

**Universidad Tecnológica Nacional**  
**Facultad Regional Avellaneda**



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

**Materia: Laboratorio III**

Apellido:		Fecha:	28/6/2022
Nombre:		Docente <sup>(2)</sup> :	Christian Baus
División:	3°D	Nota <sup>(2)</sup> :	
Legajo:		Firma <sup>(2)</sup> :	
Instancia <sup>(1)</sup> :	<div style="display: flex; justify-content: space-around;"> <span><b>PP</b></span> <span></span> <span><b>RPP</b></span> <span></span> <span><b>SP</b></span> <span><b>X</b></span> <span><b>RSP</b></span> <span></span> <span><b>FIN</b></span> <span></span> </div>		


(1) Las instancias válidas son: 1<sup>er</sup> Parcial (**PP**), Recuperatorio 1<sup>er</sup> Parcial (**RPP**), 2<sup>do</sup> Parcial (**SP**), Recuperatorio 2<sup>do</sup> Parcial (**RSP**), Final (**FIN**) . Marque con una cruz.

(2) Campos a ser completados por el docente.

**TEMA 1:**


**Realizar una ABM de Anuncios de Automóviles.**

- Desarrollar una aplicación que cuente con una única página donde exista el listado de los anuncios dados de alta(mostrar un spinner mientras cargan), así como un formulario para dar de alta a nuevos anuncios. Para esto deberá usar HTML y CSS para darle agradable “look and feel”.
- La lista de anuncios deberá ser generada dinámicamente(con código JS) en base a la estructura del objeto utilizado, además deberá contar con algún manejador de eventos, tal que al hacer click en algún elemento de la lista, se carguen los valores del anuncio seleccionado en el formulario. Se deberán habilitar los botones necesarios para permitirnos cancelar la edición o eliminar el anuncio.



[Nosotros](#)
[Anuncios](#)
[Blog](#)
[Contacto](#)

## Administración



### Información Del Anuncio

**TITULO:**

**TRANSACCION:**  

VENTA

ALQUILER

**DESCRIPCION:**

**PRECIO:**

Puertas

KMs

Potencia

Guardar

id	titulo	transaccion	descripcion	precio	num_puertas	num_KMs	potencia
1	Jeep Renegade	venta	Jeep Renegade 2017	1,000,000	5	10,000	120
2	Renault Megane	venta	Renault Megane 2014	800,000	5	80,000	140

[Nosotros](#)
[Anuncios](#)
[Blog](#)
[Contacto](#)

Todos los Derechos Reservados 2019 ©

## Especificaciones

### header



[Nosotros](#)
[Anuncios](#)
[Blog](#)
[Contacto](#)

1) Crear un Logo de la marca

2) Agregar un navbar

3) A partir de cierto ancho de pantalla, cambiar(a gusto, usando media queries) la manera en la que se muestran las opciones de navegación. Ejemplo:



**titulo**

## Administracion



- 1) Agregar un encabezado de Administracion
- 2) Una foto relacionada al negocio de Automoviles.

**form**

Información Del Anuncio

**TITULO:**

**TRANSACCION:**  

☒ VENTA

☐ ALQUILER

**DESCRIPCION:**

**PRECIO:**

 Puertas

 KMs

 Potencia

 **Guardar**

- 1) Respetar los controles de la imagen (el posicionamiento es a gusto).
- 2) Agregar iconos a los input de características del automóvil.

3)Agregar icono al botón de “Guardar”.

### **tabla**

Guardar

titulo	transaccion	descripcion	precio	num_puertas	num_KMs	potencia
Jeep Renegade	venta	Jeep Renegade 2017	1.000.000	5	10.000	120
Renault Megane	venta	Renault Megane 2014	800.000	5	80.000	140

1)pintar por css el encabezado

2)centrar la tabla en la pantalla

3)No mostrar el id por pantalla.

### **footer**

Nosotros	Anuncios	Blog	Contacto	Todos los Derechos Reservados 2019 ©
----------	----------	------	----------	--------------------------------------

1)Respetar posicionamiento

### **spinner**



1)Agregar un spinner acorde al negocio de Automóviles.

### **Javascript**

1)Utilizar una clase “Anuncio” que contenga los atributos:

- id
- titulo
- transaccion

- descripcion
- precio

2) Utilizar una clase "Anuncio\_Auto" que herede de "Anuncio" con el resto de los atributos necesarios

## Backend

- El backend será en localStorage. Agregar delay de 3 segundos en cada operación, de manera de poder visualizar el spinner.

## PAGINA PRINCIPAL

Como ya tenemos nuestro ABM funcionando, ahora se deben mostrar los elementos almacenados en la página principal de la aplicación.

Crear una página principal que consuma los datos del localStorage y genere dinámicamente los anuncios.



- Agregar una imagen de background que ocupe el 100% del viewport.
- Mantener el mismo navbar que en la página de administración.

# Autos en Venta y Alquiler

## Jeep Renegade

Jeep Renegade 2017

\$1,000,0000



5



10.000



120

Ver Vehiculo

## Renault Megane

Renault Megane 2014

\$800,0000



5



80.000



140

Ver Vehiculo

- Para cada elemento almacenado en localStorage, crear un artículo en la sección principal de la página.
- El diseño es a gusto.

### RECUPERATORIO 1

- Agregar al formulario algún campo que soporte checkboxes(multi selección). Los valores seleccionados deberán ser parte del ABM existente.
- Todos los campos del formulario deberán tener validaciones de “requerido” para poder submitir el formulario.
- Por cada acción (alta/baja/modificación) se deberá mostrar una alerta custom en pantalla. **No usar el “alert” por default.**

- Cada control input debe ser del “type” que corresponda según la información a ingresar.
- Usar media queries para ocultar la imagen de “Administración” en pantallas pequeñas.

## **SEGUNDO PARCIAL**

### **Map/Reduce/Filter**

- Agregar una sección que permita filtrar la tabla que se muestra por pantalla por tipo de transacción, y muestre por pantalla el promedio de precio(solo mostrar en caso de que se seleccione “venta” o “alquiler”). Mostrar “N/A” si no hay filtros aplicados.
- Agregar controles que permitan seleccionar qué columnas de la tabla mostrar por pantalla

**FILTRAR TRANSACCION**

**PROMEDIO**  

900,000

TITULO

TRANSACCION

DESCRIPCION

PRECIO

NUM\_PUERTAS

NUM\_KMS

☒

☒

☒

☒

☒

☒

POTENCIA

☒

id	titulo	transaccion	descripcion	precio	num_puertas	num_KMs	potencia
1	Jeep Renegade	venta	Jeep Renegade 2017	1000000	5	10.000	120
2	Renault Megane	venta	Renault Megane 2014	800000	5	80.000	140

### **Bootstrap**

- Utilizar la librería de Bootstrap y su esquema de columnas donde crea conveniente para lograr un comportamiento “responsive”.
- Utilizar entre 5 y 10 clases de Bootstrap(además de las ya utilizadas en el punto anterior).

### **Backend**

- Reemplazar la capa de acceso a datos por llamadas AJAX al servidor.
- Utilizar **xmlhttprequest** en al menos dos de las cuatro operaciones.
- Utilizar promesas para el “get” de la página principal.
- El spinner debe ser visible mientras se realizan dichas operaciones.