modificación". Aplicar Validaciones Acorde a las Restricciones descritas en el diagrama UML.

Al hacer Click en el botón "Modificar" de una fila, debe ocultar el "Formulario Lista" y mostrar el "Formulario ABM"(centrado en la pantalla). Este debe indicar en el encabezado la acción a realizar, bloquear el ingreso de Id y mostrar únicamente los campos correspondientes al tipo de elemento que se está modificando con los valores correspondientes.

Al hacer Click en "Aceptar", bloquear la pantalla con el contenedor "Spinner", y realizar la solicitud al API sobre el endpoint /Vehiculos/ModificarAuto si es un Auto o /Vehiculos/ModificarCamioneta si es una Camioneta, con el Verbo POST, con encabezado Content-Type de valor application/json y cuerpo un string de objeto JSON que representa los atributos del elemento a modificar.

En caso de recibir un resultado con código 200, actualizar el ID con el provisto en la respuesta, actualizar el elemento de la lista, ocultar el contenedor "Spinner", ocultar "Formulario ABM" y finalmente mostrar "Formulario Lista" el cual debe reflejar los cambios.

En caso de NO recibir un código 200 ocultar el contenedor "Spinner", ocultar "Formulario ABM", mostrar "Formulario Lista" y realizar una advertencia de que no se pudo realizar la operación.

En caso de Hacer click en Cancelar, ocultar "Formulario ABM" y mostrar "Formulario Lista".

Garantizar Secuencialidad.

6- Implementar Funcionalidad de "Eliminación". Aplicar Validaciones Acorde a las Restricciones descritas en el diagrama UML.

Al hacer Click en el botón "Eliminar" de una fila, debe Ocultarse el "Formulario Lista" y mostrar el "Formulario ABM"(centrado en la pantalla). Este debe indicar en el encabezado la acción a realizar, bloquear el ingreso de Id y mostrar únicamente los campos correspondientes al tipo de elemento que se está eliminando con los valores correspondientes.

Al hacer Click en "Aceptar", bloquear la pantalla con el contenedor "Spinner", y realizar la solicitud al API sobre el endpoint /Vehiculos/EliminarVehiculo, con el Verbo DELETE, con encabezado Content-Type de valor application/json y cuerpo un string de objeto JSON con el id del elemento a eliminar.

En caso de recibir un resultado con código 200, actualizar la lista, ocultar el contenedor "Spinner", ocultar "Formulario ABM" y finalmente mostrar "Formulario Lista" el cual debe reflejar los cambios.

En caso de NO recibir un código 200 ocultar el contenedor "Spinner", ocultar "Formulario ABM", mostrar "Formulario Lista" y realizar una advertencia de que no se pudo realizar la operación.

En caso de Hacer click en Cancelar, ocultar "Formulario ABM" y mostrar "Formulario Lista".

Utilizar Función Asíncrona, **NO utilizar** la clase XMLHttpRequest, **Utilizar** Fetch **Garantizar Secuencialidad**.