

Read Me

Rb87

Contents

1	Menus	1
1.1	General Info	1
1.2	How to add menu	2
1.3	Technical informations	2
1.3.1	How ajax call work ?	2
1.4	List of functions and files	4
2	Publications	4
2.1	•	4

1 Menus

1.1 General Info

Menus are handled by the `menus.php` file. Each plugin has its own entry. There are two types of entries : *menu* and *submenu*. Each *menu* is associated to a CPT, *submenus* are not. Each *menu* and *submenu* entry call a function named `menu_name_page_setup` lying inside `menu_name/index_menu_name.php`. They are loaded if the corresponding plugin is activated, this is handle by the option `activated_plugins`.

There is one exception : the **RbWeb** menu, which is always present. It corresponds to the settings page that handle all other menus. It calls the function `Handle_all_menus` in the `activated_plugins.php` file.

List of *menus* and *submenus*

1. Publications

- (a) Arxiv
- (b) Bibtex
- (c) Orcid
- (d) HAL
- (e) zotero

2. Members

3. Teams

4. Jobs

1.2 How to add menu

In order to add a new *menu*:

1. In `menus.php` add the following

```
1 if (in_array('NEW',get_option("activated_plugins"))){
2     add_action('admin_menu', 'add_menu_for_NEW');
3     function add_menu_for_NEWr() {
4         add_menu_page('NEW', //Menu Name
5             'NEW', // Menu label
6             'manage_options', // capability
7             'NEW', // slug
8             'NEW_page_setup', //function to call -> /NEW_folder/index_NEW.php
9             'NEW', // icon
10            '11' //position
11        );
12    }
13 }
```

Every 'NEW' must be changed.

2. In `activated_plugins.php` add, inside the `Handle_all_menus` functions

```
1 <div>
2     <?php on_off_button('Name of the NEW menu','id of the NEW menu');?>
3 </div>
```

`id of the NEW menu` will be included in the option `activated_plugins`. `Name of the NEW menu` is just the label written on the button.

3. In `index.php` add :

```
1 if (in_array('id of the NEW menu',get_option("activated_plugins"))){
2     include(plugin_dir_path(__FILE__).'NEW_folder/index_NEW.php');
3
4 }
```

the "id of the NEW menu" must be the same as the one included in the `activated_plugins.php` file.

4. Create a `NEW_folder`, within a `index_NEW.php` file, that must include the `NEW_page_setup` function
5. Update the `ReadMe` file list of menus.

1.3 Technical informations

Menus and plugins activations go through an ajax call. The 'send' button is useless, only used to refresh the page. Maybe it's not the best idea ?

1.3.1 How ajax call work ?

1. PHP file. Include the javascript file

```
1 function add_NEW_script(){
2     wp_register_script('NEW Slug of script', plugin_dir_url(__FILE__).'js/NEWn
3     wp_enqueue_script('NEW Slug of script');
4 }
```

¹ and use the hook for admin script :

```
1 add_action( 'admin_enqueue_scripts', 'add_NEW_script' );
```

2. PHP file (PHP→ JS) Localize the script... that means you can pass PHP data to javascript.

In the `add_NEW_script` function you add :

```
1 wp_localize_script( 'NEW Slug of script', 'NEW_name_of_variable_localized',  
2     array( 'ajaxurl' => admin_url( 'admin-ajax.php' ), 'var2'=>php_data ) );
```

Now you can use in the javascript script the `NEW_name_of_variable_localized` list of variables.

3. JS file (JS, JS↔PHP) Create the ajax script :

```
1 jQuery( document ).ready( function($) {  
2     php_data = NEW_name_of_variable_localized.var2 // This is where the  
3         localize is usefull ! (PHP→JS)  
4     $.ajax({  
5         url : NEW_name_of_variable_localized.ajaxurl,  
6         // method : 'POST', // GET par default  
7         action : NEW_ajax_action_name // (JS→PHP, this is where you can make  
8             the php through JS )  
9         success : function( return_data ) { //  
10             // console.log(return_data); (PHP→JS)  
11         },  
12         error : function( data ) { // en cas d'echec  
13             console.log( 'Erreur...' );  
14         }  
15     });  
16  
17 });
```

4. (PHP) Register the `NEW_ajax_action_name` and link it to your favorite php function

```
1 add_action( 'wp_ajax_NEW_ajax_action_name$', 'NEW_call_back_function' );
```

5. (PHP) And finally create `NEW_call_back_function` :

```
1 function NEW_call_backfunction(){  
2     do php stuff, send JSON info :  
3     wp_send_json_success( $php_data );  
4 }
```

For the menu case here are the references of the preceeding function :

- `add_NEW_script` = `activated_plugin_script_and_style` (`activated_plugins.php`)
 - `NEW Slug of script` = `activated_button_js`
 - `NEWname_of_file` = `activated_button_js.js` (in `plugin_dir_url(__FILE__).'js/'`)
 - `NEW_name_of_variable_localized` = `adminAjax`
- `NEW_ajax_action_name` = `activated_button`
- `NEW_call_back_function` = `activate_plugin_ajax`

¹Still does not understand the utility of the two functions `register` and `enqueue`

1.4 List of functions and files

PHP

- `index.php` use only for including other php file.
- `menus.php`
 - `add_menu_for_rbWeb`
 - `add_menu_for_publication`
 - * `add_submenu_for_arxiv`
 - * `add_submenu_for_Bibtex`
 - * `add_submenu_for_zotero`
 - * `add_submenu_for_orcid`
 - * `add_submenu_for_hal`
 - `add_menu_for_member`
 - `add_menu_for_job`
 - `add_menu_for_team`
- `activated_plugins.php`
 - `activated_plugin_script_and_style`
 - `activate_plugin_ajax` (ajax callback function)
 - `on_off_button($name,$id)` (on/off button for the rbWeb plugin setting page, \$name→ label shown, \$id once chosen it has to be similar in other pages eg. in `index.php`)
 - * `hide_and_show_buttons()` (js function)
 - `Handle_all_menus`

JS

- `activated_button_js.js`
 - `hide_and_show_buttons`
 - ajax call for updating option `activated_plugins`

CSS

- `css`
 - `activated_button_css.css` (style the on/off buttons)

2 Publications

2.1 •