

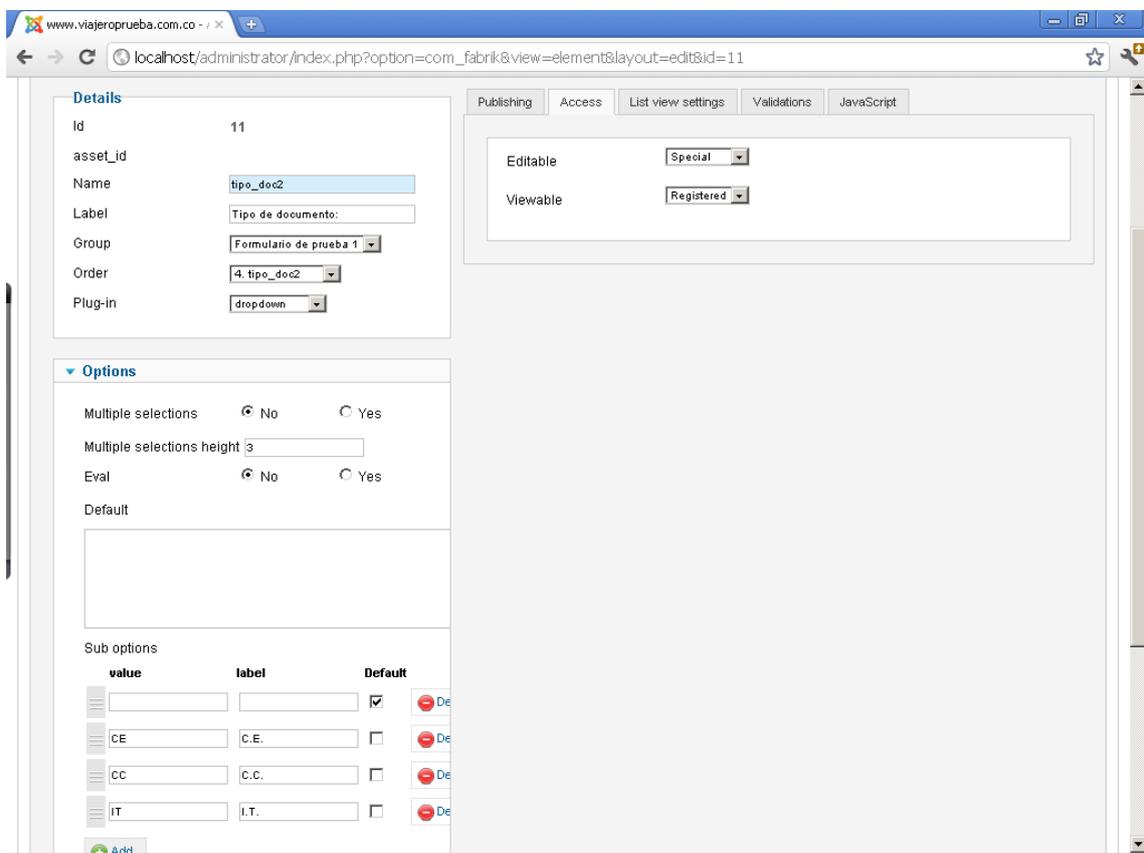
## Validación de campos vacíos “notempty” y esconder o mostrar elementos – Fabrik – 2

### Validar que no quede el campo vacío “notempty”

Para empezar, lo que en mi caso quería hacer era que si el usuario no completa el campo o no selecciona una opción requerida, no grabe los datos y me muestre un error. La mayoría lo hacen por defecto el mismo Fabrik.

Lo primero que vamos a hacer es crear un nuevo elemento de tipo “**Dropdown**”.

Para eso vamos a Components > Fabrik > Elements > New...



Estaremos en una pantalla parecida a esta (pero sin los datos colocados).

Le asignamos un nombre de fácil ubicación y su Label correspondiente (el que nos va a mostrar el formulario en la web).

Seleccionamos el grupo al que pertenecerá este elemento y el orden en que éste aparecerá.

En “plugin” realmente se elige el tipo de elemento... en este caso seleccionaremos “dropdown” y esperaremos a que se desplieguen las opciones que aparecen debajo.

Aparecen varias opciones:

**“Multiple selections”** nos permite indicar si queremos que el usuario pueda elegir más de una opción del menú desplegable (en este caso no nos interesa, por lo tanto lo dejamos en NO).

Por el momento vamos a saltar las demás opciones hasta donde dice **“Sub Options”**.

Ésta aparecerá vacía por defecto, entonces le haremos click en **“add”** y en la columna **“Value”** pondremos el valor que tendrá la primer opción (en mi caso **“CE”**, este tipo de valores a mi parecer es mejor ponerlos en mayúsculas para que al momento de imprimir o bien crear un código personalizado las mayúsculas se ven más claras y se distinguen más de el resto del código). En la columna **“Label”** simplemente pondremos cómo queremos que se vea la opción cuando el usuario tenga que seleccionar una (es este caso puse **“CE”** también).

Como verán en la imagen, creo una opción (la primera) en blanco y la seleccioné como **“default”** para la opción que aparezca cuando cargue el formulario sea en blanco, porque quiero obligar al usuario a tener que seleccionar una opción, dado que le pondremos una validación **“notempty”** para que no se puedan dejar campos vacíos (¿se entiende?, lea de nuevo si no).

Seguimos agregando opciones, como se ve en la pantalla, para tener varias.

Más debajo de las **“Sub Options”** hay 3 opciones más... esas las dejamos en **“No”** por el momento, dado que no queremos que los usuarios modifiquen o agreguen nuevas opciones en nuestro formulario (sólo lo manejaremos nosotros).

En las pestañas de la derecha:

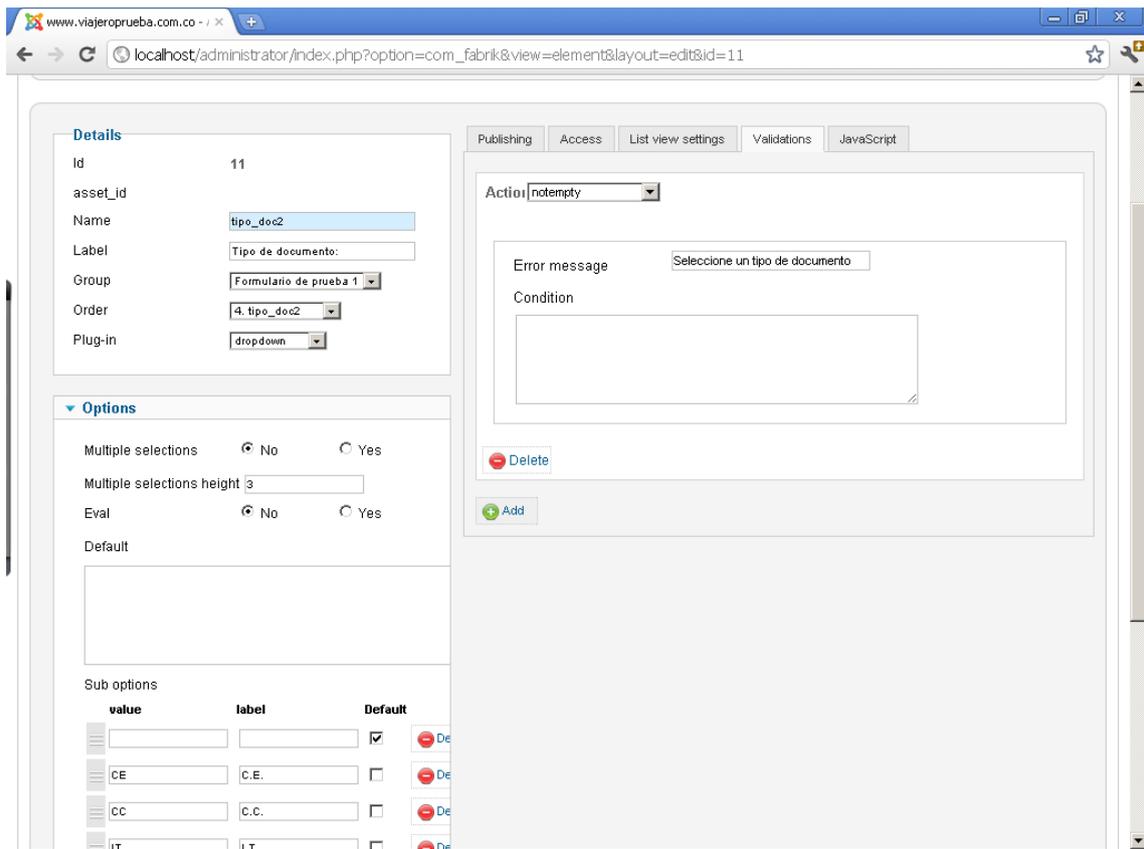
En **“Publishing”** dejamos el estado en **“yes”**.

En **“Access”**, en este caso no queremos que cualquier usuario lo vea, así que dejamos que sólo los registrados lo puedan ver y que sólo los admins puedan editarlo.

En **“List view settings”** solo activaremos la primera opción de mostrar el elemento en la lista de elementos.

Ahora empieza lo interesante...

En **“Validations”** haremos click en **“add”** y nos aparecerá lo siguiente...



En las opciones desplegadas de **“Action”** seleccionaremos **“notempty”** validación para que no quede este elemento vacío o sin seleccionar alguna opción.

En **“Error Message”** colocaremos el mensaje que queremos que nos muestre en el caso de que el elemento quede vacío. Y por ahora en **“Condition”** no pondremos nada. Pero éste último, te permite ingresar código personalizado en formato PHP que tiene que retornar **“false”** o **“true”** condicionando si la validación se ejecuta o no.

Si se fijan bien, el botón **“add”** sigue ahí, por lo que nos permitirá seguir agregando validaciones.

Por defecto el Fabrik trae sólo unos 3 o 4 tipos de estas validaciones, pero en otro documento les contaré cómo agregar aún más.

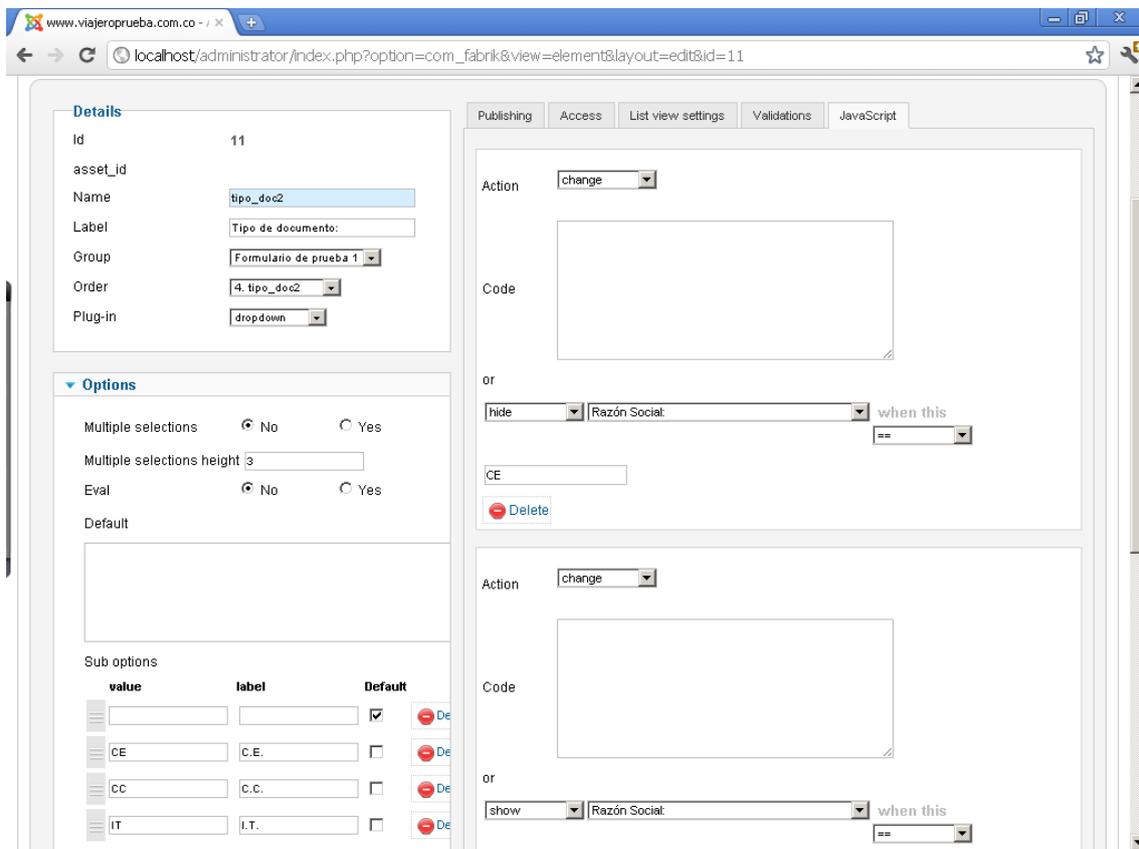
Todo este proceso de validación pueden hacerlo con todos los elementos del formulario que quieran que sean requeridos y comprobados en tiempo real (al tiempo del llenado del formulario).

Ahora pasemos al siguiente paso.

### **Esconder o mostrar elementos en Fabrik**

En mi caso particular, necesitaba que cuando se seleccione una opción del menú desplegable **“tipo\_doc2”**, esconda o muestre el elemento **“razón\_social”**. Veamos cómo se hace.

En la misma configuración iremos la pestaña “JavaScript”.



Esta pestaña nos permite asignar un evento al elemento. Cuando hablo de eventos me refiero a cualquier cosa que suceda en tiempo real con este elemento (Ej.: un click sobre éste).

La idea es decirle que cuando se seleccione una opción del menú desplegable revise el valor seleccionado y esconda o muestre el elemento “Razon Social: “.

Para eso haremos click en “add” (porque debería estar vacío por defecto) y nos aparecerán las siguientes opciones:

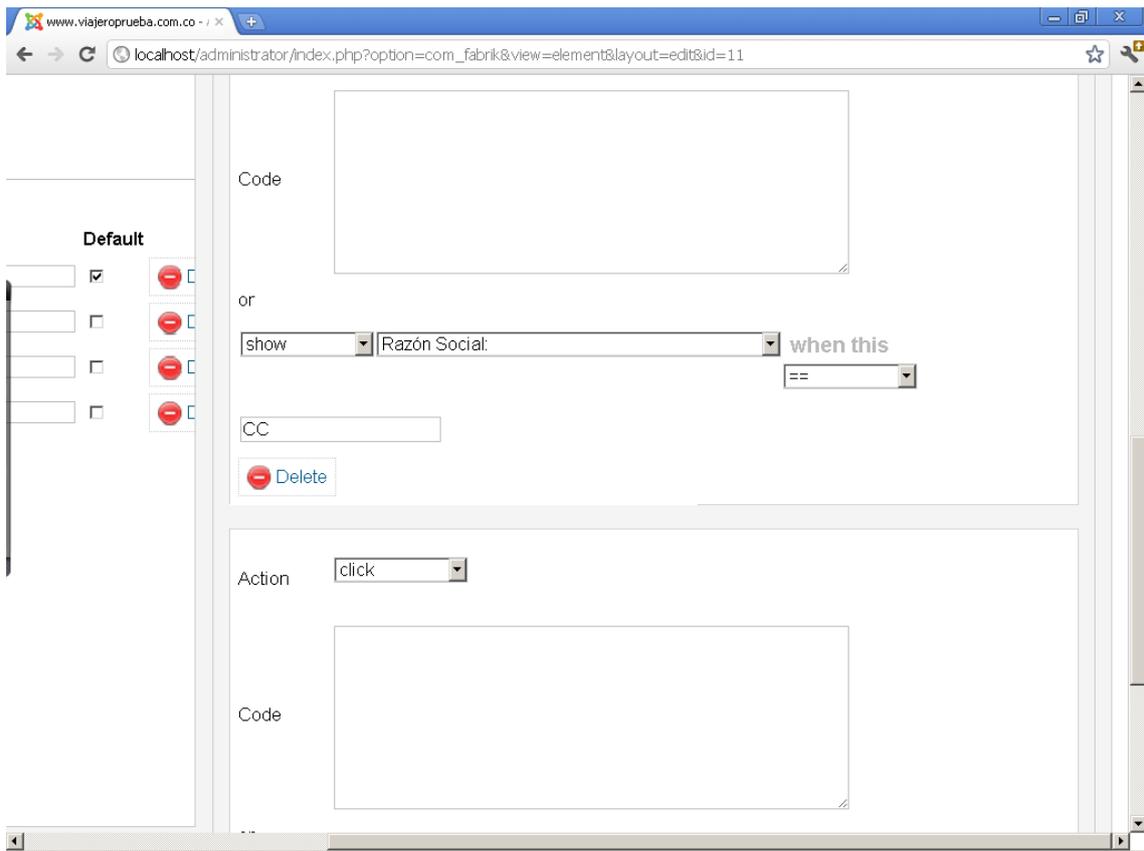
“Action” nos muestra un listado de eventos que podemos asignarle al elemento.

Si buscan las definiciones o explicación de estos eventos les servirá mucho para conocer las distintas funcionalidades y aplicaciones.

En mi caso seleccioné “Change” (que viene del evento onChange de JavaScript), que básicamente lo que hace es detectar cualquier cambio sobre el objeto (o elemento), ya sea la ubicación del puntero, un click, salir del elemento con tabulador, entre otras, y es lo que necesito.

Ahora, si miramos bien, veremos que debajo dice “Code” y aún más abajo dice “or”. Lo que nos da la opción de colocarle un código personalizado en JavaScript (JS Custom Code) “o” utilizar las opciones que ya nos brinda Fabrik.

Por ahora vamos a utilizar las opciones que esta herramienta nos da.



Entonces le decimos:

Si hay un cambio en el elemento “tipo\_doc2”, “**show**” (muestre) “**Razón Social:**”(el otro elemento) cuando éste “**=**” (sea igual) a “**CC**”.

Básicamente le decimos que cuando el usuario seleccione “**CC**” nos muestre el campo Razón Social.

Hacemos click en “**add**” abajo del todo y hacemos lo mismo: “**Change**” pero que ahora diga “**hide**”(esconda) – “**Razon Social**” – cuando este sea igual a “**CE**”. Salvamos y vamos a ver en el navegador cómo esconde y muestra este campo!.

Hasta ahora muy fácil! ¿no?.

Bueno, el problema que suele surgir al hacer este tipo de eventos es el siguiente (**lee atentamente**):

Lo que pasa es que... cuando se esconde el campo “**Razón Social**” (en este caso), el campo sigue pidiendo la validación “**notempty**” que alguna vez le colocamos y el hecho de

“**esconderlo**” no lo “**deshabilita**”. Entonces lo que pasaría si lo dejamos así, es que cuando el usuario elija “**CE**” y “**Razón Social:**” se esconda, una vez que termine de completar los demás

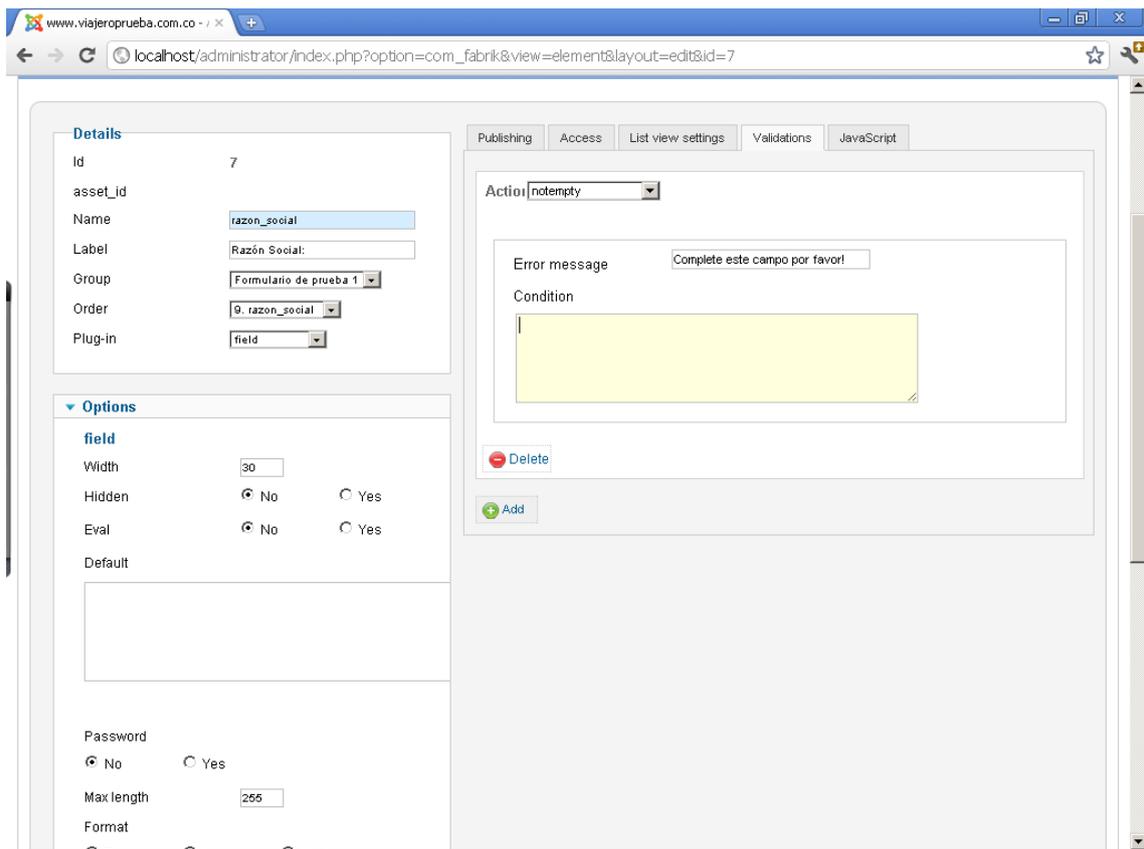
campos y presione **“Enviar”**, lo devolverá y le mostrará el campo **“Razón Social”** y le dirá que **“Debe ingresar una razón social”**.

Pero no importa porque existen varias soluciones aunque por ahora les diré la que me pareció más práctica.

### **Deshabilitar validación “notempty” de un elemento cuando se esconde**

Siguiendo el caso que veníamos explicando previamente, vamos a pasar a solucionar el problemita de la validación **“notempty”** cuando se esconde el elemento que la tiene.

En este caso, vamos a ir a Components > Fabrik>Elements> razón\_social (o el elemento que se esconde). Y vamos a la pestaña **“Validations”**.



Ahí ya tendríamos la acción **“notempty”** seleccionada y con su respectivo mensaje de error como muestra la imagen.

Ahora lo que vamos a hacer es colocarle un código personalizado (**PHP**) en el campo “**Condition**” que, como habíamos dicho en tutoriales anteriores, será una condición que retornará verdadero o falso, siendo verdadero la ejecución de la validación “notempty” y falso para evitar que dicha se ejecute.

El código es el siguiente:

```
if ({nombre_de_tabla__tipo_doc2}=='CE') {return false;}else{return true;}
```

//en este caso le estamos diciendo: “Si el valor del menu desplegable es CE retorne falso y evite la validación (ya que estaría escondido) o si no retorne verdadero y corra la validación (porque estaría a la vista).

Y listo! Salven y cierren. Borren cookies del navegador de prueba de la web, actualicen y prueben!.