Widgets Mon Moodle Amélioré

Les widgets de la page my améliorée

La page My améliorée utilise des widgets pour composer une structure personnalisée du tableau de bord de l'utilisateur. Ces widgets peuvent être appelés dans l'ordre de parution dans la configuration centrale du composant local `Ma page améliorée`

Les widgets sont conçus pour adopter une apparence de blocs, afin de ne pas perturber la logique d'affichage si des blocs sont mobilisés dans les espaces latéraux de Moodle.

Liste des widgets

- `me` : Mon identité (simple)
- `fullme` : Mon identité (développée)
- `my_caption` : Permet de renseigner une étiquette avec le contenu de son choix. S'affichera en haut.(étudiants et enseignants)
- `static_<n>` : Affichage d'un bloc d'un texte avec le message de son choix sur l'espace ou le widget est implanté
- `admin_stats` : Des données statistiques utiles (en tant qu' administrateur)
- `my_courses` : Mes cours (en tant qu'étudiant) (affichage en liste simple)
- `my_courses_slider` : Mes cours affichés dans un slider ou en grille (en tant qu'étudiant)
- `my_courses_grid` : Mes cours affichés en grille (en tant qu'étudiant)
- `my_favorite_courses_by_role_grid` :Mes cours favoris avec prise en compte des rôles
- `authored_courses` : Mes cours en tant qu'auteur (enseignant editeur)
- `authored_courses_slider` : Mes cours affichés dans un slider ou en grille (enseignant éditeur)
- `managed_courses` : Mes cours en tant que coordinateur (sur capacité local/my:iscoursemanager)
- `managed_courses_slider` : Mes cours affichés dans un slider ou en grille pour rôle d'enseignant éditeur responsable de cours(sur capacité local/my:iscoursemanager)
- `teacher_courses` : Une version combinée d'affichage des cours (en liste repliables et dépliables mémorisée en préférences) en tant qu'enseignant éditeur et non éditeur, qui affiche une icône signalétique différentiante des cours ou j'ai les droits d'édition.
- `teacher_courses_slider` : Affichage des cours dans un slider pour les enseignants éditeurs ou non éditeur
- `teacher_courses_grid`: Affichage des cours en grille pour les enseignants éditeurs ou non éditeur
- `available_courses` : Les cours disponibles en auto-inscription
- `my_remote_courses` : les cours disponibles sur une autre plateforme , via réseau mnet
- `my_templates` : Les modèles de cours qui m'appartiennent (nécessite rôle enseignant dans les modèles concernés)
- `course_areas` : Des “zones de cours” statiques (qui peut pointer 9 catégories de cours) qui

affichent les cours en les excluant de tous les autres widgets d'affichage.
- `course_areas2` : Une seconde “zones de cours” statiques (qui peut pointer 9 catégories de cours) qui affichent les cours en les excluant de tous les autres widgets d'affichage.
- `course_areas_and_availables` : Idem ci-dessus, mais intègre aussi les cours accessibles en auto-inscription.
- `course_categories_grid` et `course_categories2_grid` : Un affichage des catégories désignées pour une mise en valeur sur le tableau de bord (deux possibles, pour différencier catégories pour étudiants et enseignants par exemple)
- `latestnews_full` : Les nouvelles du site, en mode complet (messages développés)
- `latestnews_headers` : Les nouvelles du site, en mode compact (titres)
- `latestnews_simple` : Les nouvelles du site, en affichage simple (non hiérarchique)
- `my_heatmap` : Une table de “température d'activité”
- `my_network` : Mes sites accessibles du réseau (au sens du bloc User Mnet Hosts)
- `my_calendar` : L'expression du bloc Calendar standard
- `course_search` : L'entrée du moteur de recherche de cours
- `left_edition_column` : Dédouble le colonnage et ajoute une colonne éditoriale sur la gauche.

- Widget des zones de cours (course_areas et course_areas2.
- Guide d'utilisation et cas d'usage

Impact du double colonage

Si le double colonage est activé par la présence du `widget left_edition_column`, alors il est possible de répartir les widgets entre la colonne de droite et le colonne de gauche. Ajoutez le suffixe _L pour obliger un widget à s'empiler dans la colonne de gauche.

Contenus statiques et conditionnels

Le widget `static_<n>` a un comportement particulier. Il permet en général de publier des contenus statiques sur le tableau de bord, entre les widgets, éventuellement lié à une condition à remplir. L'utilisation des widgets statiques requiert l'installation du plugin Contenus statiques qui fournira le principe d'édition.

La forme générale du nom de widget est `static_<extension>`.

- **Contenus simples :** si <extension> est un entier (ex: `static_1`, `static_2`, etc), le contenu est unique et est affiché inconditionnellement.
- **Contenus dépendant d'une valeur de profil :** si extension à la forme `profile_field_<fieldshortname>__<modalvalue>`, le contenu est unique et n'est affiché que si la valeur modale attendue est dans le profil de l'utilisateur.

shortname de nom de profil : le local_my accepte désormais un nom de champs de profil avec underscore. Si c'est une valeur qui est à renseigner on mettra double underscore

- **Contenus dépendant du profil :** si l'extension ne mentionne pas de valeur (`profile_field_<fieldshortname>`), alors l'administrateur pourra entrer un contenu

différent par valeur modale disponible.

Valeurs modales

Les valeur modales sont les différentes valeurs disponibles pour un champ de profil. Si le champ est de type “liste” ce sont les valeurs de liste. Si il est du type “texte”, ce sera l'ensemble de tous les textes entrés pour ce champ dans moodle.

Le nom du champ et la valeur modale sont séparés par un DOUBLE “_”. Les noms de champs de profil contenant un seul '_' sont donc admis.

Les valeurs modales sont filtrées comme suit pour obtenir les clefs modales :

- Les espaces et les '_' sont convertis en '-',
- Tous les caractères non strictement alphanumériques et différents du tiret (-) sont supprimés.

Extensibilité/customisation du composant

La fabrication de nouveaux widgets est une opération relativement simple, et consiste en l'écriture d'une fonction de nom préfixé par `local_my_print_<widgetname>(&$excludedcourses, &$courseareacourses)` dans un nouveau fichier `local_modules.php` à la racine du composant. Les modules générique sont codés dans le fichier `modules.php` au même endroit.

La fonction widget reçoit deux paramètres :

- `&$excludedcourses` : un tableau des ids de cours qui devront être exclus des affichages de widget ultérieurs à ce widget. Ce tableau est complété par chaque widget pour indiquer aux widgets suivants les entrées de cours qui ont déjà été présentées à l'utilisateur, et éviter ainsi une multitude de redondances d'affichage. Le code de la fonction du widget devra ajouter à ce tableau les id des cours qu'elle présente dans son contenu.
- `&$courseareacourses` : un tableau des cours qui vont être présentés dans les widgets de type 'Aires de cours', même si les aires de cours sont affichées plus bas dans la page. Ce tableau doit être utilisé en lecture uniquement.

La fonction widget est un pseudo renderer qui doit retourner le HTML contenu dans le widget. Vous pouvez générer directement ce contenu dans la fonction

```
  function local_my_print_mysimplewidget(&$excludedcourses,
&$courseareacourses) {
      // Récupérer une liste de cours par une fonction personnalisée
(associatif par id).
      $courses = local_my_get_custom_courseset();
      // Enlever les cours capturés par les aires de cours.
      if (!empty($courseareacourses)) {
          foreach ($courseareacourses as $cid) {
              if (array_key_exists($cid, $courses)) {
                  unset($courses[$cid]);
              }
          }
      }
```

```
      // Enlever les cours exclus avant.
      if (!empty($excludedcourses)) {
          foreach ($excludedcourses as $cid) {
              if (array_key_exists($cid, $courses)) {
                  unset($courses[$cid]);
              }
          }
      }
      // Ajouter ce qui reste aux exclusion.
      foreach (array_keys($courses) as $cid) {
          $excludedcourses[] = $cid;
      }
      $str = '';
      // Imprimer les listes...
      return $str;
  }
```

Vous pouvez également comme dans certains widgets, utiliser un template qui devra se situer dans le répertoire `templates` du composant :

```
  function local_my_print_templatedwidget(&$excludedcourses,
&$courseareacourses) {
      global $OUTPUT;
      // ... code des exlusions (voir ci-dessus)
      $template = new StdClass();
      $template->data1 = 'value1';
      // ....
      return $OUTPUT->render_from_template('local_my/customtemplate',
$template);
  }
```

Revenir au guide technique - Revenir à l'index du composant - Revenir à l'index des plugins - Retour au catalogue

From:
https://docs.activeprolearn.com/ - **Documentation Moodle ActiveProLearn**

Permanent link:
https://docs.activeprolearn.com/doku.php?id=local:my:modules

Last update: **2026/01/13 08:19**