GwAction

executeGwAction()

This will cause the execution of the gwAction, with name gwActionName, existing under the gwClass with name gwClassName the gwAction will be invoked with the declared set of parameters, availables under the specified variables names

Parametri

- **gwActionName** String, required
- **gwClassName** String, required
- **functionParametersNames** String[], optional default []
- **functionParameters** Object[], optional default []

Esempi

example.js

```
var gwActionName = 'gwActionName';
var gwClassName = 'gwClassName';
var functionParametersNames = ['param1', 'param2'];
var functionParameters = ['abc', 123];

executeGwAction(gwActionName, gwClassName, functionParametersNames,
functionParameters);
```

Suggerimenti

DA REPORT

Se si hanno più pulsanti che aprono la stessa azione della stessa classe, ma con diverso numero di parametri passati

es:

```
// pulsante 1
var avoidReturn = executeGwAction('open_detail', 'sec_service',
['itemId','layoutName'], ['newItem','new_service']);
```

```
// pulsante 2
var avoidReturn = executeGwAction('open_detail', 'sec_service',
['itemId','layoutName','cod_order'], ['newItem','edit_service','ORD1']);
```

è necessario passare lo stesso numero di parametri, ovvero il numero massimo (nell'esempio 3 per entrambi, anche se la prima non utilizza il terzo parametro), altrimenti se viene eseguita per prima l'azione con meno parametri verrà passato il numero di parametri di quell'azione anche quando si proverà a lanciare una qualsiasi delle altre azioni.

Diventa quindi:

es:

```
// pulsante 1
var avoidReturn = executeGwAction('open_detail', 'sec_service',
['itemId','layoutName','cod_order'], ['newItem','new_service']);
```

```
// pulsante 2
var avoidReturn = executeGwAction('open_detail', 'sec_service',
['itemId','layoutName','cod_order'], ['newItem','edit_service','ORD1']);
```

executeGwActionOnGwClassDetail()

Execute js action in the gwClassDetail context. Like gwAction in toolbal, or in '...' button for the single record. ACL and DACL are checked server side.

Parametri

- **gwClassName** String, required
- **itemId** String, required
- **gwActionName** String, required

Esempi

example.js

```
var itemId = '123';
var gwClassName = 'gwClassName';
var gwActionName = 'gwActionName';
executeGwActionOnGwClassDetail(gwClassName, itemId , gwActionName);
```

checkGwActionPermissions()

This will check static and dynamic permissions for the gwAction, with name gwActionName, existing under the gwClass with name gwClassName The callback is invoked with argument the server side computed data If the gwAction with the specified name is not found a warn is however logged If gwActiveUser ha no execute permission upon the gwAction with the specified name a warn is however logged

Parametri

- **gwActionName** String, required
- **gwClassName** String, required
- **itemId**, String, required
- **callback**, Object, required. Invoked at end with an incoming parameter Object done this way: {hasAction: true, gwAction: {..}, hasPermission: true}

Esempi

example.js

```
var gwActionName = 'gwActionName';
var gwClassName = 'gwClassName';
var itemId = '123';
var callback = function(
    parmas //Object
){
    var hasAction = parmas.hasAction;
    var gwAction = parmas.gwAction;
    var hasPermission = parmas.hasPermission;
    var item = parmas.item;
    if(hasAction && gwAction && hasPermission){
        //TODO
    }
};

checkGwActionPermissions(gwActionName, gwClassName, itemId, callback);
```

From:
https://wiki.geowebframework.com/ - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=custom:api_js_gw_action

Last update: **2023/02/13 17:55**