



UNIVERSIDAD INTERAMERICANA PARA EL DESARROLLO

CAMPUS CANCÚN

“Etiquetas de formulario”

“Notificaciones en Node.js”

“Procedimientos almacenados”

“Data Tables”

DESARROLLO ORIENTADO A OBJETOS

ELABORADO POR:

MARIO DAVID MORALES CASTRO

**LIC. INGENIERÍA EN TECNOLOGÍAS DE
INFORMACIÓN Y COMUNICACIÓN**

MARZO 2019

Etiquetas de formulario

Una de las áreas en las que Spring destaca es en la separación de las vistas con el resto del framework MVC. Spring ofrece múltiples soluciones para realizar un enlace de datos una de ellas son las etiquetas de formulario. Cada etiqueta da soporte a un grupo de atributos, los cuales corresponden con su correspondiente etiqueta HTML.

La librería de etiquetas está integrada en Spring MVC, dando a las etiquetas acceso a los objetos y referencias de datos de las que se ocupa el controlador.

Para usar las etiquetas de esta librería se agrega la siguiente directiva al comienzo de la página o vista JSP:

```
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %>
```

Podemos encontrar las siguientes etiquetas de formulario dentro de Spring MVC:

- **Input Tag**

Por default esta etiqueta viene ligada a un input de tipo "text"

```
<form:input path="nombre" />
```

A partir de Spring 3.1 se pueden utilizar tipos específicos tales como email, date u otros.

```
<form:input type="email" path="email" />
```

```
<form:input type="date" path="dateOfBirth" />
```

- **Password Tag**

Esta etiqueta viene ligada a un input de tipo "password" que oculta los valores ingresados dentro del campo.

```
<form:password path="password" />
```

- **Textarea Tag**

Esta etiqueta muestra un input de tipo Textarea.

```
<form:textarea path="notas" rows="2" cols="10"/>
```

- **Checkbox(s) Tag**

Esta etiqueta viene ligada a un input de tipo "checkbox" pero ofrece un acercamiento diferente al de html.

```
<form:checkbox path="lenguajes" />
```

Esta etiqueta como tal genera solo un checkbox con un valor booleano.

Para mostrar múltiples checkboxes se puede utilizar la etiqueta "checkboxes".

```
<form:checkboxes items="{favouriteLanguageItem}" path="favouriteLanguage" />
```

- **Radiobutton Tag**

Similar a los checkbox, esta etiqueta muestra un input de tipo radio con

Hombre: <form:radiobutton path="genero" value="H"/>

Mujer: <form:radiobutton path="genero" value="M"/>

Pero también es posible generarlos de manera múltiple con la etiqueta "radiobuttons"

<form:radiobuttons items="{jobItem}" path="job" />

- **Select Tag**

Esta etiqueta muestra un input de tipo select

<form:select path="country" items="{countryItems}" />

Viene acompañada de la etiqueta option para indicar los elementos que queremos mostrar dentro del select

<form:select path="book">

<form:option value="-" label="--Please Select--"/>

</form:select>

- **Hidden Tag**

Genera un input escondido "Hidden"

<form:hidden path="id" value="12345" />

Notificaciones Node.Js

Node.js es un entorno de ejecución de JavaScript que permite correr código de JavaScript del lado del servidor y no en el navegador como siempre.

Eso es una explicación a grandes rasgos, pero sus aplicaciones pueden ir mucho más allá, ya que JavaScript en sí mismo al ser un lenguaje completo puede ser usado en varios contextos similar a otros lenguajes de programación.

Hoy en día, el uso de notificaciones se ha globalizado en el mundo de las aplicaciones web. Las notificaciones pueden ser provocadas al momento de abrir una aplicación o pueden ser “empujadas” del servidor hacia el usuario incluso si la aplicación no está corriendo. Se pueden usar para hacer recordatorios al usuario en un cierto momento del día.

Tipos de notificaciones:

Notificación: Un mensaje que es desplegado al usuario fuera de la UI normal de la aplicación (o navegador)

Mensaje Push: Un mensaje enviado desde el servidor al cliente

Notificación Push: una notificación creada en respuesta a un mensaje Push.

API Push: una interfaz usada para subscribir una aplicación a un servicio Push y recibir mensajes Push dentro del servicio

Web Push: un término informal para referirse al proceso o componentes involucrados en el proceso de *pushing* de un servidor a un cliente en la web.

Push Service: un sistema que rutea mensajes Push de un servidor al cliente. Cada navegador implementa su propio Push service.

Cuando se recibe un mensaje el servicio despierta solo lo suficiente para desplegar la notificación y entonces regresa a dormir. El servicio puede estar atento de las interacciones con las notificaciones en segundo plano sin utilizar recursos.

Node-notifier: Es un módulo de Node.js capaz de enviar notificaciones de escritorio a través de cualquier sistema operativo que se puede obtener en el siguiente link: <https://github.com/mikaelbr/node-notifier>

Sus requerimientos de software son:

MacOS: Versión mayor o igual a la 10.8

Linux: notify.osd o libnotify.bin instalador

Windows versión mayor o igual Windows 8



Procedimientos almacenados

Un procedimiento almacenado es un conjunto de sentencias con un nombre asignado que están almacenadas en un sistema manejador de bases de datos relacionales como un grupo, así pueden ser reutilizadas y compartidas por varios programas.

Los procedimientos almacenados pueden acceder o modificar datos en una base de datos, pero no están ligados a una base de datos u objeto específico, el cual ofrece ventajas para el sistema en el que se vaya a utilizar.

Para utilizar un procedimiento almacenado, primero es necesario crearlo y entonces correrlo. Para correrlo este se "llama" y este suministra cualquier parámetro que se pueda requerir. El procedimiento almacenado se ejecutará usando los parámetros en cualquier manera que el código lo especifique.

Sintaxis para crear un procedimiento almacenado:

```
CREATE PROCEDURE sp_name(p_1 INT)
```

```
BEGIN
```

```
...código...
```

```
END;
```

El cuerpo del procedimiento va entre el BEGIN y END. Estas palabras son usadas para escribir instancias que se pueden incluso anidar entre ellas.

En la mayoría de los casos también puede ser necesario crear procedimiento con un delimitador:

```
DELIMITER //
```

```
CREATE PROCEDURE sp_name(p_1 INT)
```

```
BEGIN
```

```
...code goes here...
```

```
END //
```

```
DELIMITER;
```

Este permite decirle a MySQL diferenciar la instancia del procedimiento almacenado.

Para llamar a un procedimiento almacenado se utiliza la instancia:

```
CALL sp(1);
```

Data Tables

Data tables es un poderoso plugin para JQuery para la creación de tablas interactivas. Provee de búsqueda, filtrado y paginación sin ninguna configuración.

Data tables muestran información de manera fácil, la librería de puede obtener a través de del siguiente enlace: <https://www.datatables.net/download/>

Permite administrar el contenido de tablas de manera jerárquica, alfabética u otras maneras similares.

Las Data tables pueden incluir tres o mas columnas. Una fila de encabezado al principio e cada columna, y todas las filas subsecuentes que contengan datos. Si los usuarios necesitan interactuar con una fila de datos, checkboxes deben acompañar cada fila.

<input type="checkbox"/> Dessert (100g serving)	Calories	Fat (g)	Carbs (g)	Protein (g)
<input type="checkbox"/> Frozen yogurt	159	6.0	24	4.0
<input type="checkbox"/> Ice cream sandwich	237	9.0	37	4.3
<input type="checkbox"/> Eclair	262	16.0	24	6.0
<input type="checkbox"/> Cupcake	305	3.7	67	3.9
<input type="checkbox"/> Gingerbread	356	16.0	49	0.0
<input type="checkbox"/> Jelly bean	375	0.0	94	0
<input type="checkbox"/> Lollipop	392	0.2	98	6.5
<input type="checkbox"/> Honeycomb	408	3.2	87	4.9

Incluso es posible añadirle temas para cambiar su apariencia y adaptar la tabla a según se requiera.

Conclusiones

Etiquetas de formulario

Spring MVC permite manejar los enlaces de datos entre sus vistas, el controlador y modelo de una manera eficiente que reemplaza y adapta las clásicas inputs de HTML con etiquetas, que permiten su reutilización y versatilidad para el desarrollador.

Notificaciones Node.Js

Las notificaciones Push permiten a una aplicación extenderse más allá del navegador, y son una manera de entrar en contacto con el usuario. Es posible sacar mucho provecho de ellas para poder mantener al usuario arraigando a una aplicación y servicio.

Procedimientos almacenados

Los procedimientos almacenados permiten ejecutar instancias dentro del servidor de base de datos y permite añadir distintas medias de seguridad u otras funciones a las bases de datos relacionales.

Data tables

Es un plugin que simplifica y facilita la utilización de tablas para poder mostrar y manipular información en una base de datos, si versatilidad lo hace una opción muy factible para esta tarea.

Bibliografía

- View technologies. 03/03/2019, de Spring.io Sitio web: <https://docs.spring.io/spring/docs/3.2.x/spring-framework-reference/html/view.html>
- baeldung. (06/11/2018). Exploring SpringMVC's Form Tag Library. 03/03/2019, de baeldung.com Sitio web: <https://www.baeldung.com/spring-mvc-form-tags>
- Manuel Kiessling & Herman A. Junge. (29/01/2015). JavaScript y Node.js. En El Libro para Principiantes en Node.js» Un tutorial completo de node.js (40). <http://www.nodebeginner.org/indexes.html>: http://ayllusolar.cl/wp-content/uploads/2016/08/node_js_Guia_Principiantes.pdf.
- Mikael Brevik. node-notifier. 03/03/2019, de github.com Sitio web: <https://github.com/mikaelbr/node-notifier>
- David Walsh. (09/02/2019). System Notifications with Node.js. 03/03/2019, de davidwalsh.name Sitio web: <https://davidwalsh.name/system-notifications-node>
- node-notifier. 03/03/2019, de npmjs.com Sitio web: <https://www.npmjs.com/package/node-notifier>
- Margaret Rouse. stored procedure. 03/03/2019, de techtarget.com Sitio web: <https://searchoracle.techtarget.com/definition/stored-procedure>
- MySQL Stored Procedures. 03/03/2019, de quackit.com Sitio web: https://www.quackit.com/mysql/tutorial/mysql_stored_procedures.cfm
- Data tables. 03/03/2019, de material.io Sitio web: <https://material.io/design/components/data-tables.html>