

Universidad Tecnológica Nacional
Facultad Regional Avellaneda



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

Materia: Laboratorio III

Apellido:		Fecha:	8/5/2018
Nombre:		Docente ⁽²⁾ :	Baus/Mutti
División:		Nota ⁽²⁾ :	
Legajo:		Firma ⁽²⁾ :	
Instancia ⁽¹⁾ :	<div style="display: flex; justify-content: space-around; padding: 2px;"> PP X RPP SP RSP FIN </div>		

(1) Las instancias validas 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.

Realizar una aplicación de blog.

- La aplicación deberá contar con una página principal donde se muestren los posts. Esta página es muy similar a la ya realizada previamente en la cursada. La unica modificacion es que los posts deberán ser cargados dinámicamente, teniendo en cuenta los posts que fueron dados de alta en la pagina de administracion.
- También deberá contar con una página de administración(**admin.html**) que nos permita dar de alta, baja y modificación de nuestros posts en el blog.
- Se usarán las siguientes tecnologías vistas en clase: HTML5, CSS3, y Javascript.
- El servidor será provisto en un archivo llamado server.js y no deberá ser modificado. Para correrlo, bastará con correr por línea de comando: "node server". El servidor quedará escuchando en el puerto 3000.
- El navegador por default va a devolvernos **index.html** al apuntarle al puerto 3000 de localhost. Habrá un link que nos redireccione a admin.html.
- La aplicación deberá Implementar un spinner al realizar la llamada AJAX que recupera los posts del servidor.

BLOGDEVIAJES

TU COMPAÑERO PARA TU PRÓXIMO VIAJE

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

Titulo

Articulo

Mas

[Guardar](#)

Otros posts

[Entrada1](#)[Entrada2](#)[Entrada3](#)[Entrada4](#)[Entrada5](#)

- La misma página deberá mostrarnos una tabla generada dinámicamente con todos los posts activos, y con botones que nos permitan realizar las bajas y modificaciones.

MIOS ARTICULOS

ID	Fecha	Titulo	Articulo	Modificar	Borrar
6	2018-4-22 11:27:31	asd	asd	Modificar	Borrar
9	2018-4-22 11:43:11	g	g	Modificar	Borrar
10	2018-4-22 11:56:45	f	f	Modificar	Borrar
12	2018-4-23 20:40:24	asda	iasda	Modificar	Borrar

SERVER:

Contrato con el servidor:

- ALTA:**
request(POST):
data = {
 "titulo": [valor],
 "articulo": [valor],
 "mas": [valor],
 "collection": "posts"
}

response:

```
response = {  
    message: "Alta exitosa",  
    data: [array con todos los posts]  
}
```

- **BAJA:**

request(POST):

```
xhr.send(JSON.stringify({"collection": "posts", "id": id}));
```

response:

```
res.send({"message": "Baja exitosa"});
```

- **MODIFICACIÓN**

request(POST):

```
data = {  
    "titulo": titulo.value,  
    "articulo": articulo.value,  
    "mas": mas.value,  
    "collection": "posts",  
    "id": postAModificar.id,  
    "active" : postAModificar.active,  
    "created_dttm" : postAModificar.created_dttm  
}
```

response:

```
res.send('Modificacion exitosa');
```

- **TRAER**

request(GET):

Pasar por url el atributo "collection" con el valor "posts".

response:

```
res.send({"message": "Carga exitosa", "data": array});
```

NOTA: En todos los casos, usar el siguiente request header para las peticiones AJAX:

```
xhr.setRequestHeader("Content-Type", "application/json");
```