

Modifica della visibilità dei campi

Si può modificare la visibilità dei vari campi (valore di \$readonly) e altre variabili nelle varie modalità (ListView, EditView...). Per registrare una nuova "Vista" usare il metodo:

SDK::addView(\$module, \$src, \$mode, \$success);

\$module : il modulo in cui applicare la vista

\$src : il file php

\$mode : la modalità di applicazione della regola (vedere sotto)

\$success: cosa fare dopo l'applicazione della regola (vedere sotto)

Le Viste definite per ogni modulo vengono applicate nell'ordine in cui sono registrate. Quando si registrano, vengono aggiunte in coda a quelle già definite per quel modulo. Il parametro \$mode ha senso solo se si modifica la variabile

\$readonly, e ammette i seguenti valori:

"constrain" : forza il valore di \$readonly ad assumere il nuovo valore dato nello script.

"restrict" : cambia il valore di \$readonly solo verso un valore più restrittivo (da 1 a 99 o 100, da 99 a 100, non viceversa)

Il parametro \$success invece può essere:

"continue" : dopo l'applicazione della vista, continua con la successiva

"stop" : se la vista restituisce \$success = true, non vengono eseguite altre regole

All'interno degli script sono disponibili le seguenti variabili:

\$sdk_mode : uno tra "" (create), "edit", "detail", "popup_query", "list_related_query", "popup", "related", "list", "mass_edit"

\$readonly : il valore di readonly per il campo che si sta per scrivere (1, 99, 100 rispettivamente lettura/scrittura, sola lettura, nascosto)

\$col_fields: valori dei campi, solo per \$sdk_mode = "edit", "detail" e ""

\$fieldname o *\$fieldName*: nome del campo corrente

\$current_user: l'utente corrente

Ed è possibile restituire la variabile \$success con true o false.

A seconda della modalità (ListView, EditView, ...) ci sono diversi modi per modificare le query e diverse variabili disponibili.

Modalità	Valore di <code>\$sdk_mode</code>	Variabili disponibili	Note
CreateView	""	<code>\$col_fields</code> <code>\$current_user</code>	1
EditView	"edit"	<code>\$fieldname</code> <code>\$readonly</code> <code>\$success</code>	
DetailView	"detail"		
MassEdit	"mass_edit"		
PopupQuery	"popup_query"	<code>\$sdk_columns</code> <code>\$success</code>	2
List/RelatedQuery	"list_related_query"	<code>\$sdk_columns</code> <code>\$success</code>	
Popup	"popup"	<code>\$current_user</code> <code>\$fieldname</code>	3
Related	"related"	<code>\$sdk_columnnames</code> <code>\$sdk_columnvalues</code> <code>\$readonly</code>	4
List	"list"	<code>\$success</code>	

Note:

1. In queste modalità i valori dei campi sono in `$col_fields[nomi-dei-campi]`
2. Queste modalità servono per modificare la query in modo da poter prelevare campi aggiuntivi. In `$sdk_columns` ci sono le colonne del db da aggiungere alla query. Includere poi il file "modules/SDK/AddColumnsToQueryView.php"
3. Per prelevare valori di altri campi, specificati nelle modalità PopupQuery e ListRelatedQuery, scriverli nell'array `$sdk_columnnames`, includere il file "modules/SDK/GetFieldsFromQueryView.php" e prenderli poi da `$sdk_columnvalues`
4. Nel caso della related "Storico attività" sono disponibili solo le variabili `$recordId` e `$readonly` e si applicano all'intera riga, non al singolo campo.

Per de-registrare la vista usare:

SDK::deleteView(\$module, \$src);

\$module : il modulo associato

\$src : il file php

Hooks

include/ListView/ListViewController.php

include/utills/DetailViewUtils.php

include/utills/EditViewUtils.php

include/utills/ListViewUtils.php

include/QueryGenerator/QueryGenerator.php

Revision #3

Created 22 February 2020 08:51:32 by ddalmaso

Updated 21 February 2022 11:30:09 by ddalmaso