

École Polytechnique de l'Université de Tours
64, Avenue Jean Portalis
37200 TOURS, FRANCE
Tél. +33 (0)2 47 36 14 14
Fax +33 (0)2 47 36 14 22
www.polytech.univ-tours.fr

Département Informatique
5^{eme} année
2008-2009

Rapport de projet de fin d'études

**Applicatifs métiers basés sur le framework
Typo3 pour la gestion du Laboratoire
d'Informatique**

Étudiant :

Bruno GALLET

bruno.gallet@etu.univ-tours.fr

Encadrants :

Sébastien AUPETIT

sebastien.aupetit@univ-tours.fr

Jean-Charles BILLAUT

jean.billaut@univ-tours.fr

Université François-Rabelais, Tours

Table des matières

Introduction	8
1 Cahier des charges	9
1.1 Préambule	9
1.2 Présentation du projet	9
1.2.1 Contexte	9
1.2.2 Objectifs	9
1.2.3 Descriptif de l'existant	10
1.2.4 Date de livraison	10
1.2.5 Critères d'acceptabilité du produit	10
1.3 Expression des besoins	10
1.3.1 Besoins fonctionnels	10
1.3.2 Besoins non fonctionnels	13
1.4 Contraintes	14
1.4.1 Coûts	14
1.4.2 Délais	14
1.4.3 Contraintes techniques	14
1.4.4 Contraintes organisationnelles	14
1.5 Déroulement du projet	15
1.5.1 Planification	15
1.5.2 Plan d'assurance qualité	15
1.5.3 Responsabilités	16
2 Travail Réalisé	17
2.1 Etude préliminaire	17
2.1.1 Séparation des tables par extension	17
2.1.2 Conséquence de cette séparation	18
2.2 Organisation du travail	18
2.3 Extension de gestion des membres	18
2.4 Extension de gestion des thèses	23
2.5 Extension de gestion des publications	23
2.6 Génération d'un rapport d'activité	24
3 Créer une extension pour Typo3	25
3.1 Création de l'extension	25
3.2 Les formulaires	29
3.2.1 Faire correspondre des champs de la base de données avec les champs prédéfinis par Typo3	30
3.2.2 Personnaliser l'affichage d'un champs	31
3.2.3 Créer ses propres classes pour contrôler les informations contenues dans les champs	33
3.2.4 Ajouter des rubriques d'aide : CSH	34
3.3 Les options du plugin	36

3.4	L'affichage Frontend	37
3.4.1	Utilisation de Template	37
3.4.2	Récupération des options du plugin	40
3.4.3	Interactions avec la base de données	41
	Conclusion	42
3.5	Perspectives	42
A	Bibliographie	43
B	Liens utiles	44
B.1	Créer une extension pour Typo3	44
B.2	TYPO3xdev, site communautaire de développement d'extensions pour Typo3	44
B.3	Ajouter des options sur un plugin	44
C	Modélisation de la base de données	45
D	Installation et Utilisation de l'extension de Gestion de Membres	48
D.1	Installation	48
D.2	Remplissage de la base de données	49
D.2.1	Ajout d'enregistrements	49
D.2.2	Modifications d'enregistrements	52
D.2.3	Formulaires multilingues	53
D.2.4	Cas particulier du formulaire "Membre du Labo"	55
D.3	Insertion et configuration du plugin dans une page	56
D.3.1	Insertion du plugin dans une page	56
D.3.2	Configuration du plugin	58
D.4	Création et utilisation de gabarits	63
D.4.1	Créer son gabarit	64
E	Installation et Utilisation de l'extension de Gestion de Thèses	71
E.1	Installation	71
E.2	Remplissage de la base de données	71
E.2.1	Ajout et modifications d'enregistrements	71
E.3	Insertion et configuration de plugin dans une page	72
E.3.1	Insertion du plugin dans une page	72
E.3.2	Configuration du plugin	72
E.4	Création et utilisation de gabarits	76
E.4.1	Créer son gabarit	76
F	Installation et Utilisation de l'extension de Gestion de Publications	81
F.1	Installation	81
F.2	Remplissage de la base de données	81
F.2.1	Ajout et modifications d'enregistrements	81
F.3	Insertion et configuration de plugin dans une page	82
F.3.1	Insertion du plugin dans une page	82
F.3.2	Configuration du plugin	82
F.4	Création et utilisation de gabarits	87
F.4.1	Créer son gabarit	87

Table des figures

1.1	Planning du projet	16
2.1	Un menu dans le Backend	19
2.2	Affichage dans le Frontend	19
2.3	Exemple de formulaire	20
2.4	L'ajout d'une traduction	21
2.5	Boutons sur un formulaire	22
2.6	Exemple de liste	23
2.7	Options de surbrillance	24
2.8	Exemple de liste avec surbrillance	24
3.1	Admin Tools	25
3.2	Créer une nouvelle extension	25
3.3	Choix du nom de l'extension	26
3.4	General info	26
3.5	Ajout de tables	27
3.6	Ajout de tables 2	27
3.7	Extension de tables	28
3.8	Frontend Plugins 1	28
3.9	Frontend Plugins 2	29
3.10	Création de l'extension	29
3.11	La rubrique d'aide	34
3.12	La rubrique d'aide : description	35
3.13	La rubrique d'aide : detail	35
C.1	Modélisation de la base de données 1	45
C.2	Modélisation de la base de données 2	46
C.3	Modélisation de la base de données 3	47
D.1	Installation : Frontend Plugins	48
D.2	Installation : Création des tables	49
D.3	L'installation réussi	49
D.4	Accéder au menu	50
D.5	Choix du formulaire	50
D.6	Formulaire	52
D.7	Modifications	53
D.8	Multilingue : formulaire	53
D.9	Multilingue : choix de la langue	53
D.10	Multilingue : choix de la traduction d'origine	54
D.11	Multilingue : affichage de la traduction d'origine	54
D.12	Multilingue : remplissage des champs	55
D.13	Champs supplémentaires	55
D.14	Insertion du plugin 1	56

D.15 Insertion du plugin 2	57
D.16 Insertion du plugin 3	57
D.17 Insertion du plugin 4	58
D.18 Onglet Données	58
D.19 Onglet Filtres	59
D.20 Onglet Affichage : Surbrillance	60
D.21 Onglet Affichage : Surbrillance 2	61
D.22 Onglet Affichage : Format des dates	61
D.23 Onglet Affichage	61
D.24 Onglet Séparateur	62
D.25 Onglet Requete	63
D.26 Onglet Gabarit	63
D.27 Onglet gabarit	64
E.1 Choix du formulaire	71
E.2 Onglet Données	72
E.3 Onglet Filtres	73
E.4 Onglet Affichage	74
E.5 Onglet Séparateurs	74
E.6 Onglet Requête	75
E.7 Onglet Gabarit	76
F.1 Choix du formulaire	81
F.2 Onglet Données	82
F.3 Onglet Filtres	83
F.4 Onglet Affichage	84
F.5 Onglet Séparateurs	85
F.6 Onglet Requête	86
F.7 Onglet Gabarit	87

Liste des tableaux

D.1	Tables associées à chaque formulaire	51
D.2	Formulaire associé aux boutons d'ajout et de modification	56
D.3	Marqueurs pour le membre	65
D.4	Marqueurs pour l'équipe	66
D.5	Marqueurs pour la dernière équipe	66
D.6	Marqueurs pour les catégories	67
D.7	Marqueurs pour la dernière catégorie	67
D.8	Marqueurs pour les diplomes	67
D.9	Marqueurs pour le dernier diplôme	68
D.10	Marqueurs pour les postes	68
D.11	Marqueurs pour le dernier poste	68
D.12	Marqueurs pour les fonctions et structures	69
D.13	Marqueurs pour les dernières fonctions et structures	69
E.1	Tables associées à chaque formulaire	72
E.2	Marqueurs pour les thèses/HDR	77
E.3	Marqueurs pour les dirigeants de thèses/HDR	79
E.4	Marqueurs pour les cotutelles des thèses/HDR	80
F.1	Tables associées à chaque formulaire	82
F.2	Marqueurs pour les publications	89
F.3	Marqueurs pour les auteurs de la publication	90
F.4	Marqueurs pour les fichiers de la publication	90
F.5	Marqueurs pour les équipes liées à la publication	91
F.6	Marqueurs pour les thèmes de la publication	91

Introduction

Au cours de la cinquième année au Département Informatique de l'Ecole Polytechnique de l'Université de Tours, chaque étudiant doit réaliser, seul, un Projet de Fin d'étude entre le mois d'octobre et le mois de mai. Deux jours de la semaine, le mercredi et le jeudi sont prévus pour la réalisation du projet. Bien évidemment il est également possible de travailler en dehors de ces horaires.

Le sujet de ce Projet de Fin d'Etude est "Applicatifs métiers basé sur le framework Typo3 pour la gestion du Laboratoire d'Informatique". Ce projet est encadré par Sébastien Aupetit et Jean-Charles Billaut, deux enseignants-chercheurs de Polytech'Tours. L'objectif est de réaliser des extensions pour le site du Laboratoire d'Informatique afin de faciliter ses mises à jour. Ce site a été réalisé entièrement avec le système de gestion de contenu libre Typo3. Le développement de quatre extensions est prévu. Les trois premières doivent permettre la gestion et l'affichage de listes de membres, de thèses/HDR ou de publications suivant différents paramètres définis lors de l'insertion des extensions sur des pages du site. La dernière extension doit générer un rapport d'activité regroupant diverses informations sur les activités du laboratoire.

CHAPITRE 1

Cahier des charges

1.1 Préambule

Dans le cadre de la cinquième année au Département Informatique de l'Ecole d'Ingénieur Polytechnique de l'Université de Tours, nous devons réaliser un Projet de Fin d'Etude (PFE) entre le mois d'octobre et le mois de mai.

Le projet décrit dans ce cahier des charges se nomme : "Applicatifs métiers basé sur le Framework Typo3 pour la gestion du Laboratoire d'Informatique". Ce projet encadré par Sébastien AUPETIT et Jean-Charles BILLAUT, qui feront office de clients, a pour but de réaliser plusieurs extensions Typo3 qui permettant de générer simplement et automatiquement des pages pour le site du Laboratoire d'Informatique de l'Université de Tours (LI) à partir d'une base de données en sélectionnant simplement quelques paramètres.

Typo3 est un CMS (Content Management Systems ou Système de Gestion de Contenu en français) permettant de créer dynamiquement toute sorte de site web. Il s'agit d'un logiciel libre écrit en PHP structuré autour d'une base de données MySQL ou Oracle. Typo3 est sans doute le plus puissant CMS Open Source disponible sur le marché avec notamment le module TemplaVoila qui permet de réaliser rapidement du mapping de page html pour insérer un contenu dynamique, la gestion très simple de droits pour les utilisateurs du site ou la possibilité de créer soi-même des extensions si on ne trouve pas celles que l'on veut dans la bibliothèque d'extensions de Typo3 contenant plus de 2000 références !

Dans Typo3, tous les objets sont organisés de façon hiérarchique. Par exemple, on mettra la première page du site comme racine dans l'arborescence Typo3. Les pages auxquelles on pourra accéder à partir de celle-ci seront positionnées comme fils de la racine, et ainsi de suite. Cette structure permet ainsi une organisation claire de tous les éléments du site.

Typo3 est séparé en deux parties distinctes, le Backend et le Frontend. Le Backend correspond à la partie configuration du CMS alors que le Frontend correspond au site en lui-même.

1.2 Présentation du projet

1.2.1 Contexte

Les extensions réalisées au cours de ce projet sont destinées en premier lieu aux administrateurs du site du Laboratoire Informatique de l'Université de Tours. Ces extensions devront tout de même être suffisamment générales pour être utilisées par n'importe quelle personne puisqu'elles seront mises à la disposition de la communauté Typo3.

1.2.2 Objectifs

Ce projet est composé de deux parties. Dans un premier temps, 3 extensions Typo3 sont à réaliser. La première doit permettre de gérer les membres du LI et ainsi automatiser l'affichage de liste de membres en fonction de certains filtres (équipe, poste, date d'arrivée, . . .). La seconde concerne la gestion des thèses

et la troisième la gestion des publications. La deuxième partie du projet a pour but de générer à partir de Typo3 un rapport récapitulatif des activités du laboratoire.

1.2.3 Descriptif de l'existant

Toutes les extensions seront créées sans fichier de départ. Toutefois, pour faciliter leur développement et divers tests, une copie du site du LI, entièrement réalisé avec Typo3, sur lequel tout est permis est temporairement disponible à l'adresse : <http://li2.fullinfo.net/>.

Il est également à noter que pour faciliter le développement de l'extension de gestion des membres, il est possible de prendre pour exemple les pages présentant les membres de chaque équipe sur le site d'origine du LI puisque l'extension devra pouvoir afficher les mêmes données mais de manière automatique.

Les données utilisées par les extensions seront stockées dans une base de données. La structure de cette base de données a déjà été définie avant le début du projet. Il est néanmoins possible de la modifier s'il le faut.

1.2.4 Date de livraison

La date de livraison des extensions correspond à la date de fin du PFE.

1.2.5 Critères d'acceptabilité du produit

Les extensions seront validées les unes après les autres. Dès qu'une extension sera utilisable, elle subira une phase de tests réalisée par Sébastien Aupetit et Jean-Charles Billaut. S'il n'y a pas de problème, elle sera considérée comme terminée, sinon les bugs seront corrigés ou des fonctions seront ajoutées avant que l'extension soit de nouveau testée.

1.3 Expression des besoins

1.3.1 Besoins fonctionnels

Besoins communs à toutes les extensions

1. Les extensions doivent être placées sur le serveur de TYPO3xdev (site permettant la gestion du développement d'extensions Typo3).
2. Les extensions seront sous licence GPL.
3. Les extensions seront programmées avec les langages PHP, MySQL, TypoScript et JavaScript.
4. Les extensions doivent être utilisables par le plus grand nombre (éviter les fonctions trop spécifiques).
5. Les tables de la base de données doivent être générées à partir des extensions.
6. Les tables utilisées dans les extensions doivent être les plus proches possibles de celles définies par les clients.
7. Des formulaires devront être mis en place pour le remplissage des tables.

- a) Un formulaire correspond à une table.
 - b) Des "liens" entre les formulaires principaux devront être mis en place.
 - c) Certains champs, tels que les champs de type date, doivent être testés.
 - d) Certains champs seront obligatoires.
8. Les extensions devront être utilisables à partir de n'importe quel Template (ou Gabarit en Français). Le Template est une page HTML (+CSS) spécifiant l'affichage dans le Frontend à partir de balises particulières.
- a) 1 ou plusieurs Template de base devront être disponibles pour un affichage "de base".
 - b) Toutes les balises pouvant être utilisées devront être listées et expliquées dans une documentation.
9. Les extensions doivent être multilingues.
- a) Dans ce projet, les extensions seront programmées uniquement pour le français et l'anglais.
 - b) Elles devront avoir le même comportement que l'extension de gestion d'articles tt_news (formulaires et affichages du texte en fonction de la langue de l'utilisateur, affichage dans le Frontend en fonction de la langue choisie).

Pour la gestion des membres

1. Le nom de cette extension est : li_gest_membre_lab.
2. L'extension doit être capable de gérer les membres d'un laboratoire informatique.
3. Le préfixe des tables " appartenant " à cette extension est : tx_ligestmembrelabo.
4. L'extension doit pouvoir afficher une liste de membre en fonction de "filtres".
 - a) Filtrer les membres en fonction de leur Equipe.
 - b) Filtrer les membres par rapport à leur Type de poste.
 - c) Filtrer les membres par rapport à leur Catégorie.
 - d) Filtrer les membres par rapport à leur Structure. Lorsqu'une structure est sélectionnée, il faut sélectionner toutes ces structures "filles".
 - e) Filtrer les membres par rapport à leur Fonction.
 - f) Lorsque cela est possible, ajouter des filtres en fonction des dates disponibles, notamment pour les Catégorie, Structures et Fonctions. 3 choix seront possibles : Actuels (pour sélectionner uniquement les membres qui ont actuellement la "caractéristique" sélectionnée), Tous (pour sélectionner tous les membres qui ont eu ou ont toujours cette "caractéristique") et Anciens (pour sélectionner les membres qui ont eu mais qui n'ont plus cette "caractéristique").
5. Un utilisateur doit pouvoir sélectionner des dossiers contenant les membres à afficher.
 - a) On doit pouvoir sélectionner un ou plusieurs dossiers.
 - b) Lorsque l'on sélectionne un dossier, tous ces sous-dossiers doivent être sélectionnés.
6. Un utilisateur de l'extension doit pouvoir définir sa propre requête pour sélectionner les membres qu'il souhaite afficher.

Pour la gestion des thèses

1. Le nom de cette extension est : `li_gest_theses`.
2. L'extension doit être capable de gérer les thèses/HDR (Habitations à Diriger des Recherches) des membres d'un laboratoire informatique.
3. Le préfixe des tables "appartenant" à cette extension est : `tx.ligesttheses`.
4. L'extension doit pouvoir afficher une liste de thèses/HDR en fonction de "filtres".
 - a) Affichage selon les thèses/HDR " en cours ", " terminés ", " tous ".
 - b) Filtrer selon le membre qui réalise la thèse HDR.
 - c) Affichage selon l'équipe.
 - d) Filtrer selon l'encadrant (directeur ou non).
 - e) Affichage selon le résultat obtenu : " en cours ", " abandon ", " terminé : mention honorable ", " terminé : mention très honorable ", " terminé : mention très honorable avec les félicitations du jury.
 - f) Filtrer selon la structure de rattachement.
5. Un utilisateur doit pouvoir sélectionner des dossiers contenant les thèses à afficher.
 - a) On doit pouvoir sélectionner un ou plusieurs dossiers.
 - b) Lorsque l'on sélectionne un dossier, tous ces sous-dossiers doivent être sélectionnés.
6. Un utilisateur de l'extension doit pouvoir définir sa propre requête pour sélectionner les thèses qu'il souhaite afficher.

Pour la gestion des publications

1. Le nom de cette extension est : `li_gest_publications`.
2. L'extension doit être capable de gérer les publications des membres d'un laboratoire d'informatique.
3. Le préfixe des tables "appartenant" à cette extension est : `tx.ligestpublications`.
4. Une exportation des publications en différents formats doit être prévue.
 - a) Prévoir un export au format bibtex.
 - b) Prévoir un export au format XML. (Les balises doivent encore être définies)
5. Des fichiers pourront être joints à une publication.
 - a) Ajout des fichiers dans la base de données.
 - b) Ajout de règles d'accès aux fichiers.
6. L'extension doit pouvoir afficher une liste de publications en fonction de " filtres ".

- a) Filtrer les publications selon les équipes dans lesquelles elles ont été publiées.
 - b) Afficher selon les membres ayant effectués les publications.
 - c) Filtrer selon les auteurs des publications.
 - d) Filtrer par thèmes.
 - e) Filtrer par type de publication.
 - f) Filtrer selon la date : " en cours ", " toutes ", " depuis ".
 - g) Afficher une publication si elle est internationale ou non.
 - h) Afficher une publication si elle est une publication dans une conférence normale ou en tant qu'invité de la conférence.
 - i) Afficher une publication si c'est un chapitre d'un livre ou non.
 - j) Afficher une publication si elle est une vulgarisation ou non.
 - k) Afficher selon l'année de parution.
 - l) Afficher la publication si elle est parue ou non.
 - m) Définir l'ordre d'affichage des publications : " croissant ", " décroissant ".
7. Un utilisateur doit pouvoir sélectionner des dossiers contenant les publications à afficher.
- a) On doit pouvoir sélectionner un ou plusieurs dossiers.
 - b) Lorsque l'on sélectionne un dossier, tous ces sous-dossiers doivent être sélectionnés.
8. Un utilisateur de l'extension doit pouvoir définir sa propre requête pour sélectionner les publications qu'il souhaite afficher.

Génération d'un rapport d'activités

- 1. Ce rapport d'activités devra être exporté au format RTF et éventuellement au format Excel.
- 2. Il devra contenir les thèses soutenu, les postes occupés ou encore les publications depuis 2004.
Attention : Cette liste n'est pas exhaustive.

1.3.2 Besoins non fonctionnels

- 1. L'affichage des pages créées par les extensions doit être rapide (Utilisation du cache de Typo3).
- 2. N'importe quel utilisateur doit pouvoir utiliser ces extensions Typo3.
- 3. Le programme doit être utilisable avec toutes les versions de Typo3.
- 4. Les éventuels fichiers exportés ne doivent pas être trop volumineux.
- 5. Les extensions doivent être fiables, stables et documentées.
- 6. Les extensions doivent être légères.
- 7. Les extensions doivent être terminées pour la fin des PFE.
- 8. Une documentation et un manuel d'utilisation pour chaque extension devront être réalisés.

1.4 Contraintes

1.4.1 Coûts

Un ordinateur est mis à disposition de l'étudiant pour la réalisation du projet.

Un livre concernant Typo3 est prêté à l'étudiant par Sébastien Aupetit.

1.4.2 Délais

La date de livraison des extensions est la date de fin des PFE.

Un point intermédiaire sera fait au mois de janvier.

1.4.3 Contraintes techniques

L'environnement de travail est principalement Typo3.

Différents langages sont utilisés :

- PHP/MySQL et TypoScript pour la programmation des extensions et la communication avec la base de données
- HTML et CSS pour la création de gabarits.

Les modifications apportées à la base de données sont réalisées et documentées avec Visual Paradigm.

La documentation concernant le code des extensions sera générée à partir de doxygen.

Le serveur svn de TYPO3xdev est utilisé pour gérer les différentes versions des extensions. Le site de TYPO3xdev est disponible à l'adresse <http://forge.typo3.org/projects/show/extensions>.

- La page concernant l'extension de gestion des membres est : http://forge.typo3.org/projects/show/extension-li_gest_membre_labo
- La page concernant la gestion des thèses est : http://forge.typo3.org/projects/show/extension-li_gest_theses
- La page concernant la gestion des publications est : http://forge.typo3.org/projects/show/extension-li_gest_publications
- Les adresses du serveur svn pour ces trois extensions sont respectivement :
 - a) https://svn.typo3.org/TYPO3v4/Extensions/li_gest_membre_labo/
 - b) https://svn.typo3.org/TYPO3v4/Extensions/li_gest_publications/
 - c) https://svn.typo3.org/TYPO3v4/Extensions/li_gest_theses/

1.4.4 Contraintes organisationnelles

Les communications se font soit par email, soit par entretien avec les clients.

Un rapport hebdomadaire est envoyé au client pour résumer l'avancer du projet.

Lorsqu'une partie importante du projet est réalisée, une série de tests est organisée avec les clients pour valider ou non celle-ci.

1.5 Déroulement du projet

1.5.1 Planification

Ce projet est séparé en 5 phases distinctes :

- la première partie concerne la création de l'extension de gestion des membres.
- la seconde est la réalisation de l'extension de gestion des thèses.
- la troisième est l'extension de gestion des publications et des exportations bibliographiques.
- la quatrième, la génération du rapport d'activités.
- et la cinquième partie sera occupée par la rédaction du rapport final et des documentations.

La première partie du projet est la plus longue. Elle sert de "défrichage" pour la création des extensions suivantes. Elle se déroulera du début du mois d'octobre à la fin du mois de décembre.

Ensuite, le mois de janvier sera consacré à l'extension de gestion des thèses.

Le mois de février et le début du mois de mars concerneront l'extension de gestion des publications.

La génération du rapport d'activité sera réalisée pendant la fin de mois de mars et le mois d'avril.

Et enfin, le mois de mai sera employé à la finalisation des documentations, à la rédaction du rapport final et à la préparation de la soutenance.

1.5.2 Plan d'assurance qualité

Lors de la validation des extensions, plusieurs tests seront effectués par l'étudiant ainsi que par les clients :

- Test d'installation : vérification du bon déroulement de l'installation (création ou mise à jour des tables, ajout des formulaires liés à l'extension, ...)
- Test des formulaires : tous les champs des formulaires sont testés dans le cas d'un ajout ou d'une modification. Les formulaires contenant des champs multilingues seront testés en français et en anglais.
- Test de configuration : différentes possibilités de configuration sont testés (sélection de dossiers, test de filtres, création de requêtes, ...) en comparant ce qui est demandé à l'extension et ce qui est effectivement affiché dans le Frontend.

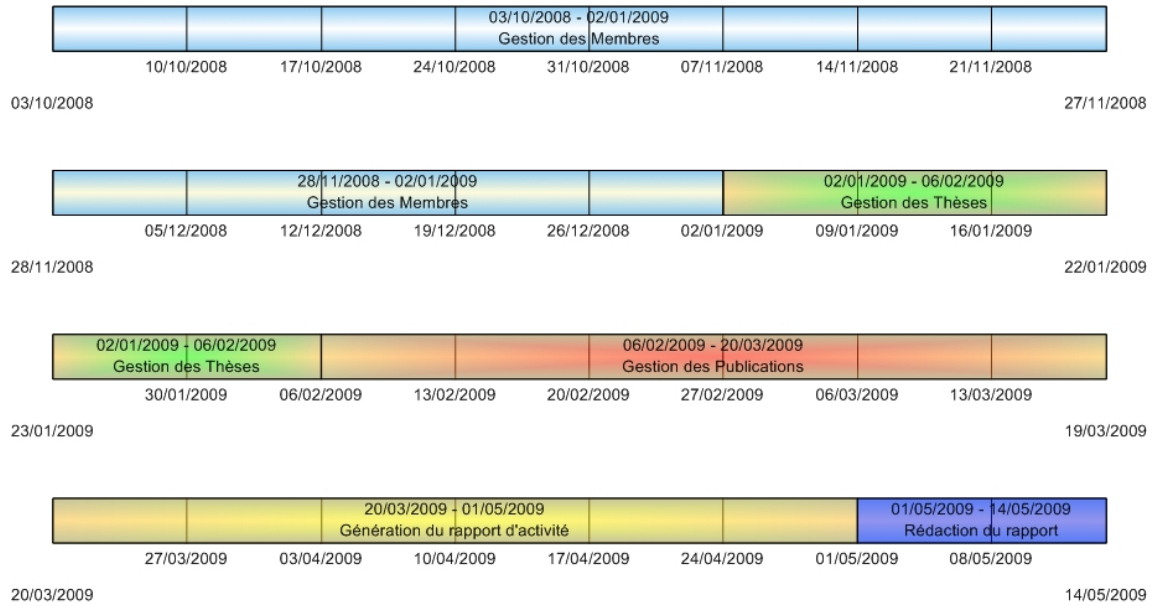


FIG. 1.1 – Planning du projet

- Test d'affichage : différents gabarits sont réalisés pour vérifier le bon fonctionnement des balises disponibles.

1.5.3 Responsabilités

Maîtrise d'ouvrage (MOA)

Sébastien Aupetit et Jean-Charles Billaut sont chargés de définir les objectifs du projet.

Maîtrise d'oeuvre (MOE)

Le projet sera réalisé par un unique étudiant : Bruno Gallet.

CHAPITRE 2

Travail Réalisé

2.1 Etude préliminaire

Avant le début du projet, une base de données reprenant toutes les informations à stocker dans Typo3 a été modélisée (Annexe C). Les extensions à réaliser doivent créer entièrement cette base de données en considérant que chaque table ne doit être créée que par une seule extension.

2.1.1 Séparation des tables par extension

Gestion de membres

L'extension de "Gestion de membres" contiendra les tables :

- MembreDuLabo
- Structure
- Fonction
- Exerce
- TypePosteWeb
- TypePoste
- Possede
- Categorie
- Categorie_MembreDuLabo
- Equipe
- EstMembreDe
- TypeDiplome
- AObtenu

Gestion des thèses

L'extension de "Gestion des thèses" contiendra les tables :

- TheseHDR
- Dirige

Gestion des publications

L'extension de "Gestion des thèses" contiendra les tables :

- Publication
- Theme_Publication
- Theme
- Appartenir
- Fichier
- TypePublication
- Publication_Auteur
- Auteur

2.1.2 Conséquence de cette séparation

Ces trois extensions ont toutes besoins d'avoir accès à la table MembreDuLabo. Celle-ci ne sera créée qu'à partir de l'extension de Gestion de Membres. Par conséquent, il sera nécessaire d'installer l'extension de Gestion de Membres avant les extensions de Gestion de Thèses ou de Gestion de Publications. Dans le cas inverse, il faudra désinstaller les extensions de Gestion de Thèses **ET** de Gestion de Publications avant de désinstaller l'extension de Gestion de Membres.

2.2 Organisation du travail

Ne connaissant pas du tout le développement d'une extension pour Typo3, j'ai tout de même choisi de commencer directement l'extension de Gestion de membres. En effet, ayant déjà travaillé sur Typo3 au cours du Parcours Ingénieurs Polytech, j'ai trouvé préférable de travailler à partir d'exemple simple pour construire la structure de l'extension pour ensuite ajouter des fonctionnalités au fur et à mesure de mes recherches et découvertes.

Durant tout le projet, dès qu'une version d'extension était suffisamment avancée par rapport aux demandes du cahier des charges, une réunion était organisée avec mes deux encadrants Sébastien Aupetit et Jean-Charles Billaut pour vérifier le bon fonctionnement de l'extension ou proposer des améliorations possibles à partir du site de développement.

2.3 Extension de gestion des membres

L'extension de gestion de membres permet l'affichage d'une liste personnalisée de membres d'un Laboratoire d'Informatique sur une page d'un site réalisé avec Typo3 à partir d'une base de données. Pour faciliter l'enregistrement des membres, des formulaires ont été mis en place. L'affichage de la liste sur le site est entièrement paramétrable, il est possible de réaliser toutes sortes de sélection (par équipe, par structure, par fonction,...) et l'affichage en lui-même se définit à partir d'un Template. Il s'agit d'une page HTML composée à base de balises particulières à l'extension permettant de disposer soi-même les informations que l'on souhaite (voir Annexe D pour consulter la documentation utilisateur).

Voici le déroulement de la création de cette extension :

J'ai commencé par développer une petite extension permettant de créer les tables correspondant aux membres en passant par le Kickstarter, un module de création d'extension intégrée à Typo3. Ce module permet de créer toutes les tables et tous les champs que l'on veut. Ce module crée ensuite tous les fichiers de l'extension. Celle-ci, une fois installée sur un site, propose des formulaires pour remplir plus simplement la base de données mais également, lors que l'on insère le plugin sur une page, l'affichage Frontend d'un petit texte prédéfini.

J'ai ensuite rencontré mon premier obstacle, comment transmettre des informations du Backend pour choisir des options d'affichage au Frontend pour effectivement afficher ce que l'on veut. J'ai ainsi découvert l'existence des Flexform. Il s'agit d'une feuille XML permettant de créer son propre menu Backend à base de champs, de boîtes de sélection ou encore de cases à cocher. Il est ensuite possible de récupérer les informations sélectionnées pour personnaliser l'affichage Frontend.



FIG. 2.1 – Un menu dans le Backend

HaNT > Membres 2007-2008 (Test plugin)

Membres 2007-2008 (Test plugin)

Hostname:Ceci est le nom de l'hôte
 Login:Ceci est mon login
 Password:test
 Base:test
 Labo:HaNT
 Liens:0

FIG. 2.2 – Affichage dans le Frontend

Pour tester l'extension au fur et à mesure de son avancée, j'ai créé un fichier sql me permettant de remplir la base de données.

Puis, j'ai travaillé sur les formulaires de saisie de données. Je me suis rapidement aperçu que les formulaires fonctionnaient à partir d'une clé primaire prédéfinie "uid", j'ai donc supprimé les clés et champs que j'avais créé à partir de la modélisation. J'ai ensuite repris le contenu de tous les champs du formulaire pour affiner le contenu des champs. Par exemple, pour une question de facilité d'utilisation j'ai remplacer les dates créés par le Kickstarter en tant qu'entier par des champs du type Date, préremplir des champs avec des valeurs par défaut, afficher dans des boîtes de sélection le contenu d'autres tables,...

FIG. 2.3 – Exemple de formulaire

J'ai ensuite travaillé sur l'affichage Frontend pour récupérer des informations à partir de la base de données (fonction `exec.SELECTquery` de Typo3) en fonction d'informations fournies par le menu d'options du plugin.

Dans le cahier des charges, il est demandé de pouvoir sélectionner les membres à partir du dossier où ils se trouvent. J'ai ainsi implémenté un algorithme permettant de rechercher les membres présents dans le dossier sélectionné et dans ses sous-dossiers. Le principe est très simple : dans Typo3 toutes les pages/dossiers sont stockées dans la table "pages" avec un identifiant "uid" et l'identifiant de la page parente "pid", la fonction recherche donc toutes les pages qui ont pour page parente la page sélectionnée puis on relance récursivement la fonction sur les sous pages trouvées jusqu'à ce que l'on ait la liste de tous les identifiants des pages et sous-pages. Chaque enregistrement de la base de données possède également un champ "pid" qui correspond à l'identifiant de la page dans laquelle il est stocké. Il suffit ensuite de prévoir une clause sur la requête pour ne choisir que les enregistrements présents dans la page sélectionnée.

J'ai ensuite travaillé sur la gestion du multi-langues pour certaines tables où il y a besoin d'une traduction comme par exemple la table gérant les types de postes. Dans Typo3 il existe tout un système pour la gestion du multilangue à partir de trois champs à ajouter dans la table de la base de données : "sys.language_uid" qui contient l'identifiant de la langue (0 pour la langue par défaut), l18n_parent qui donne l'identifiant de la traduction par défaut (0 sinon) et le l18n_diffsource. Il faut ensuite modifier le fichier `tca.php` de l'extension qui sert à définir les formulaires et insérer l'affichage de ses trois champs. Lorsque l'on veut ajouter/modifier un enregistrement on voit alors une boîte de sélection permettant de sélectionner la langue de l'enregistrement puis si l'on a choisi une autre langue que celle par défaut une nouvelle boîte de sélection apparaît pour choisir l'enregistrement par défaut auquel on veut être rattaché. Dans ce cas il est ensuite possible d'afficher à l'écran les valeurs de l'enregistrement par défaut au dessus des champs pour aider au remplissage de la traduction en ajoutant l'option `'l10n_mode' => 'mergelfNotBlank'`

dans la déclaration du champ dans le fichier tca.php. Au niveau du Frontend il est possible de connaître la langue sélectionnée avec la fonction `GLOBALS['TSFE']->sys_language_content`.

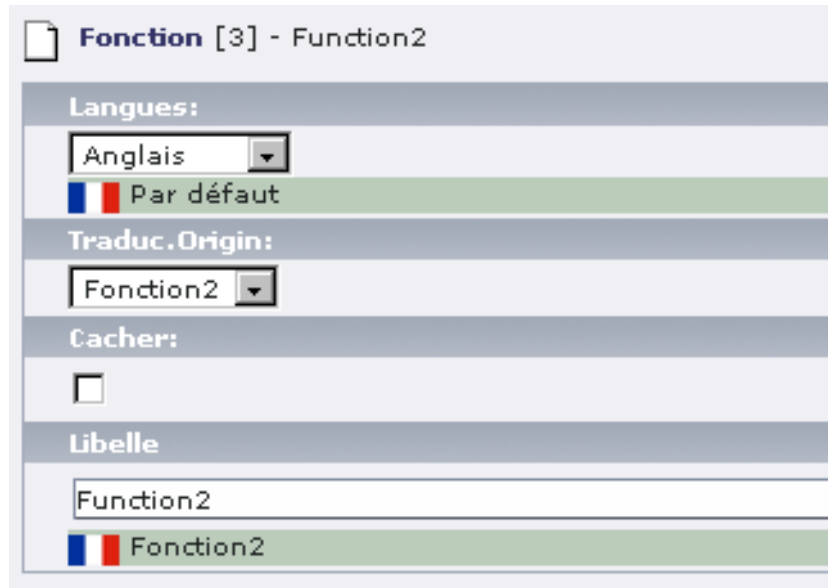


FIG. 2.4 – L'ajout d'une traduction

Pour permettre la personnalisation à l'infini de l'affichage Frontend, j'ai mis en place la gestion de Template sur le même principe que l'extension `tt_news` (extension de gestion de nouvelles très connu dans la communauté Typo3). Un template est une page HTML utilisant des balises ou des marqueurs définis par l'extension. Ses balises et marqueurs sont remplacés lors de la génération de la page.

En testant le fonctionnement de l'extension, j'ai rencontré un problème plutôt mystérieux, en effet j'ai remarqué une erreur lors de l'enregistrement du formulaire `tx_ligestmembrelabo_Categorie_MembreDuLabo`. Après plusieurs heures de recherche et de tests en tout genre je me suis aperçu que le problème venait de la longueur de la table que j'ai donc renommée en `tx_ligestmembrelabo_CategorieMembre`.

Après ce petit contretemps, j'ai travaillé sur la création de classes pour contrôler le contenu de champs dans les formulaires. J'ai ainsi créé une classe permettant de contrôler la validité d'une date dans un champ et une autre pour vérifier la présence d'une date dans un champ.

J'ai ensuite étudié la navigation entre formulaires, afin de faciliter la création d'enregistrements liant deux tables. Par exemple pour lier un membre à une équipe il faut passer par le formulaire `EstMembreDe`. J'ai d'abord recherché les possibilités d'afficher un formulaire dans un autre. Cela s'est avéré impossible, j'ai donc recherché comment ouvrir un formulaire à partir d'un autre en affichant les informations enregistrées dans le formulaire appelant. Pour cela, j'ai ajouté un champ dans la base de données. Ce champ est ensuite utilisé pour afficher les enregistrements liés à la table sous forme de liste. À droite de cette liste, des icônes sont ajoutées à partir de l'option wizard dans le fichier `tca.php` pour ajouter, éditer ou supprimer un enregistrement. L'option wizard permet d'appeler un script externe. Certains scripts existent nativement dans Typo3 comme le script d'ajout ou le script d'édition. Bizarrement, il n'existe aucun script de suppression, j'en ai donc créé un moi-même. Celui-ci récupère l'identifiant, le champ et la table de l'enregistrement à supprimer, on requête est ensuite réalisée pour le supprimer. J'ai ensuite rencontré quelques difficultés avec l'ajout d'un enregistrement, le script d'ajout de Typo3 ne permet pas de pré-remplir des champs ce qui peut être source d'erreur quand on est obligé de saisir plusieurs fois la même chose. Pour corriger cela, j'ai donc

créé un script créant un enregistrement avec les champs pré-remplis. Un appel au script de modification en ensuite fait pour pouvoir modifier l'enregistrement nouvellement créé. Pour réactualiser la liste dans le formulaire appelant, il est impératif de d'enregistrer une nouvelle fois le formulaire.



FIG. 2.5 – Boutons sur un formulaire

Une réunion a ensuite été organisée pour vérifier le fonctionnement de l'extension. Celle-ci a permis de découvrir une faille dans la modélisation de la base de donnée. En effet, à l'origine, il n'était pas prévu qu'un membre ait pu appartenir à plusieurs équipes. Des champs de date de début et de date de fin dans une équipe ont donc été ajoutés. De même, une table permettant la gestion des PEDR (Prime d'Encadrement Doctoral et de Recherche), non prévu dans la base de données initiale a été ajoutée. Cette réunion a également soulevé le problème de l'explication du contenu de certains champs. En effet certains libellés n'étaient pas très explicites, la présence d'une aide donnant quelques explications supplémentaires et parfois quelques exemples m'a été demandée. En prenant exemple sur l'extension tt_news, j'ai ainsi ajouté des csh. Il s'agit de bulles d'aide cliquable ajoutée à gauche de chaque champ. De plus, j'ai du ajouter des "séparateurs" pour l'affichage Frontend, ce sont des balises liées aux balises principales permettant d'afficher un caractère de séparation uniquement si le champ auquel il est lié affiche quelque chose à l'écran pour éviter d'avoir des virgules qui se baladent en fin de ligne.

Après ces modifications, une nouvelle réunion a été programmée, celle-ci n'a donné lieu qu'à des modifications mineures comme l'affichage d'une date Frontend en plusieurs formats ou une gestion des initiales un peu plus pointue (par exemple Jean-Charles doit donner J-C. et non J.).

A la suite de cette réunion, l'extension a été mise en service sur le site du Laboratoire d'Informatique.

Le manuel utilisateur indiquant toutes les fonctionnalités de l'extension est disponible en annexe D.

Voici un exemple de liste qu'il est possible d'obtenir à partir de cette extension.

AUPETIT Sébastien

- ♦ Catégorie1, 2006-06-06, 2030-04-13

GALLET Bruno

- ♦ Catégorie1, 2008-09-01, 2009-09-01
- ♦ Catégorie2, 2004-02-01, 2004-03-01

GAUCHER Pierre

- ♦ Catégorie2, 1837-08-25, ???-??-??

MONMARCHE Nicolas

- ♦ Catégorie2, 1946-12-13, 2007-01-11

SLIMANE Mohamed

- ♦ Catégorie2, 2006-11-03, 2007-01-01

FIG. 2.6 – Exemple de liste

2.4 Extension de gestion des thèses

Cette extension, qui permet d'afficher une liste de thèses, a été commencée pendant la fin du développement de l'extension de gestion des membres. Elle utilise la même structure que l'extension de gestion des membres. Une table permettant de gérer des cotuteurs extérieurs au laboratoire a été ajoutée par rapport à ce que avait été prévu par le cahier des charges.

Le manuel utilisateur indiquant toutes les fonctionnalités de l'extension est disponible en annexe E.

2.5 Extension de gestion des publications

Cette extension, qui permet l'affichage d'une liste de publications, a été commencée pendant la fin du développement de l'extension de gestion des thèses. Elle utilise la même structure que l'extension de gestion des membres. Cette extension a nécessité le développement de deux nouvelles fonctionnalités. La première est la possibilité d'afficher une liste de publication avec le nom de certains auteurs en surbrillance (mise en gras, souligné,...). A partir des options du plugin, il est possible de sélectionner une ou plusieurs équipes, tous les auteurs appartenant à ces équipes seront surlignés sur l'affichage Frontend (cette fonctionnalité a ensuite été ajoutée à l'extension de gestion des membres). La deuxième concerne les fichiers qu'il est possible de lier à une publication. L'extension permet de restreindre l'accès aux fichiers suivant la personne connectée sur le site en utilisant les groupes Frontend.



FIG. 2.7 – Options de surbrillance

AUPETIT Sébastien

BILLAUT Jean-Charles

BONHOMME Patrice

BOUQUARD Jean-Louis

CLAIR Romain

COLAS Sonia

ESSWEIN Carl

GAUCHER Pierre

HUYNH TUONG Nguyen

KERGOSIEN Yannick

LABANE Malika

FIG. 2.8 – Exemple de liste avec surbrillance

Cette extension n'est pas encore totalement terminée, en effet quelques libellés doivent encore être modifiés, les champs du formulaire d'ajout de publication doivent être réorganisés et quelques modifications doivent être apportés à l'affichage sur le site. Ces modifications devraient être terminées au cours des semaines à venir.

Par manque de temps, une fonctionnalité ne sera pas implémentée. Il s'agit de l'exportation de listes de publications aux formats bibtex et xml.

Le manuel utilisateur indiquant toutes les fonctionnalités de l'extension est disponible en annexe F.

2.6 Génération d'un rapport d'activité

Par manque de temps, cette partie du projet de fin d'études n'a pas été étudiée.

CHAPITRE 3

Créer une extension pour Typo3

3.1 Création de l'extension

Typo3 contient un gestionnaire d'extensions qui permet de créer automatiquement tous les fichiers de base d'une extension. Ce gestionnaire est disponible depuis le menu de gauche dans la section "Admin tools".

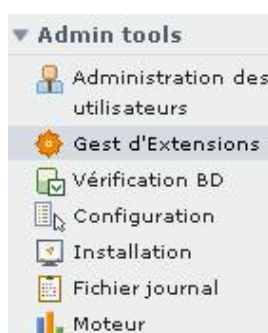


FIG. 3.1 – Admin Tools

On sélectionne "Make new extension" dans le menu déroulant.

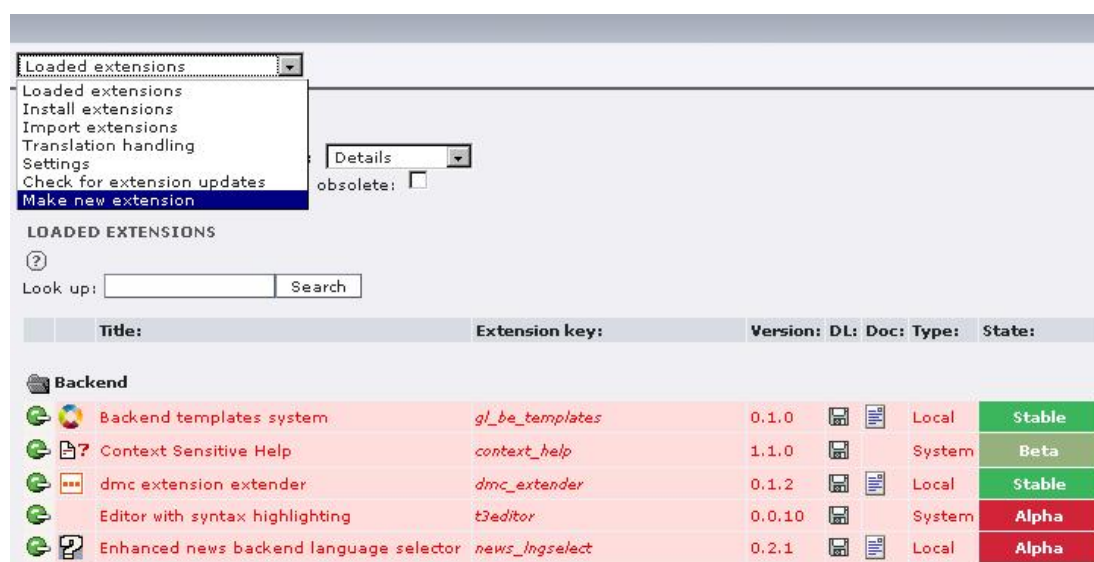


FIG. 3.2 – Créer une nouvelle extension

On arrive maintenant dans le "Kickstarter" qui permet de configurer notre extension.

Pour commencer, il faut choisir un nom pour notre extension. Ici nous choisirons "li_gest_theses".

MVC: Views +

MVC: Actions +

Enter extension key:

Make sure to enter the right extension key from the beginning here! You can register one here.

FIG. 3.3 – Choix du nom de l'extension

On clique sur *Update* pour recharger la page en tenant compte ce nom.

On clique maintenant sur la section "*General info*". Après avoir rempli tous les champs, on clique de nouveau sur *Update*.

KICKSTARTER WIZARD

General info

[Click to Edit] 🗑️

Setup languages +

New Database Tables +

Extend existing Tables +

Frontend Plugins +

Backend Modules +

Integrate in existing Modules +

Clickmenu items +

Services +

Static TypoScript code +

TSconfig +

MVC: Frontend Plugin +

MVC: Controller +

MVC: Models +

MVC: Templates +

MVC: Views +

MVC: Actions +

General info

Enter general information about the extension here: Title, description, category, author...

Title:

Description:

Category:

State

Dependencies (comma list of extkeys):

Author Name:

Author email:

FIG. 3.4 – General info

Dans "*Setup languages*", choisissez les langues pour lesquels vous souhaitez faire une traduction (la langue "*anglais*" est considéré comme la langue par défaut et n'est donc pas disponible dans la liste). Une fois les langues sélectionnée, cliquez sur *Update*.

Le menu "*New Database Tables*" permet de créer les tables utilisées par l'extension.

FIG. 3.5 – Ajout de tablesFIG. 3.6 – Ajout de tables 2

Applicatifs métiers pour Typo3

Extend existing Tables

Add custom fields to existing tables, such as the "pages", "tt_content", "fe_users" or "be_users" table.

Which table:

tx_ligestmembrelabo_MembreDuLabo (Membre du labo) (What is this?)

FIELD: afficher_theses

Field name: afficher_theses (Remove: ☐   )

Field title: Display thesis of member [English]

Afficher les thèses liés au membre [French]

Field type: Integer, 10-1000 ☒ Is Exclude-field (What is this?)

Field title

☒ 1000

FIG. 3.7 – Extension de tables

Le menu *Frontend Plugins* permet de configurer le type de plugin que l'on veut réaliser ainsi que le nom qui sera affiché dans le backend pour sélectionner le plugin. Ici nous conservons l'affichage *"Add to 'Insert Plugin' list in Content Elements"* et on coche la case *"Add icon to 'New Content Element'"* en ajoutant une description.

Frontend Plugins

Create frontend plugins. Plugins are web applications running on the website. A message board, shop, rating feature etc. are examples of plugins.

Enter a title for the plugin:

Managing Thesis/HDR [English]

Gestion de thèses/HDR [French]

☐ USER cObjects are cached. Make it a non-cached USER_INT instead

☐ Enable this option if you want the TypoScript code to be set by default. C included in the template record (recommended is to *not* set this option).

☒ Add to 'Insert Plugin' list in Content Elements'

FIG. 3.8 – Frontend Plugins 1

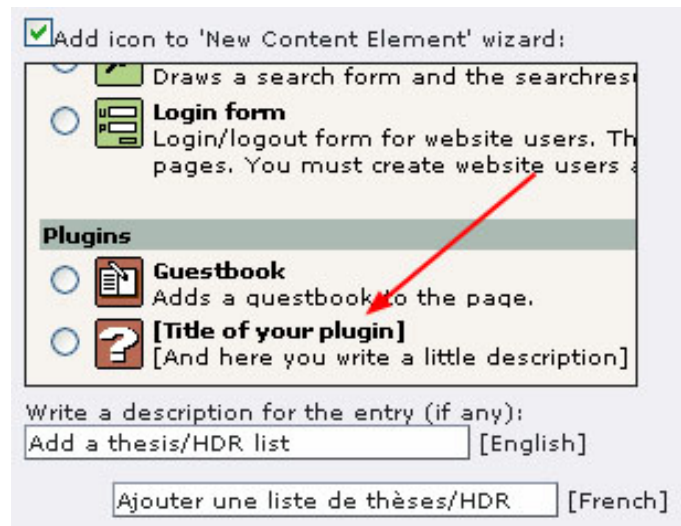


FIG. 3.9 – Frontend Plugins 2

On va maintenant créer les fichiers sur le serveur. On clique sur "View result". La liste des fichiers qui vont être créés est maintenant affichée à l'écran.

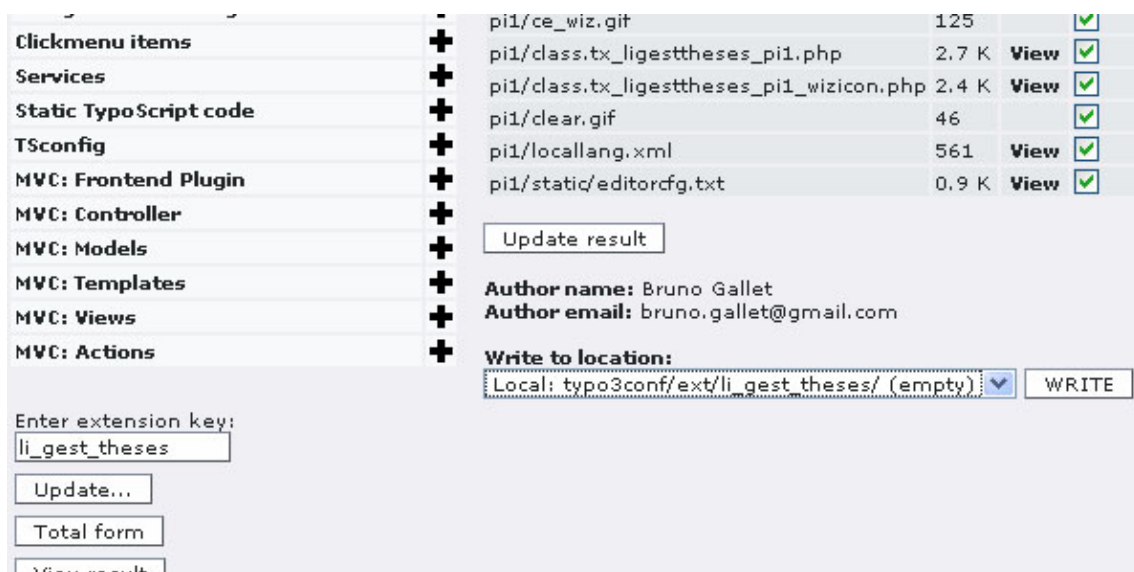


FIG. 3.10 – Création de l'extension

Cliquez sur "WRITE" pour créer les fichiers sur le serveur. Ces fichiers permettent d'avoir la structure principale d'une extension. Ainsi le plugin peut maintenant être installé comme n'importe quel plugin et les formulaires pour remplir la base de données sont fonctionnels.

3.2 Les formulaires

Attention : lorsque l'on modifie soi-même les fichiers d'une extension, il est recommandé de ne plus utiliser le Kickstarter. En effet, toutes les modifications effectuées manuellement ne sont pas prises en compte par le Kickstarter. En cas de mauvaise manipulation (demander l'écriture sur le serveur,...), le

Kickstarter risque d'effacer tout votre travail.

3.2.1 Faire correspondre des champs de la base de données avec les champs prédéfinis par Typo3

Dans Typo3 permet une gestion particulière pour certains champs comme les champs de suppression, de traductions, de date de création de l'enregistrement,... Il peut donc être utile d'indiquer à Typo3 quels champs de votre base de données correspondent aux champs prédéfinis dans Typo3.

En principe, la construction d'une extension avec le Kickstarter suffit pour avoir les bonnes représentations pour les champs, toutefois, il peut y avoir besoin d'en modifier certaines.

Les caractéristiques principales du formulaires sont définies dans le fichier `ext_tables`. Voici un exemple commenté de ce que vous pouvez y rencontrer :

```
// Ces 2 lignes ajoute la possibilité d'insérer un enregistrement
// TypePosteWeb sur une page normale
t3lib_extMgm::allowTableOnStandardPages('tx_ligestmembrelabo_TypePosteWeb');
t3lib_extMgm::addTolInsertRecords('tx_ligestmembrelabo_TypePosteWeb');

$TCA["tx_ligestmembrelabo_TypePosteWeb"] = array (
    "ctrl" => array (
        // Titre du formulaire. Ici le titre est chargé à partir
        // du fichier locallang_db.xml
        'title' => 'LLL:EXT:li_gest_membre_labo/
locallang_db.xml:tx_ligestmembrelabo_TypePosteWeb',
        // Champs principaux afficher dans le Backend
        //
        'label' => 'LibelleWeb',
        // Champs secondaire afficher dans le Backend en cas
        // d'absence des champs principaux
        'label_alt' => '',
        // La mise à 1 de label_alt_force signifie que l'on
        // force l'affichage des champs secondaires
        'label_alt_force' => '1',
        // On fait correspondre le champ tstamp de notre base
        // de données au champs tstamp de Typo3
        'tstamp' => 'tstamp',
        // idem que pour tstamp
        'crdate' => 'crdate',
        // idem que pour tstamp
        'cruser_id' => 'cruser_id',

        // Permet de spécifier l'ordre de tri dans le Backend
        'default_sortby' => "ORDER_BY_sys_language_uid, _LibelleWeb",

        // Correspondance des champs de traduction de notre base
        // de données avec ceux de Typo3
        'copyAfterDuplFields' => 'sys_language_uid',
        'useColumnsForDefaultValues' => 'sys_language_uid',
```

```

    'transOrigPointerField' => 'l18n_parent',
    'transOrigDiffSourceField' => 'l18n_diffsource',
    'languageField' => 'sys_language_uid',
    'shadowColumnsForNewPlaceholders' => 'sys_language_uid',
l18n_parent',

    // On fait correspondre le champ deleted de notre base
    // de données au champs delete de Typo3
    'delete' => 'deleted',

    // On indique le fichier contenant la description des champs
    // (tca.php). t3lib_extMgm::extPath($_EXTKEY) donne le chemin
    // vers le dossier principal de l'extension
    'dynamicConfigFile' => t3lib_extMgm::extPath($_EXTKEY).
'tca.php',

    // On indique l'icone choisi pour cette table
    'iconfile' => t3lib_extMgm::extRelPath($_EXTKEY).
'icon_tx_ligestmembrelabo_TypePosteWeb.gif',
    ),
);

```

Pour plus de précision veuillez vous référer à la documentation Typo3 http://typo3.org/documentation/document-library/core-documentation/doc_core_api/4.0.0/view/4/2/

3.2.2 Personnaliser l'affichage d'un champs

Le Kickstarter permet de définir les principales particularités d'un champ (type, longueur,...). Si vous souhaitez affiner ses paramètres, rendez-vous dans le fichier tca.php qui contient la définition de tous les champs des formulaires. Voici un exemple commenté de ce que vous pouvez y trouver :

```

// Déclaration du formulaire pour la table tx_ligestmembrelabo_MembreDuLabo
$TCA["tx_ligestmembrelabo_MembreDuLabo"] = array (
    "ctrl" => $TCA["tx_ligestmembrelabo_MembreDuLabo"]["ctrl"],
    // Liste des champs
    "interface" => array (
        "showRecordFieldList" => "Nom, _Genre, _TypePublication"
    ),
    // Configuration des champs
    "columns" => array (
        // Configuration du champ Nom de la base de données
        "Nom" => Array (
            // Le champ est toujours disponible
            "exclude" => 1,
            // Texte inscrit au dessus du champ
            "label" => "LLL:EXT:li_gest_membre_labore/locallang_db
.xml:tx_ligestmembrelabo_MembreDuLabo.nom",
            // Configuration du champ
            "config" => Array (
                // Le type de champ est un champ de texte
                "type" => "input",
                // Le nombre de caractères affichés à l'écran

```



```

        // est 48
        "size" => "48",
        // Le champ contient au maximum 255 caractères
        "max" => "255",
        // required: le champ est obligatoire
        // trim: les espaces autour du texte sont
        // effacés
        "eval" => "required,trim",
    )
),
// Configuration du champ Genre de la base de données
"Genre" => Array (
    "exclude" => 1,
    "label" => "LLL:EXT:li_gest_membre_labo/locallang_db
.xml:tx_ligestmembrelabo_MembreDuLabo.genre",
    "config" => Array (
        // Champ du type boîte de sélection
        "type" => "select",
        // 1 seul choix est affiché en même temps
        "size" => 1,
        // L'utilisateur ne peut sélectionner qu'un
        // seul item
        "maxitems" => 1,
        // on définit les choix possibles
        "items" => Array (
            // Le tableau ci dessous contient, le texte affiché
            // à l'écran et le texte mis dans la base de données
            Array("LLL:EXT:li_gest_membre_labo/locallang_db.xml:
tx_ligestmembrelabo_MembreDuLabo.genre.h", "H"),
            Array("LLL:EXT:li_gest_membre_labo/locallang_db.xml:
tx_ligestmembrelabo_MembreDuLabo.genre.f", "F"),
        ),
        // Par défaut l'item "H" sera sélectionné
        "default" => "H",
    )
),
// Configuration du champ TypePublication de la base de données
"TypePublication" => Array (
    "exclude" => 1,
    "label" => "LLL:EXT:li_gest_publications/locallang_db
.xml:tx_ligestpublications_Publication.typepublication",
    "config" => Array (
        // Champ du type boîte de sélection
        "type" => "select",
        // Les choix possibles sont définis
        // à partir de la base de données
        "foreign_table" =>
"tx_ligestpublications_TypePublication",
        "foreign_table_where" => "ORDER_BY
_tx_ligestpublications_TypePublication.Libelle",
    )
),

```



```

        "size" => 1,
        "minitems" => 0,
        "maxitems" => 1,
    )
),
),
// On définit l'ordre d'affichage des champs
"types" => array (
    "0" => array("showitem" => "Nom, _Genre, _TypePublication")
)
);

```

De nombreux types de champ sont disponibles dans Typo3, on peut ainsi afficher des boîtes de texte plus ou moins complexe (avec les options de mises en page,...), des checkbox, des boîtes de sélection,... Pour plus de précision veuillez vous référer à la documentation Typo3 http://typo3.org/documentation/document-library/core-documentation/doc_core_api/4.0.0/view/4/2/

3.2.3 Créer ses propres classes pour contrôler les informations contenues dans les champs

Avec Typo3, il est possible de rajouter des classes pour contrôler le contenu des champs dans les formulaires. Le contrôle peut se faire avant l'affichage du formulaire, pendant la saisie du texte ou lors de la validation du formulaire.

Pour commencer, il faut créer une classe php :

```

<?php

class nomDeLaClasse {

}

if (defined('TYPO3_MODE') &&
$TYPO3_CONF_VARS[TYPO3_MODE]['XCLASS']['ext/nomExtension/nomFichier'])
{
    include_once($TYPO3_CONF_VARS[TYPO3_MODE]['XCLASS']
['ext/nomExtension/nomFichier']);
}

?>

```

Ensuite en fonction des besoins, on peut ajouter les fonctions suivantes (les noms des fonctions doivent être conservés) :

- `deevaluateFieldValue($params)` pour tester le champ avant l'affichage du formulaire.

```

function deevaluateFieldValue($params) {
    $valeur = $params['value'];

    //Traitement de la variable $valeur ...

    return $valeur;
}

```

```
}
```

- `returnFieldJS()` pour tester le champ lors de la saisie. Attention, le contenu du texte retourné par cette fonction est en Javascript !

```
function returnFieldJS() {
    return "var _modif_ = _value;

    .....// Traitement de la variable _modif

    .....return _modif;";
}
```

- `evaluateFieldValue($value, $is_in, &$set)` pour tester le champ après validation du formulaire.

```
function evaluateFieldValue($value, $is_in, &$set) {
    $modif = $value;

    // Traitement de la variable $modif

    return $modif;
}
```

Il faut maintenant ajouter dans le fichier `ext_tables.php` un lien vers la classe en modifiant la variable `$TYPO3_CONF_VARS` avec `$TYPO3_CONF_VARS['SC_OPTIONS']['tce']['formevals']['nomDeLaClasse'] = 'EXT :nomExtension/nomFichier'` ;

Pour terminer, il faut préciser dans la configuration du(des champ(s) pour lequel(s), la classe à utiliser.

```
"nomChamp" => Array (
    "exclude" => 1,
    "label" => "libelle",
    "config" => Array (
        "type" => "input",
        "size" => "10",
        "max" => "10",
        "eval" => "trim,nomDeLaClasse",
    )
),
```

3.2.4 Ajouter des rubriques d'aide : CSH

Parfois, le libellé d'un champ n'est pas suffisant pour aider l'utilisateur à le remplir. Il est ainsi possible d'associer une aide pour chaque champ. Cette aide est disponible à partir d'un point d'interrogation située à gauche du libellé du champ.



FIG. 3.11 – La rubrique d'aide

Il existe deux niveaux d'aide, le niveau "description" qui lorsque l'on passe sur le point d'interrogation affiche un petit texte à l'écran et le niveau "detail" qui permet d'ouvrir une fenêtre lorsque l'on clic sur le point d'interrogation.

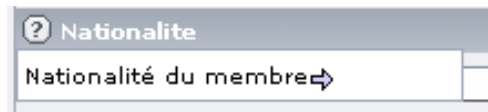


FIG. 3.12 – La rubrique d'aide : description

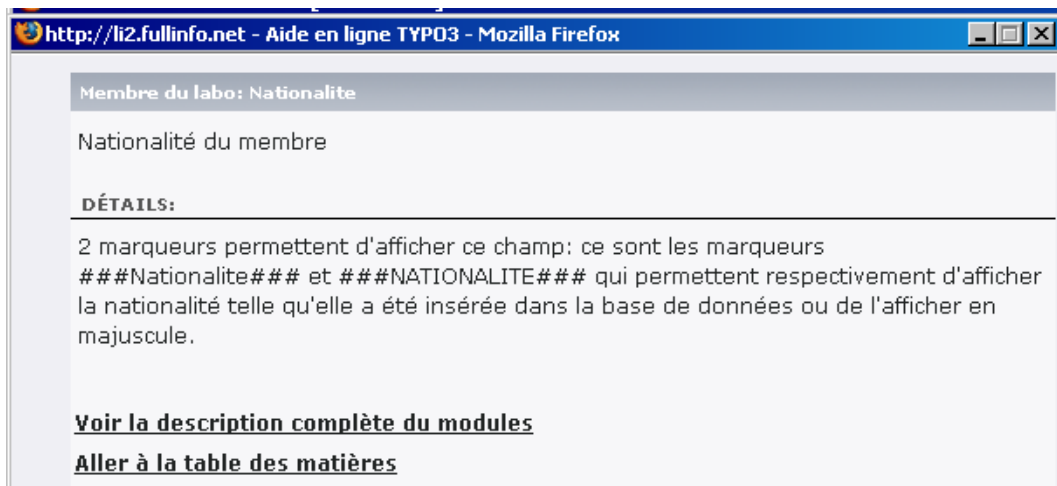


FIG. 3.13 – La rubrique d'aide : detail

Pour ajouter ce champ d'aide, il faut créer un fichier xml du type suivant (un fichier par table) :

```
<?xml version="1.0" encoding="utf-8" standalone="yes" ?>
<T3locallang>
  <meta type="array">
    <description>
      CSH for pour la table tx_ligestpublications_Theme
    </description>
    <type>CSH</type>
    <csh_table>tx_ligestpublications_Theme</csh_table>
    <fileId>EXT:li_gest_publications/csh/ligestpublications
    _locallang_csh_tx_ligestpublications_Theme.xml</fileId>
  </meta>
  <data type="array">
    <!-- Aide dans la langue par défaut (généralement en anglais) -->
    <languageKey index="default" type="array">
      <label index=".description">
        Description générale de la table
      </label>
    <!-- Ici, on doit remplacer Libelle par le nom du champ -->
    <!-- pour lequel on veut ajouter une aide générale -->
    <label index="Libelle.description">
      Description générale du champ Libelle.
```

```

        </label>
<!-- Ici , on doit remplacer Libelle par le nom du champ -->
<!-- pour lequel on veut ajouter une aide détaillée -->
        <label index=" Libelle.details">
                Description détaillé du champ Libelle .
        </label>
        <label index=" Libelle2.description">
                Description générale du champ Libelle2 .
        </label>
        <label index=" Libelle2.details">
                Description détaillé du champ Libelle2 .
        </label>
</languageKey>
<!-- Aide pour le français -->
<languageKey index=" fr" type=" array">
        <label index=" .description">
                Description générale de la table
        </label>
        <label index=" Libelle.description">
                Description générale du champ Libelle .
        </label>
        <label index=" Libelle.details">
                Description détaillé du champ Libelle .
        </label>
        <label index=" Libelle2.description">
                Description générale du champ Libelle2 .
        </label>
        <label index=" Libelle2.details">
                Description détaillé du champ Libelle2 .
        </label>
</languageKey>
</data>
</T3locallang>

```

Pour que ce fichier xml soit pris en compte par votre extension, vous devez ajouter un lien vers celui-ci dans le fichier ext_tables.php avec la fonction t3lib_extMgm : :addLLrefForTCAdescr('NomDeLaTable','LienVersTable') ;. Par exemple :

```

t3lib_extMgm :: addLLrefForTCAdescr( ' tx_ligestpublications_Theme ' ,
'EXT:li_gest_publications/csh/ligestpublications_locallang_csh_
tx_ligestpublications_Theme.xml' );

```

3.3 Les options du plugin

L'ajout d'options pour le plugin (lors de l'insertion sur une page) se fait à l'aide de flexform. Il s'agit d'une feuille xml permettant un formulaire sur le même principe que ceux du fichier tca.php proposant les différents choix possibles.

Vous pouvez retrouver un tutoriel complet sur la page http://wiki.typo3.org/index.php/Extension_Development,_using_Flexforms.

3.4 L'affichage Frontend

Dès la création d'une extension Frontend, il est possible d'afficher une page ayant un contenu prédéfini. Bien évidemment, il est possible de personnaliser cet affichage à l'aide de fonctions PHP ou Typoscript pour récupérer des informations de la base de données, obtenir des renseignements sur l'utilisateur connecté à l'intranet,... Typo3 permet également la mise en place de Template ou Gabarit, cet outil très puissant permet la personnalisation de la page à afficher.

La page php générant l'affichage Frontend est située dans le dossier "pi1" sous le nom "class.nomExtension_pi1.php".

3.4.1 Utilisation de Template

Un template est une page HTML comportant des balises définies dans la page php de l'extension qui sont remplacées lors de la génération de la page.

Un gabarit est créé sur la base d'une page HTML (en-tête, corps,...). Dans le corps de cette page (body), on peut ensuite rajouter différentes balises pour obtenir un affichage personnalisé. Engénérale, on utilise les balises TEMPLATE (début du gabarit), CONTENT (début d'un contenu) et ITEM (liste des informations à afficher) comme structure de base.

Les balises se notes toutes de la même façon :

- une balise de départ : `<!-- ###NOM_Balise### begin-->`
- une balise de fin : `<!-- ###NOM_Balise### end-->`

Il existe deux types de balise. En premier lieu, celles qui permettent de séparer le code en plusieurs parties, elles permettent d'afficher soit une autre partie, soit le texte contenu entre sa balise de départ et sa balise de fin, si nécessaire, ce texte peut être répété plusieurs fois. En second lieu des balises permettant de créer des liens. Dans ce deuxième cas, la balise de départ est ainsi remplacer par un texte et la balise de fin par un autre. Pour l'affichage simple d'un texte, on utilisera plus simplement les marqueurs notés "###NOM_Marqueur###". Ce marqueur est remplacé par un texte.

Nous allons maintenant developper un exemple d'utilisation de template dans une extension.

Voici le template que nous allons utiliser :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
</head>

<body>

<!-- ###TEMPLATE### begin-->

<!-- ###CONTENT### begin -->

<!-- ###ITEM### begin -->

<p>
```

```

<!--###Balise1Lien### begin -->
      ###Marqueur1### ###Marqueur2###
<!--###Balise1Lien### end -->
<!--###BALISE_PARTIE### begin -->
      <br />###Marqueur3###
<!--###BALISE_PARTIE### end -->
</p>

<!-- ###ITEM### end -->

<!-- ###CONTENT### end -->

<!-- ###TEMPLATE### end-->

</body>
</html>

```

Dans cet exemple, nous pouvons remarquer la présence des 3 balises de base (TEMPLATE, CONTENT et ITEM). Tout le texte contenu entre la balise de départ et la balise de fin d'ITEM sera répété plusieurs fois comme pour l'affichage d'une liste. La balise "Balise1Lien" permettra d'afficher un lien externe. Les marqueurs "Marqueur1" et "Marqueur2" seront remplacés par du texte. La balise "BALISE_PARTIE" permettra d'afficher une deuxième liste avec le "Marqueur3". On essaiera ainsi d'obtenir l'affichage du code HTML suivant :

```

<p>
  <a href="www.toto.com">Toto Root</a><br/>
  Info1<br/>
  Info2
</p>

<p>
  <a href="www.toto.com">Toto Root</a><br/>
  Info1<br/>
  Info2
</p>

<p>
  <a href="www.toto.com">Toto Root</a><br/>
  Info1<br/>
  Info2
</p>

```

Nous allons maintenant voir le code permettant d'utiliser le template et définir les balises pour afficher cette liste. Ce code est à insérer dans la fonction main du fichier "class.nomDeLExtension_pi1.php".

Pour commencer il faut lire le fichier html en indiquant le lien vers le fichier avec la fonction "cObj->fileResource" :

```
$this->templateCode = $this->cObj->fileResource("lien_template");
```

On crée ensuite un tableau qui contiendra le contenu à mettre à la place des marqueurs et balises :

```
$template = array();
```

On remplit maintenant le tableau pour les balises principales en indiquant les parties qu'elles vont compléter :

```
$template['total'] = $this->cObj->getSubpart($this->templateCode, '###TEMPLATE###');
$template['item'] = $this->cObj->getSubpart($template['total'], '###ITEM###');
$template['balisePartie'] = $this->cObj->getSubpart($template['item'], '###BALISE_PARTIE###');
```

On initialise le tableau contenant les marqueurs contenu dans la partie ITEM :

```
$markerArray = array();
```

On crée maintenant une boucle (dans notre exemple elle sera parcourue 3 fois) pour remplir la partie ITEM :

```
$i=0
while(i<3){
...
i++;
}
```

A l'intérieur de cette boucle, on remplit les marqueurs avec le code :

```
$markerArray['###Marqueur1###'] = "Toto";
$markerArray['###Marqueur2###'] = "Root";
```

Pour créer un lien vers une page, on utilise le code suivant :

```
$this->local_cObj = tslib_div::makeInstance('tslib_cObj');
$this->local_cObj->setCurrentVal($GLOBALS['TSFE']->id);
$this->typolink_conf = $this->conf['typolink.'];
$temp_conf = $this->typolink_conf;
$temp_conf['parameter'] = "www.toto.com";
$temp_conf['extTarget'] = '';
$temp_conf['parameter.']['wrap'] = "|";
$wrappedSubpartContentArray['###Balise1Lien###'] =
    $this->local_cObj->typolinkWrap($temp_conf);
```

On va maintenant lier les balises et marqueurs à la partie ITEM :

A la fin de la boucle on insère le code :

```
$contentItem .= $this->cObj->substituteMarkerArrayCached(
    $template['item'], $markerArray, Array(), $wrappedSubpartContentArray);
```

Après la boucle on doit maintenant lier la partie ITEM à la partie CONTENT puis la partie CONTENT à la partie TEMPLATE. Enfin, on retourne la page affichée. On insère ainsi les lignes suivantes à la fin de

la fonction main :

```
$subpartArray['###CONTENT###'] = $contentItem;
$content = $this->cObj->substituteMarkerArrayCached($template['total'],
    array(), $subpartArray);
return $this->pi_wrapInBaseClass($content);
```

Pour ajouter la seconde liste on crée une nouvelle boucle dans la boucle principale après avoir initialisé le tableau contenant les marqueurs de cette partie :

```
$subpartArray_Item = Array();"
$markerArrayBalisePartie = array();
$j=1
while(j<3){
    $markerArrayBalisePartie['###Marqueur3###'] = "Info".j;
    $contentBalisePartie .=
        $this->cObj->substituteMarkerArrayCached($template['balisePartie'],
            $markerArrayBalisePartie, array(), array());
    j++;
}
$subpartArray_Item['###BALISE_PARTIE###'] = $contentEquipe;
```

On remplace ensuite la ligne :

```
$contentItem .= $this->cObj->substituteMarkerArrayCached(
    $template['item'],$markerArray, Array(), $wrappedSubpartContentArray);
```

Par :

```
$contentItem .= $this->cObj->substituteMarkerArrayCached(
    $template['item'],$markerArray, $subpartArray_Item, $wrappedSubpartContentArray);
```

3.4.2 Récupération des options du plugin

Comme nous l'avons vu dans une partie précédentes, il est possible d'ajouter des options pour plugin. Bien évidemment, il est possible de récupérer les informations sélectionnée lors de la configuration du plugin. Le code suivant permet de stocker les options choisies dans un tableau :

```
$this->pi_initPIflexForm();
$this->lConf = array();
$piFlexForm = $this->cObj->data['pi_flexform'];
foreach ($piFlexForm['data'] as $sheet => $data)
    foreach ($data as $lang => $value)
        foreach ($value as $key => $val)
            $this->lConf[$key] = $this->pi_getFFvalue($piFlexForm, $key, $sheet);
```

On peut ensuite obtenir les informations du tableau en utilisant `$this->lConf['nomDeLOption']`). `nomDeLOption` correspond à nom donné à l'option dans le fichier flexform.

3.4.3 Interactions avec la base de données

Des fonction Typo3 permettent de communiquer avec la base de données utilisée par Typo3 sans avoir à donner le nom de la base ou le mot de passe. Tout est géré directement par le CMS. Il s'agit des fonctions :

- `$GLOBALS['TYPO3_DB']->exec_SELECTquery($select_fields, $from_table, $where_clause, $groupBy, $orderBy, $limit)` : pour réaliser des requêtes du type SELECT
- `$GLOBALS['TYPO3_DB']->exec_INSERTquery($table, $fields_values)` : pour réaliser des requêtes du type INSERT
- `$GLOBALS['TYPO3_DB']->exec_UPDATEquery($table, $where, $fields_values)` : pour réaliser des requêtes du type UPDATE
- `$GLOBALS['TYPO3_DB']->exec_DELETEquery($table, $where)` : pour réaliser des requêtes du type DELETE

Pour ces fonctions, les paramètres `$select_fields`, `$from_table`, `$where_clause`, `$groupBy`, `$orderBy`, `$limit`, `$table` ou `$where` sont des chaînes de caractère dans lesquels on indique la partie de la requête qui correspond au nom de la variable. Par exemple, pour le champ `$select_fields`, on indique la partie SELECT de notre requête sans mettre le mot clé SELECT (par exemple : "nomDeLaTable.*"). Pour la variable `$where_clause` on indique par exemple : "nomDeLaTable.Champ1='Toto' AND nomDeLaTable.Champ2='Root'". Pour la variable `$fields_values`, il faut créer un tableau contenant les nom des champs à créer ou éditer ainsi que les valeurs à donner à ces champs. On peut par exemple écrire :

```
$fields_values = array(
    'Champ1' => 'Toto',
    'Champ2' => 'Root'
);
```

On peut ensuite utiliser ses requêtes pour générer notre page. Les requêtes INSERT, UPDATE et DELETE retourne un booléen, TRUE si la requête s'est bien passé, FALSE sinon. Pour une requête SELECT il est possible de parcourir les reponses renvoyées de la manière suivante :

```
$select_fields = '*';
$from_table = 'nomDeLaTable';
$where_clause = '';
$groupBy = '';
$orderBy = 'Champ1';
$limit = '';

$res = $GLOBALS['TYPO3_DB']->exec_SELECTquery($select_fields,
    $from_table, $where_clause, $groupBy, $orderBy, $limit);

// On parcourt les résultats
while($row = $GLOBALS['TYPO3_DB']->sql_fetch_assoc($res))
{
    // On peut accéder à un champ avec $row['nomDuChamp']
    $test = $test.$row['champ1'].'_';
}
```

Conclusion

Bien que ce sujet ne fasse pas partie de mes premiers choix, je suis finalement très content de l'avoir choisi. Au cours de mes années au Parcours Ingénieurs Polytech, j'avais eu l'occasion de travailler sur un mini-projet de construction de site web avec Typo3, cette expérience m'a sans doute beaucoup aidée dans mon approche de ce projet. J'ai ainsi pu éviter la partie "appropriation" du framework Typo3 qui m'avait pris de nombreuses heures lors du mini-projet.

Ce temps de gagné sur le début du projet de m'a malheureusement pas permis de le mener à son terme. En effet, il reste encore quelques corrections à apporter à l'extension de gestion des publications et l'extension de génération de rapport d'activité n'a pas été commencée. Par rapport à mon planning initiale, j'ai passé plus de temps que prévu sur l'extension de gestion des membres. Il s'agissait de ma première extension Typo3 et la découverte des fonctionnalités ne s'est pas toujours bien passée. A chaque avancée de l'extension, de nouvelles idées apparaissaient et j'ai ainsi réalisé de nouvelles recherches sur la possibilité de les réaliser. Au final les deux premières extensions, gestion de membres et gestion de thèses/HDR ont été terminées. Il est ainsi possible lorsqu'elles sont installées de générer automatiquement sur des pages du site réalisé avec Typo3 des listes de membres ou des listes de thèses/HDR à partir des informations stockées dans la base de données du site. L'installation de ses extensions crée les tables nécessaires à leur utilisation dans la base de données et ajoute des formulaires pour remplir ses tables. Il est ensuite possible d'intégrer les plugins sur une page du site et de configurer les listes en sélectionnant les caractéristiques des membres/thèses/HDR que l'on veut afficher. L'affichage Frontend est personnalisable à l'aide de gabarits qui gèrent la disposition des éléments sur la page.

Actuellement, seule l'extension de gestion de membres est réellement installée sur le site du Laboratoire d'Informatique de l'Ecole. Pour faciliter la mise en service de cette extension, une demi-journée de formation sera organisée avec Jean-Charles Billaut et Christelle Grange le 10 juin 2009.

Je tiens enfin à remercier mes deux encadrants Sébastien Aupetit et Jean-Charles Billaut pour leurs conseils, leurs encouragements et leur disponibilité tout au long du projet.

3.5 Perspectives

Quelques améliorations peuvent être apportées aux extensions réalisées. Par exemple pour l'extension de gestion des publications, pour le moment les fichiers à associer à une publication doivent être chargés il doit être possible de charger en dehors du formulaire à partir de l'espace "média". Il est sans doute possible de charger les fichiers directement dans le formulaire soit par l'intermédiaire d'une fonction en TypoScript s'il en existe une, soit en passant par un script php. Il est également possible d'améliorer la cohérence des informations données à la base de données en créant de nouvelles classes de contrôle pour les formulaires.

De plus, la partie exportation d'une liste de publications aux formats bibtex et xml de l'extension de gestion des publications n'a pas été réalisée. Pour plus de facilité, il est préférable de réaliser cette fonctionnalité en créant une extension à part entière.

ANNEXE A

Bibliographie

Voici une liste d'ouvrage très intéressant sur Typo3 :

- Werner Altmann, René Fritz et Daniel Hinderink, *Typo3*. Eyrolles. 2006. 533 p. (Open Source Press).
- Daniel Kock, *Mastering TypoScript - TYPO3 Website, Template, and Extension Development*. Packt publishing. 2006. 404 p.
- René Fritz, Daniel Hinderink et Werner Altmann, *Typo3 : Enterprise Content Management*. Packt publishing. 2005. 624 p.
- Dmitry Dulepov, *Typo3 : Extension Development*. Packt publishing. 2008. 232 p.

ANNEXE B

Liens utiles

B.1 Créer une extension pour Typo3

- <http://www.typo3journal.info/articles/developpement.html>
- <http://hachmeister.org/en/typo/tutorials/development/create-your-own-frontend-plugin/>

B.2 TYPO3xdev, site communautaire de développement d'extensions pour Typo3

- Présentation : <http://typo3.org/development/collaborative-extension-development/>
- Site : <http://forge.typo3.org/projects/show/extensions>

B.3 Ajouter des options sur un plugin

- Utilisation des Flexforms : http://wiki.typo3.org/index.php/Extension_Development,_using_Flexforms

Modélisation de la base de données

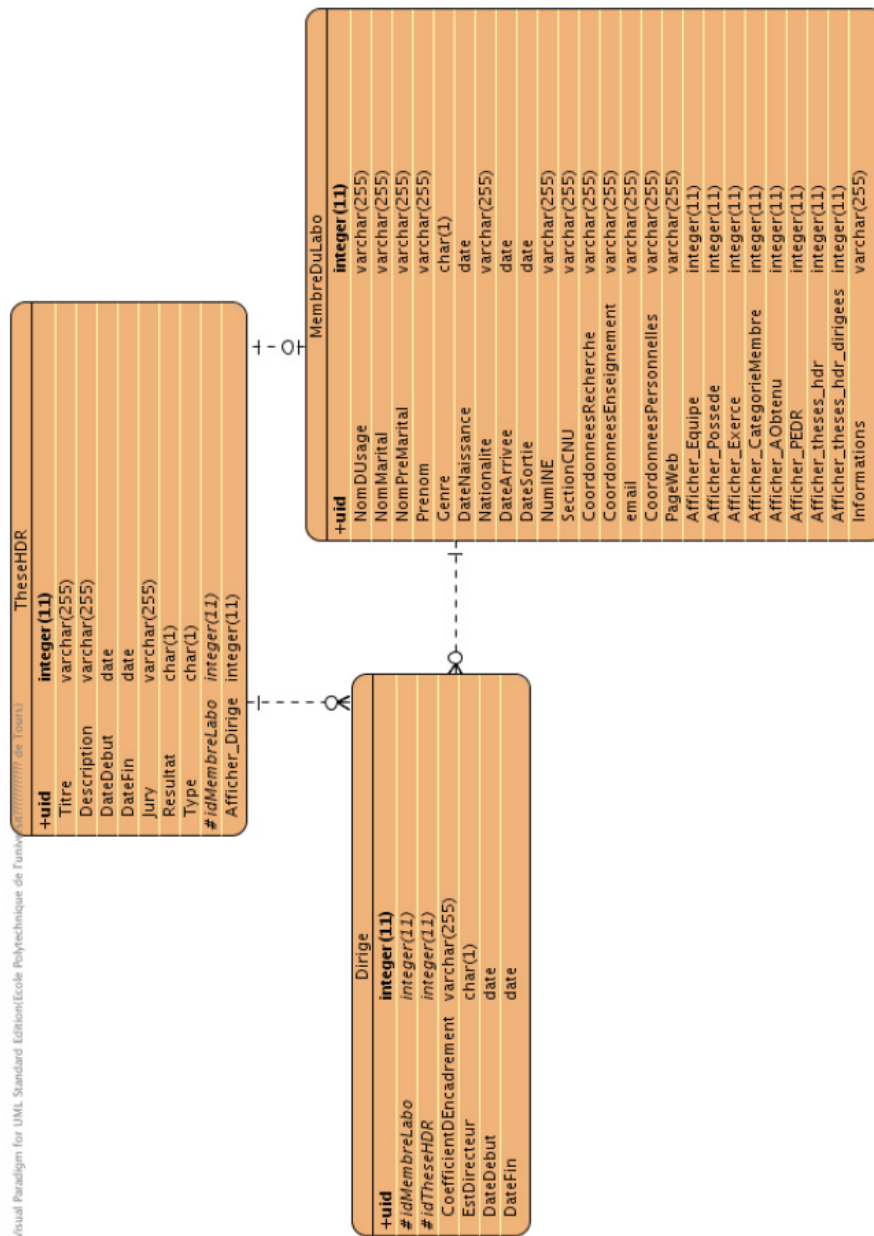
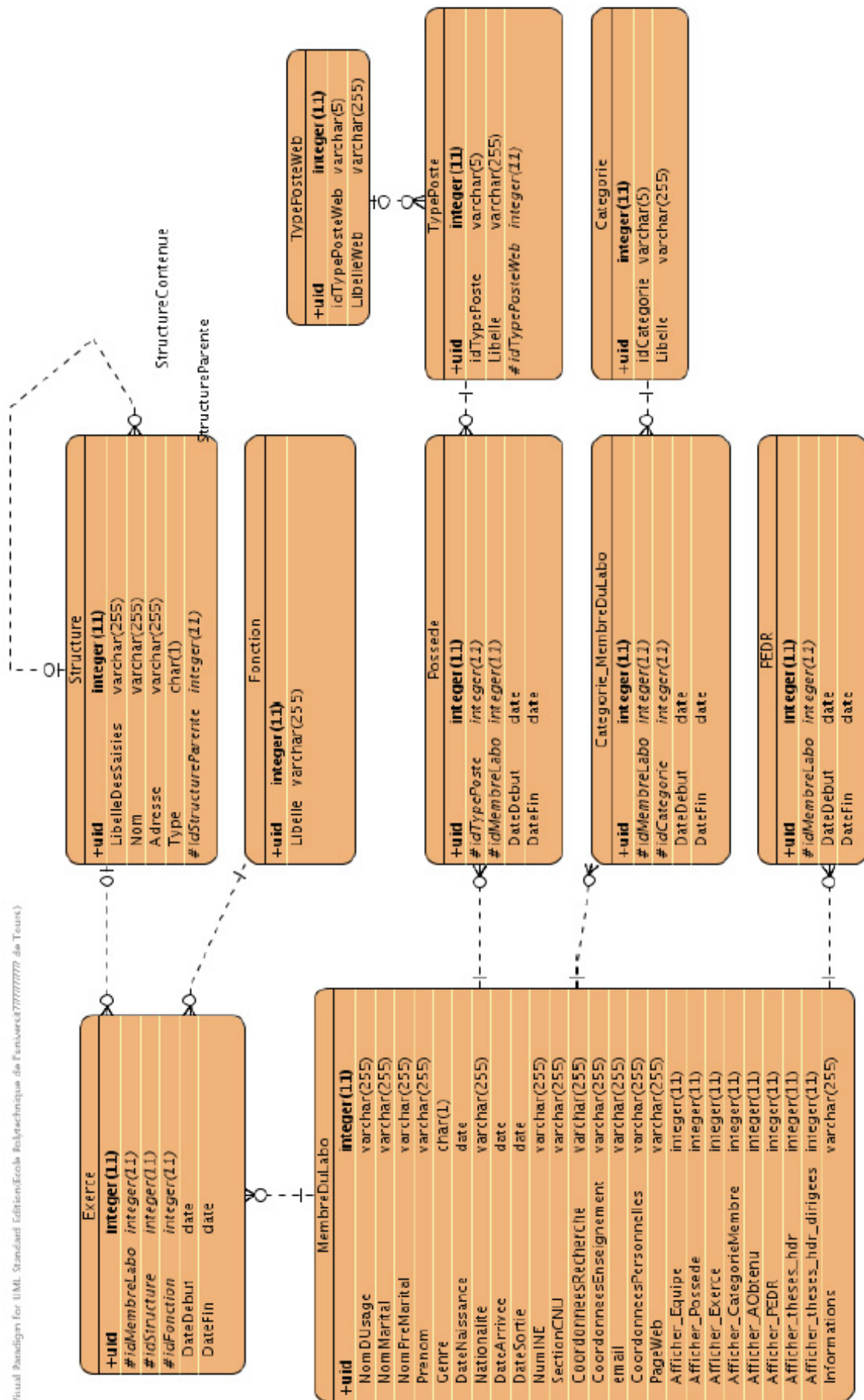
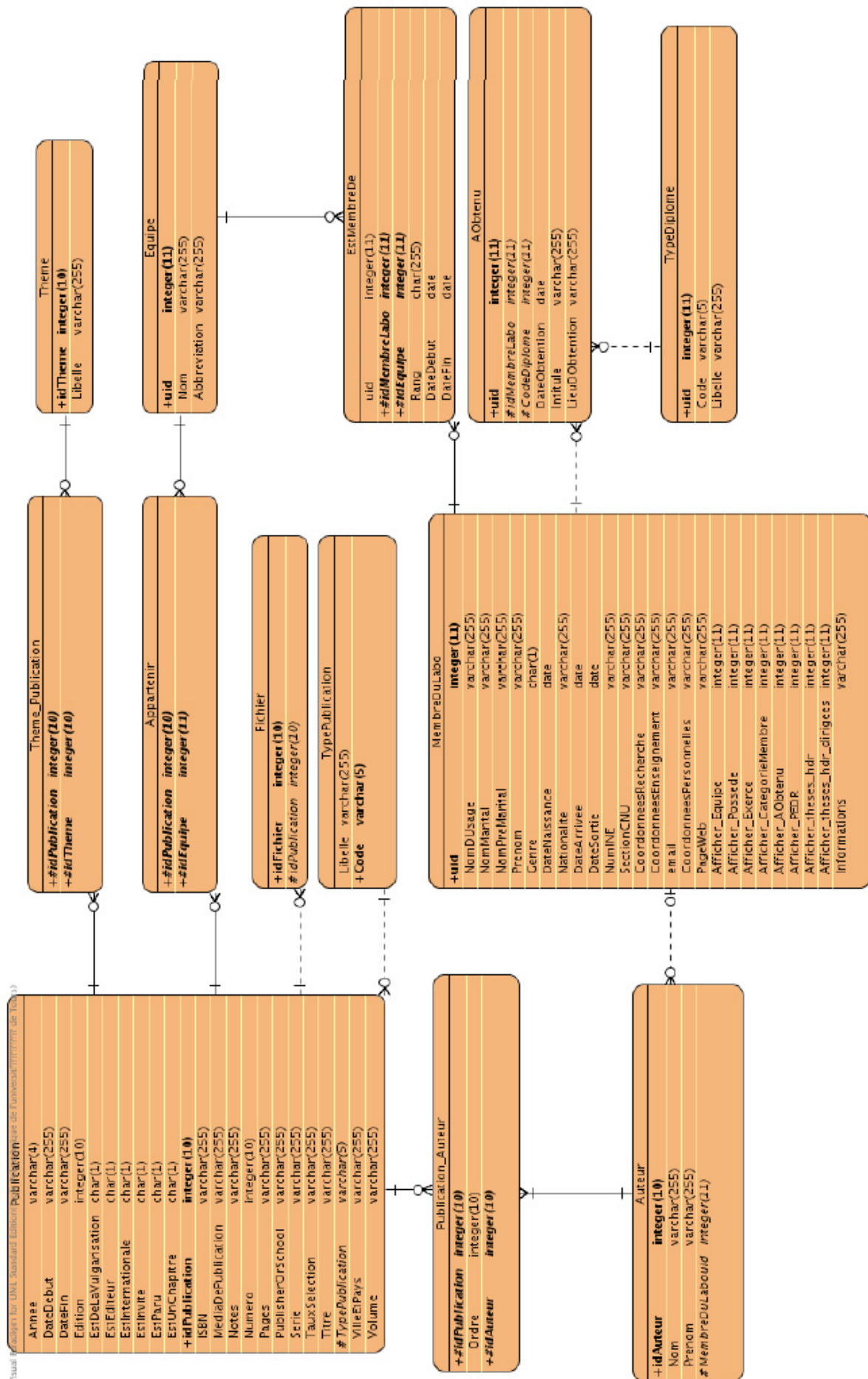


FIG. C.1 – Modélisation de la base de données 1





ANNEXE D

Installation et Utilisation de l'extension de Gestion de Membres

Cette extension permet de créer une base de données contenant un grand nombre d'informations sur les membres d'un laboratoire d'informatique à partir de formulaires et d'afficher une liste de membres suivant certaines contraintes choisies par lors de la configuration de l'extension.

D.1 Installation

L'installation de l'extension de Gestion de Membres est la même que pour toute extension Typo3. Dans le menu de droite, on choisi "Admin Tool", puis "Gest d'Extensions" et soit "Install Extensions" (si l'extension est déjà présente en local), soit "Import Extensions" (pour installer l'extension à partir du fichier .t3x)

Une fois l'extension installer sur le serveur, il faut ensuite l'activer dans Typo3. Pour cela, rendez-vous dans le gestionnaire d'extensions dans la partie "Frontend Plugins". ici vous trouverez celle qui nous intéresse :



	Frontend news submitter with RTE	<i>fe_rtenews</i>	1.4.1		
	Google Earth	<i>google_earth</i>	0.1.1		
	Indexed Search Engine	<i>indexed_search</i>	2.11.0		
	Indexed Search Engine Templates	<i>ch_is_templates</i>	1.1.14		
	Language Selection	<i>sr_language_menu</i>	1.4.0		
	Last update	<i>lastupdate</i>	3.1.7		
	Managing Members	<i>li_gest_membre_labo</i>	1.0.5		
	Mini news	<i>mininews</i>	1.0.0		
	Mozilla/Firefox Search Plugin Generator	<i>mozsearch</i>	0.1.0		
	News	<i>tt_news</i>	2.5.2		

FIG. D.1 – Installation : Frontend Plugins

Cliquez sur le "+" devant l'extension pour lancer son installation. Une nouvelle page décrivant les tables à créer ou modifier s'ouvre alors.

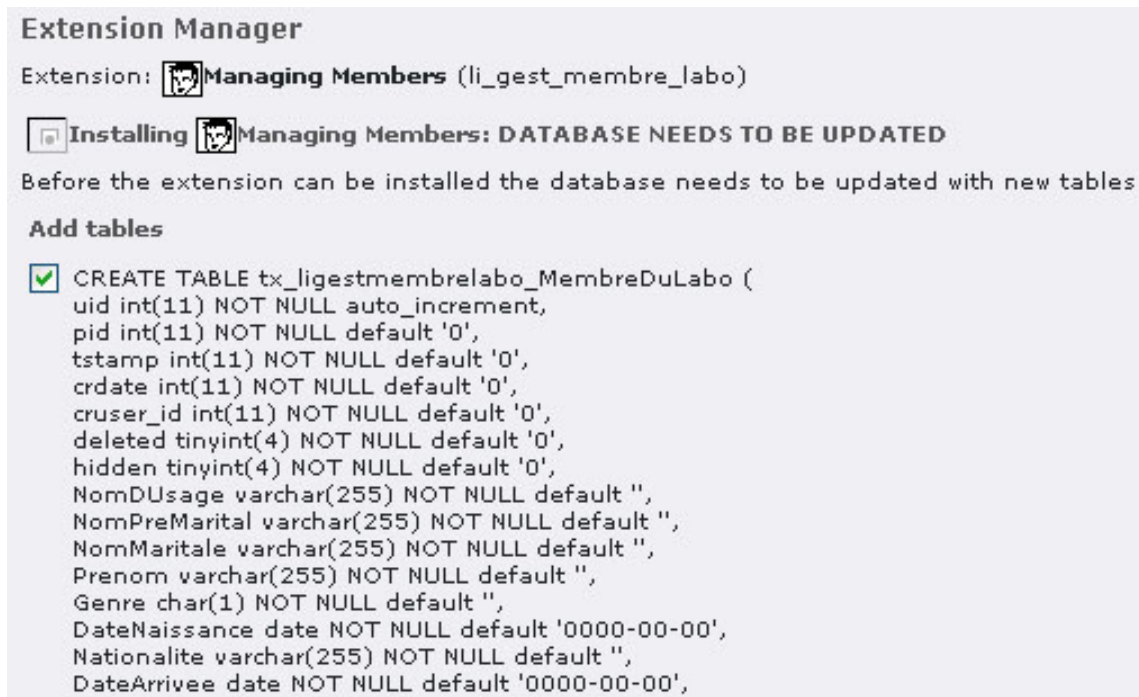


FIG. D.2 – Installation : Création des tables

Descendez jusqu'en bas de la page et cliquez sur "Make Update".

L'installation est maintenant terminée !

	Frontend news submitter with RTE	<i>fe_rtenews</i>	1.4.1			Local
	Google Earth	<i>google_earth</i>	0.1.1			Local
	Indexed Search Engine	<i>indexed_search</i>	2.11.0			System
	Indexed Search Engine Templates	<i>ch_is_templates</i>	1.1.14			Local
	Language Selection	<i>sr_language_menu</i>	1.4.0			Local
	Last update	<i>lastupdate</i>	3.1.7			Local
	Managing Members	<i>li_gest_membre_labo</i>	1.0.5			Local
	Mini news	<i>mininews</i>	1.0.0			Local
	Mozilla/Firefox Search Plugin Generator	<i>mozsearch</i>	0.1.0			Local
	News	<i>tt_news</i>	2.5.2			Local

FIG. D.3 – L'installation réussie

D.2 Remplissage de la base de données

D.2.1 Ajout d'enregistrements

Avant d'ajouter le moindre enregistrement, il est conseillé de créer un ou plusieurs dossiers système ou une ou plusieurs pages afin de stocker ces enregistrements.

Différents formulaires ont été créés pour permettre de remplir simplement la base de données. Un formulaire correspond à une seule et unique table. Ils sont disponibles à partir du menu "Web" puis "Liste"

en faisant un clic droit sur un dossier ou clic gauche sur l'icone du dossier et en choisissant "Nouveau".



FIG. D.4 – Accéder au menu

On peut maintenant choisir le formulaire que l'on souhaite :



FIG. D.5 – Choix du formulaire

Comme vous pouvez le remarquer, les formulaires ne sont pas triés par ordre alphabétique. En effet, ils sont triés dans un ordre plus "pratique" pour pouvoir repérer plus facilement les tables liées entre elles. C'est pourquoi le formulaire "Membre du labo" est positionné en premier, il est le point de départ de l'extension et toutes les autres tables lui sont plus ou moins associées. Ensuite les tables peuvent être regroupées par deux ou trois. Par exemple, le premier ensemble regroupe les formulaires "Fonction", "Structure" et "Exerce". En effet il est primordial de créer des enregistrements de fonctions et de structures avant d'utiliser le formulaire "Exerce" qui permet d'associer un membre à une fonction et à une structure.

Voici la liste des tables associées à chaque formulaire (les formulaires Exerce, Possede, Catégorie Membre Du Labo, Est Membre De, A Obtenu et PEDR sont disponible uniquement à partir du formulaire Membre du Labo) :

Nom du formulaire	Table associée	Contenu des tables
Membre du labo	tx_ligestmembrelabo_MembreDuLabo	Informations sur les membres du labo
Fonction	tx_ligestmembrelabo_Fonction	Les fonctions (directeur,...)
Structure	tx_ligestmembrelabo_Structure	Les structures d'appartenance
Exerce	tx_ligestmembrelabo_Exerce	Permet de lier un membre à une fonction dans une structure
Type Poste Web	tx_ligestmembrelabo_TypePosteWeb	Les types de poste simplifiés pour un affichage sur le site
Type Poste	tx_ligestmembrelabo_TypePoste	Les type de postes (ATER, MCF, Ingénieur, CR,...)
Possede	tx_ligestmembrelabo_Possede	Permet de lier un membre à un type de poste
Catégorie	tx_ligestmembrelabo_Categorie	Les catégories
Categorie Membre Du Labo	tx_ligestmembrelabo_CategorieMembre	Permet de lier un membre à une catégorie
Equipe	tx_ligestmembrelabo_Equipe	Les équipes
Est Membre De	tx_ligestmembrelabo_EstMembreDe	Permet de lier un membre à une équipe
Type Diplome	tx_ligestmembrelabo_TypeDiplome	Les diplômes
A Obtenu	tx_ligestmembrelabo_AObtenu	Permet d'ajouter un diplôme à un membre
PEDR	tx_ligestmembrelabo_PEDR	Permet d'ajouter une PEDR à un membre

TAB. D.1 – Tables associées à chaque formulaire

Une fois le formulaire choisi, vous pouvez maintenant remplir les champs avec les informations que vous souhaitez. Pour sauvegarder l'enregistrement, il suffit de cliquez sur la disquette en haut de la page.

Voici un exemple de formulaire :

FIG. D.6 – Formulaire

Les champs des formulaires ont tous les mêmes caractéristiques :

- Un champ contenant uniquement une boîte dialogue tel que le champ "Le nom de famille avant tout mariage" peut contenir tout type de caractères. Si ce champ est précédé d'un point d'exclamation comme pour le champ "Prénom", celui signifie qu'il est obligatoire.
- L'enregistrement ne sera donc pas accepté si le champ est vide. Les boîtes de sélection comme celle du champ "Genre" s'utilisent comme n'importe quelle boîte de sélection. Si cela est nécessaire, un champ vide est disponible.
- Tous les champs comportant des dates doivent être saisis sous la forme aaaa-mm-jj. Ainsi, le 11 décembre 1816 (Java est rendu aux Pays-Bas par les anglais...) devient 1816-12-11. Attention : pour les dates comportant des nombres à 1 chiffre, les zéros sont obligatoires. Ainsi, le 2 avril 1985 devient 1985-04-02. Une date peut être obligatoire (un point d'exclamation est situé devant). Il est à noter que si vous sauvegardez sans avoir rempli une date obligatoire, ce champ sera automatiquement rempli par la date du jour. Si vous laissez un champ date vide et que celui-ci n'était pas obligatoire, il sera automatiquement rempli par la date 0000-00-00. Une date 0000-00-00 concernant la fin de quelque chose sera interprétée par l'extension comme une date infinie (Si par exemple un membre est toujours au laboratoire mais on ne sait pas encore quand il va partir, on remplit ce champ par 0000-00-00).

D.2.2 Modifications d'enregistrements

Modifier un enregistrement est très simple. Il suffit de cliquer (avec le bouton gauche) sur la petite page blanche située à gauche du nom de l'enregistrement que l'on souhaite modifier (celui-ci est situé dans le dossier dans lequel il a été enregistré...) puis de sélectionner "Editer" dans le menu déroulant. Le formulaire contenant les informations présente dans la base de données est alors accessible. Dès que les modifications ont été apportées, vous pouvez sauvegarder l'enregistrement en cliquant sur la disquette.



FIG. D.7 – Modifications

D.2.3 Formulaires multilingues

6 formulaires offrent la possibilité de créer des enregistrements multilingues. Il s'agit des formulaires : Catégorie, Equipe, Fonction, TypeDiplome, TypePoste et TypePosteWeb.

Pour créer un enregistrement dans une langue autre que celle par défaut, vous devez tout d'abord créer un enregistrement comme s'il s'agissait d'un enregistrement "normal".



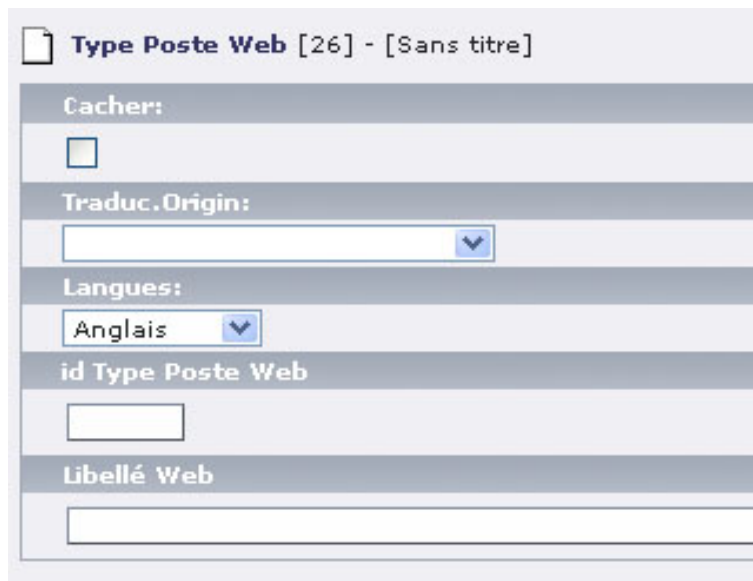
FIG. D.8 – Multilingue : formulaire

Dans la boîte de sélection "Langues", choisissez la langue de l'enregistrement :



FIG. D.9 – Multilingue : choix de la langue

Cliquez sur la disquette pour enregistrer. Une nouvelle boîte de sélection, "Traduc.Origin", vous permet maintenant de choisir l'enregistrement à partir duquel vous souhaitez faire la traduction.



The screenshot shows a web form titled "Type Poste Web [26] - [Sans titre]". It contains several fields: "Cacher:" with a checkbox, "Traduc.Origin:" with a dropdown menu, "Langues:" with a dropdown menu showing "Anglais", "id Type Poste Web" with a text input, and "Libellé Web" with a text input. The "Traduc.Origin:" dropdown is currently empty, showing only a downward arrow.

FIG. D.10 – Multilingue : choix de la traduction d'origine

Une fois l'enregistrement d'origine sélectionné, sauvegardez une nouvelle fois. Vous obtenez maintenant, pour chaque champ, toutes les informations saisies pour la langue par défaut.



The screenshot shows the same web form as Figure D.10, but now the "Traduc.Origin:" dropdown menu is set to "Docteur". Below the "Langues:" dropdown, there is a green bar with the French flag and the text "Par défaut". Below the "id Type Poste Web" text input, there is a green bar with the French flag and the number "3". Below the "Libellé Web" text input, there is a green bar with the French flag and the text "Docteur".

FIG. D.11 – Multilingue : affichage de la traduction d'origine

Après avoir complété tous les champs avec la traduction, enregistrez une dernière fois.



FIG. D.12 – Multilingue : remplissage des champs

D.2.4 Cas particulier du formulaire "Membre du Labo"

Plusieurs boîtes de sélections ont été ajoutées au formulaire "Membre du Labo" pour pouvoir modifier/créer/supprimer plus facilement les enregistrements liés au membre courant. Les boîtes ayant cette propriété concernent : les équipes, les postes, les fonctions et structures, les catégories, les diplômes et les PEDR.

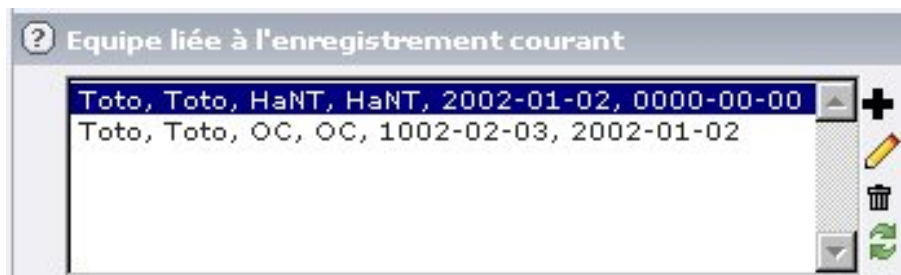


FIG. D.13 – Champs supplémentaires

Quelque soit l'enregistrement sélectionné dans l'une de ces boîtes, cela n'a pas d'importance. En effet, même si le numéro de l'enregistrement sélectionné est enregistré dans la base de données, il n'est pas réutilisé par la suite dans l'extension.

Lorsque vous cliquez sur le bouton "+", un nouveau formulaire s'ouvre. Le nouvel enregistrement sera ajouté dans le même dossier que celui du membre duquel vous avez appelé le formulaire. Attention : un enregistrement est automatiquement créé dès le clic sur le bouton. Si vous cliquez par erreur sur le bouton, vous devrez supprimer cet enregistrement.

Lorsque vous cliquez sur le bouton en forme de crayon, une nouvelle fenêtre s'ouvre pour vous permettre de modifier l'enregistrement sélectionné.

Lorsque vous cliquez sur le bouton représentant une corbeille, l'enregistrement sélectionné sera supprimé.

Le bouton en forme de flèches permet de réactualiser la page pour prendre en compte des modifications apportés par des modifications dans les formulaires d'ajout ou de modifications.

Voici les formulaires vers lesquels vous allez être renvoyé en cliquant sur les boutons d'ajout ou de modification :

Nom du champ	Nom du formulaire
Fonctions et Structures liées à l'enregistrement courant	Exerce
Postes liés à l'enregistrement courant	Possede
Catégories liées à l'enregistrement courant	Categorie Membre Du Labo
Equipe liée à l'enregistrement courant	Est Membre De
Diplomes liés à l'enregistrement courant	A Obtenu
PEDR liées à l'enregistrement courant	PEDR

TAB. D.2 – Formulaire associé aux boutons d'ajout et de modification

D.3 Insertion et configuration du plugin dans une page

D.3.1 Insertion du plugin dans une page

Rendez-vous sur la page sur laquelle vous souhaitez afficher une liste de membres (à l'aide du menu "Page"). Vous devriez obtenir une page contenant à peu près (ici la page s'appelle "Membres" et ne contient rien) :



FIG. D.14 – Insertion du plugin 1

Cliquez sur le bouton "Créer un nouvel élément" (l'image avec le petit "+" entouré de vert). Une nouvelle page contenant tous les éléments qu'il est possible d'insérer s'ouvre. Choisissez le plugin "Gestion de membres".



FIG. D.15 – Insertion du plugin 2

Vous pouvez maintenant commencer la configuration générale du plugin. Sur cette première page vous pouvez choisir le titre à afficher sur la page au dessus de la liste des membres (ici "Enseignants-chercheurs") :

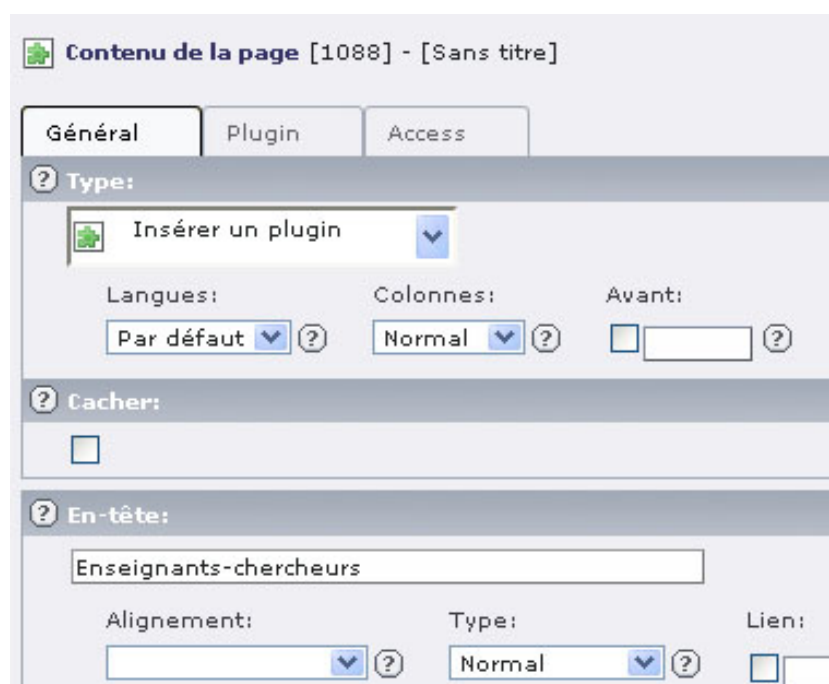


FIG. D.16 – Insertion du plugin 3

Vous pouvez désormais sauvegarder la page en cliquant sur la disquette en haut de la page. En vous arrêtant là, le frontend affichera la liste de tous les membres enregistrés dans la base de données étant encore au laboratoire :

Membres

Enseignants-chercheurs

MEMBRE1 PrenomMembre1

MEMBRE10 PrenomMembre10

MEMBRE11 PrenomMembre11

MEMBRE3 PrenomMembre3

MEMBRE5 PrenomMembre5

MEMBRE6 PrenomMembre6

FIG. D.17 – Insertion du plugin 4

D.3.2 Configuration du plugin

L'extension de Gestion des membres permet d'afficher toutes sortes de liste en sélectionnant des filtres ou en créant soit-même sa requête ou son gabarit. Pour cela, il faut cliquer sur l'onglet "plugin" lors de la configuration de celle-ci. Il y a alors plusieurs onglets disponibles : "Données", "Filtres", "Affichage", "Séparateurs", "Requête" et "Gabarit".

Onglet Données

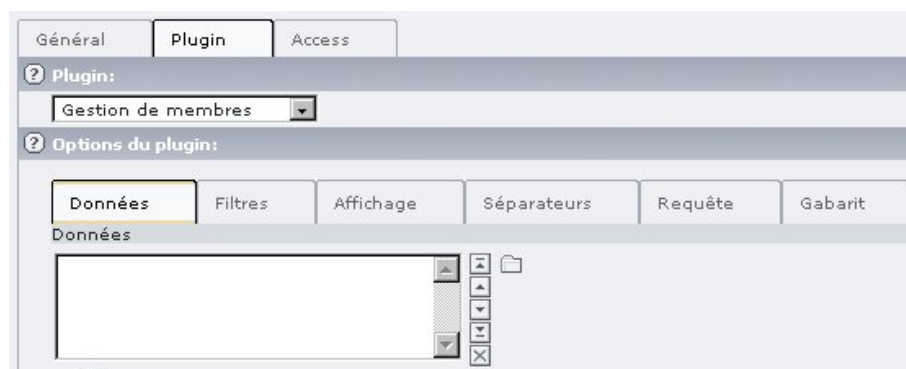


FIG. D.18 – Onglet Données

L'onglet données permet de sélectionner les membres en sélectionnant un dossier (en cliquant sur l'image représentant un dossier) dans laquelle se trouve les enregistrements à afficher.

La sélection par dossiers comporte quelques particularités :

- Si aucun dossier n'est sélectionné, cela est considéré comme si tous les dossiers étaient sélectionnés.
- Il est possible de sélectionner plusieurs dossiers.
- Lorsqu'un dossier est sélectionné, tous ses sous-dossiers sont sélectionnés.

Onglet Filtres

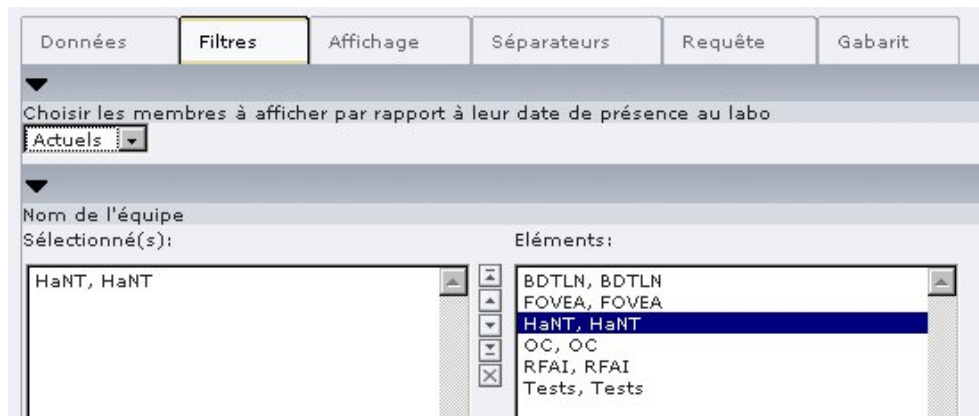


FIG. D.19 – Onglet Filtres

L'onglet filtres permet de sélectionner différentes catégories de membres. Il est ainsi possible de sélectionner les membres par rapport :

- aux dates d'arrivée et de départ du membre dans le labo (Actuels, Anciens, Tous). Par défaut, "Actuels" est sélectionné.
- aux équipes dans lesquelles le membre a été affecté. Il est possible d'ajouter une option selon que le membre est toujours dans une des équipes sélectionnées (Actuels), a été dans une des équipes sélectionnées (Anciens), ou a été ou est toujours dans l'un des équipes sélectionnées (Tous). Par défaut, "Tous" est sélectionné.
- aux types de postes des membres. Si aucun poste n'est sélectionné, cela est considéré comme si tous les postes étaient sélectionnés. Il est possible d'ajouter une option sur la "période" à laquelle ce poste à été exercé (Actuels, Anciens, Tous). Par défaut, "Tous" est sélectionné.
- à la catégorie des membres. Si aucune catégorie n'est sélectionné, cela est considéré comme si toutes les catégories étaient sélectionnés. Il est possible d'ajouter une option sur la "période" à laquelle cette catégorie à été occupée (Actuels, Anciens, Tous). Par défaut, "Tous" est sélectionné.
- aux structures dans lesquelles travaille ou ont travaillé les membres. Si aucune structure n'est sélectionné, cela est considéré comme si toutes les structures étaient sélectionnés. Il est possible d'ajouter une option sur la "période" à laquelle le membre a appartenu à une des structures sélectionnées (Actuels, Anciens, Tous). Par défaut, "Tous" est sélectionné.
- aux fonctions des membres. Si aucune fonction n'est sélectionné, cela est considéré comme si toutes les fonctions étaient sélectionnés. Il est possible d'ajouter une option sur la "période" à laquelle cette fonction a été occupée (Actuels, Anciens, Tous). Par défaut, "Tous" est sélectionné.
- aux diplômes des membres. Si aucun diplôme n'est sélectionné, cela est considéré comme si tous les diplômes étaient sélectionnés.
- à l'obtention ou non d'une PEDR (Prime d'Encadrement Doctoral et de Recherche) par le membre. Il y a 4 possibilités : soit le membre a actuellement une PEDR (Actuels), a eu une PEDR (Anciens), a eu ou a toujours une PEDR (Tous les PEDR (Actuels et Anciens)), ou encore a, a eu ou n'a j'amaïs

eu de PEDR (Tous (PEDR ou non PEDR)). Par défaut Tous (PEDR ou non PEDR) est sélectionné.

Dans l'exemple affiché ci-dessus on sélectionne tous les membres actuellement au laboratoire ayant eu soit un poste de Doctorant soit un poste de Docteur, soit un poste de Professeur mais qui ne l'exerce plus et qui ont été ou sont toujours dans la catégorie "catégorie1".

Onglet Affichage

L'onglet Affichage permet de personnaliser l'affichage de certains champs dans le Frontend.

Lorsque l'on a une liste de membres, dans certains cas, il peut être utile de vouloir mettre en évidence les membres d'une équipe en particulier. Dans cet onglet, il est possible de sélectionner les équipes que l'on veut surligner ainsi que définir les balises que l'on souhaite utiliser pour marquer cette différence (en utilisant des balises HTML ou XHTML). Cette option est utilisée dans le gabarit à partir de la balise "MembresSurlignes".

FIG. D.20 – Onglet Affichage : Surbrillance

Par exemple, sur une liste contenant des membres des équipes OC et HaNT, en utilisant les options ci-dessus pour mettre en gras les membres de l'équipe HaNT, on obtient le résultat suivant :

AUPETIT Sébastien
BILLAUT Jean-Charles
BONHOMME Patrice
BOUQUARD Jean-Louis
CLAIR Romain
COLAS Sonia
ESSWEIN Carl
GAUCHER Pierre
 HUYNH TUONG Nguyen
 KERGOSIEN Yannick
 LABANE Malika

FIG. D.21 – Onglet Affichage : Surbrillance 2

Vous pouvez également personnaliser l'affichage des dates en choisissant le format des dates. La syntaxe utilisée par ce champ est la même que pour le premier paramètre de la fonction `date()` de php (<http://fr.php.net/date>). Par défaut, la date sera affichée au format français (d/m/Y).



FIG. D.22 – Onglet Affichage : Format des dates

Il est également possible de modifier certains textes afficher comme par exemple le texte permettant de remplacer une date 0000-00-00.



FIG. D.23 – Onglet Affichage

Onglet Séparateurs

FIG. D.24 – Onglet Séparateur

L'onglet séparateur permet de personnaliser l'affichage des séparateurs dans le Frontend. Les séparateurs sont des marqueurs utilisés dans la construction d'un gabarit (template) qui permettent d'afficher un "séparateur" (une virgule, un point, deux points ou n'importe quel texte) uniquement si la balise contenant le texte provenant de la base de données n'est pas vide. Tous les marqueurs de séparation sont de la même forme : Par exemple pour un marqueur `###NomDuChamp###` on a un marqueur `###NomDuChamp_Separateur###` associé.

Onglet Requête

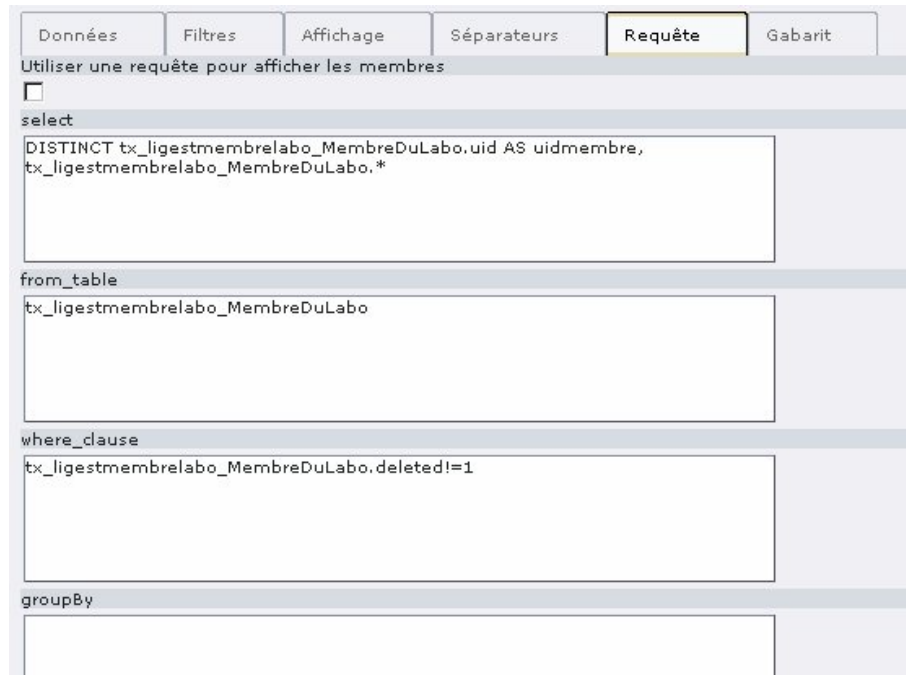
L'Onglet Requête permet de créer sa propre requête. Pour que la requête soit effectivement utilisée, il faut tout d'abord cocher la case "Utiliser une requête pour afficher les membres".

La construction de la requête se fait de la même manière que tout requête SQL, à l'exception près, qu'elle est séparée en plusieurs parties. Ainsi vous devez écrire la partie SELECT de votre requête dans le champs SELECT, la partie FROM dans le champs FROM,... Les mots-clés SELECT, FROM, WHERE, ORDER BY,... ne doivent pas être indiqués.

Pour que cette requête fonctionne, certains informations sont obligatoires. Vous devez absolument prendre pour requête de départ celle qui est sur l'exemple ci-dessus, à savoir :

- "DISTINCT tx_ligestmembrelabo_MembreDuLabo.uid AS uidmembre, tx_ligestmembrelabo_MembreDuLabo.*" dans le champ SELECT.
- "tx_ligestmembrelabo_MembreDuLabo" dans le champ FROM.
- "tx_ligestmembrelabo_MembreDuLabo.deleted<>1" dans le champ WHERE.

Cette requête permet d'afficher tous les membres qui n'ont pas été effacés. Vous pouvez ensuite rajouter vos propres conditions...



Données Filtres Affichage Séparateurs **Requête** Gabarit

Utiliser une requête pour afficher les membres ☐

select

DISTINCT tx_ligestmembrelabo_MembreDuLabo.uid AS uidmembre,
tx_ligestmembrelabo_MembreDuLabo.*

from_table

tx_ligestmembrelabo_MembreDuLabo

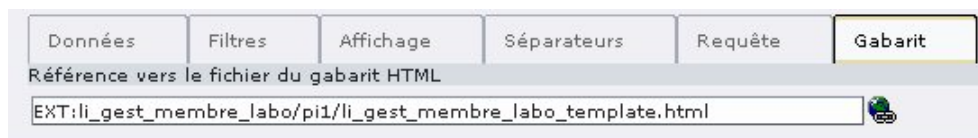
where_clause

tx_ligestmembrelabo_MembreDuLabo.deleted!=1

groupBy

FIG. D.25 – Onglet Requete

Onglet Gabarit



Données Filtres Affichage Séparateurs Requête **Gabarit**

Référence vers le fichier du gabarit HTML

EXT:li_gest_membre_labo/pi1/li_gest_membre_labo_template.html

FIG. D.26 – Onglet Gabarit

L'Onglet gabarit permet de sélectionner le gabarit que vous souhaitez utiliser pour afficher votre page. Il est possible d'utiliser soit l'un des gabarits proposés dans l'extension, soit d'en créer un. Pour plus de détails veuillez vous référer à la partie "Création et utilisation de gabarits".

D.4 Création et utilisation de gabarits

Dans Typo3, les gabarits (ou Templates) permettent de personnaliser l'affichage dans le Frontend à l'aide de balises et de marqueurs particuliers.

Dans cette extension, deux gabarits sont disponibles mais vous pouvez créer le votre pour obtenir un affichage personnalisé en fonction de vos besoins.

Lors de la configuration du plugin, pour indiquer le gabarit à utiliser, il faut remplir le champs de "Référence vers le fichier du gabarit HTML" de l'onglet "Gabarit".

Les deux gabarits disponibles sont :

- EXT:li_gest_membre_labo/pi1/li_gest_membre_labo_template.html

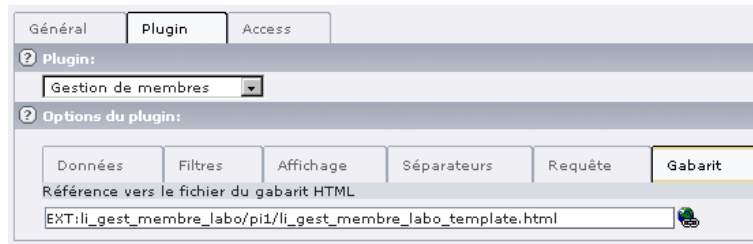


FIG. D.27 – Onglet gabarit

– EXT:li_gest_membre_labo/pi1/li_gest_membre_labo_template_libelleweb.html

Le premier permet d'afficher la liste des membres (NOMs et Prénoms). Le deuxième affiche la liste des membres avec les NOMs et Prénoms ainsi que le Libellé Web leur correspondant. Les noms et prénoms renvoient vers le site internet du membre si celui-ci en possède un.

D.4.1 Créer son gabarit

Un gabarit est créé sur la base d'une page HTML (en-tête, corps,...). Dans le corps de cette page (body), on peut ensuite rajouter différentes balises pour obtenir un affichage personnalisé. Pour l'extension de Gestion des Membres certaines balises sont obligatoires, il s'agit des balises TEMPLATE (début du gabarit), CONTENT (début d'un contenu) et ITEM (liste des informations à afficher) qui permettent de structurer l'affichage.

Les balises s'utilisent toutes de la même façon :

- une balise de départ : `<!-- ###NOM### begin-->`
- une balise de fin : `<!-- ###NOM### end-->`

La structure de base d'un gabarit est :

```
<!-- ###TEMPLATE### begin-->
  <!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
    <!-- ###ITEM### end -->
  <!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->
```

Nous pouvons maintenant ajouter dans ce code différentes balises, qui permettent un affichage de plusieurs éléments (quand la base de données renvoie plusieurs résultats) ou des marqueurs, qui n'affiche qu'un seul texte.

Pour chaque marqueur `###NomMarqueur###` présent dans cette extension, un second marqueur, appelé ici "séparateur", permet d'afficher un texte personnalisable lors de la configuration du plugin. Ce texte sera affiché uniquement si le marqueur `###NomMarqueur###` affiche un texte à l'écran (texte non vide), dans le cas contraire, le séparateur ne sera pas affiché. Tous les séparateurs ont la même forme : `###NomMarqueur_Separateur###`.

Affichage des informations générales des membres

L'affichage des informations générales d'un membre se fait essentiellement à l'aide de marqueurs (notés **###NomDuMarqueur###** à insérer dans la balise ITEM. Ces marqueurs sont :

Nom du marqueur	Caractéristique
NomDUsage	nom d'usage du membre
NOMDUSAGE	nom d'usage du membre en majuscule
NomMarital	nom marital du membre
NOMMARITAL	nom marital du membre en majuscule
NomPreMarital	nom prémarital du membre
NOMPREMARITAL	nom prémarital du membre en majuscule
Prenom	prénom du membre
PRENOM	prénom du membre en majuscule
InitialePrenom	première lettre du prénom du membre
InitialeNom	première lettre du nom du membre
Genre	genre du membre (M pour masculin ou F pour féminin)
DateNaissance	date de naissance du membre
Nationalite	nationalité du membre
NATIONALITE	nationalité du membre en majuscule
DateArrivee	date d'arrivée du membre au laboratoire
DateSortie	date de sortie du membre du laboratoire
NumINE	numéro INE du membre
SectionCNU	numéro(s) de section CNU du membre
CoordonneesRecherche	coordonnées recherche du membre
CoordonneesRecherche_Ligne	coordonnées recherche du membre sur une seule ligne
CoordonneesEnseignement	coordonnées enseignement du membre
CoordonneesEnseignement_Ligne	coordonnées enseignement du membre sur une seule ligne
CoordonneesPersonnelles	coordonnées personnelles du membre
CoordonneesPersonnelles_Ligne	coordonnées personnelles du membre sur une seule ligne
email	email du membre
PageWeb	page web du membre

TAB. D.3 – Marqueurs pour le membre

Une balise **###PageWebLien###** permet de créer un lien autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

Par exemple pour afficher un lien autour du nom et du prénom d'un membre on écrit :

```
<!-- ###ITEM### begin -->
    <p>
        <!--###PageWebLien###          begin          -->###NOMDUSAGE###
###Prenom###<!--###PageWebLien### end -->
    </p>
<!-- ###ITEM### end -->
```

Une balise **###MembresSurlignes###** permet de surligner les membres d'une équipe particulière (ou

de plusieurs équipe). Plus d'information ici : 60

Affichage des informations concernant l'équipe d'un membre

L'affichage des informations concernant l'équipe d'un membre se fait à l'aide des balises `###EQUIPE###` et/ou `###EQUIPE_DERNIER###`.

`###EQUIPE###` permet d'afficher les informations sur toutes les équipes dans lesquelles un membres a été affecté. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Equipe_Nom	nom de l'équipe
Equipe_Abréviation	abréviation utilisée pour désigner l'équipe
EstMembreDe_Rang	rang du membre dans l'équipe
EstMembreDe_DateDebut	date d'arrivée du membre dans l'équipe
EstMembreDe_DateFin	date de départ du membre de l'équipe

TAB. D.4 – Marqueurs pour l'équipe

`###EQUIPE_DERNIER###` permet d'afficher les informations sur la dernière équipe dans laquelle un membres a été affecté. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Equipe_Nom_Dernier	nom de la dernière équipe du membre
Equipe_Abréviation_Dernier	abréviation utilisée pour désigner sa dernière équipe
EstMembreDe_Rang_Dernier	rang du membre dans sa dernière équipe
EstMembreDe_DateDebut_Dernier	date d'arrivée du membre dans sa dernière équipe
EstMembreDe_DateFin_Dernier	date de départ du membre de sa dernière équipe

TAB. D.5 – Marqueurs pour la dernière équipe

Affichage des informations concernant les catégories d'un membre

Les informations concernant les catégories d'un membre peuvent être affichées par l'intermédiaire des balises `###CATEGORIES###` et/ou `###CATEGORIES_DERNIER###`.

`###CATEGORIES###` permet d'afficher **toutes** les catégories auxquelles le membre concerné a appartenu. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Categorie_Libelle	libellé de la catégorie
CategorieMembre_DateDebut	date d'arrivée du membre dans la catégorie
CategorieMembre_DateFin	date de départ du membre de la catégorie

TAB. D.6 – Marqueurs pour les catégories

Note : Les catégories sont affichées de la plus récente à la plus ancienne.

###CATEGORIES_DERNIER### permet d'afficher la **dernière** catégorie auquel le membre concerné a appartenu. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Categorie_Libelle_Dernier	libellé de la catégorie
CategorieMembre_DateDebut_Dernier	date d'arrivée du membre dans la catégorie
CategorieMembre_DateFin_Dernier	date de départ du membre de la catégorie

TAB. D.7 – Marqueurs pour la dernière catégorie

Affichage des informations concernant les diplômes d'un membre

Sur le même principe que pour l'affichage des catégories, il est possible d'afficher soit tous les diplômes avec la balise ###DIPLOMES###, soit uniquement le dernier avec ###DIPLOMES_DERNIER###.

Pour la balise ###DIPLOMES###, les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Diplomes_Libelle	libellé du diplôme
Diplomes_Code	code du diplôme
Diplomes_DateObtention	date d'obtention du diplôme
Diplomes_Intitule	intitulé du diplôme
Diplomes_LieuDObtention	lieu d'obtention du diplôme

TAB. D.8 – Marqueurs pour les diplômes

Note : Les diplômes sont affichés du plus récent au plus ancien.

Pour la balise ###DIPLOMES_DERNIER###, les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Diplomes.Libelle.Dernier	libellé du dernier diplôme
Diplomes.Code.Dernier	code du diplôme
Diplomes.DateObtention.Dernier	date d'obtention du dernier diplôme
Diplomes.Intitule.Dernier	intitulé du dernier diplôme
Diplomes.LieuDObtention.Dernier	lieu d'obtention du dernier diplôme

TAB. D.9 – Marqueurs pour le dernier diplôme

Affichage des informations concernant les postes d'un membre

Deux balises sont également disponibles : `###POSTES###` et `###POSTES.DERNIER###`.

`###POSTES###` permet d'afficher les informations de tout les types de poste du membres à l'aide des marqueurs :

Nom du marqueur	Caractéristique
Postes.LibelleWeb	libellé web du poste
Postes.Libelle	libellé du poste
Postes.DateDebut	date de début du membre au poste concerné
Postes.DateFin	date de départ du membre au poste concerné

TAB. D.10 – Marqueurs pour les postes

Note : Les postes sont affichés du plus récent au plus ancien.

`###POSTES.DERNIER###` permet d'afficher les informations de du dernier types de poste du membres à l'aide des marqueurs :

Nom du marqueur	Caractéristique
Postes.LibelleWeb.Dernier	libellé web du dernier poste
Postes.Libelle.Dernier	libellé du dernier poste
Postes.DateDebut.Dernier	date de début du membre à son dernier poste
Postes.DateFin.Dernier	date de départ du membre de son dernier poste

TAB. D.11 – Marqueurs pour le dernier poste

Affichage des informations concernant les fonctions dans une structure d'un membre

Les informations concernant la fonction d'un membre et la structure dans laquelle il l'exerce sont affiché avec les balises `###FONCTIONS_STRUCTURES###` (pour afficher toutes les fonctions et structures) et `###FONCTIONS_STRUCTURES.DERNIER###` (pour afficher uniquement la dernière fonction et la dernière structure).

Avec la balise `###FONCTIONS_STRUCTURES###` :

Nom du marqueur	Caractéristique
Fonctions_Libelle	libellé de la fonction
Structures_LibelleDesSaisies	libellé de la structure
Structures_Nom	nom de la structure
Structures_Adresse	adresse de la structure
Structures_Type	type de la structure
FonctionsStructures_DateDebut	date d'arrivée du membre à un poste dans une structure
FonctionsStructures_DateFin	date de départ du membre d'un poste dans une structure

TAB. D.12 – Marqueurs pour les fonctions et structures

Note : Les fonctions/structures sont affichées des plus récentes aux plus anciennes.

Avec la balise `###FONCTIONS_STRUCTURES_DERNIER###` :

Nom du marqueur	Caractéristique
Fonctions_Libelle_Dernier	libellé de la dernière fonction
Structures_LibelleDesSaisies_Dernier	libellé de la dernière structure
Structures_Nom_Dernier	nom de la dernière structure
Structures_Adresse_Dernier	adresse de la dernière structure
Structures_Type_Dernier	type de la dernière structure
FonctionsStructures_DateDebut_Dernier	date d'arrivée du membre à son dernier poste dans sa dernière structure
FonctionsStructures_DateFin_Dernier	date de départ du membre de son dernier poste dans sa dernière structure

TAB. D.13 – Marqueurs pour les dernières fonctions et structures

Exemple de gabarit

Voici un exemple de gabarit permettant d'afficher une liste de membre avec leur nom, prénom et leur dernier type de poste simplifié pour le web. Lorsque cela est possible, un lien vers la page du membre est disponible en cliquant sur le nom ou le prénom. Enfin, si une équipe a été sélectionnée dans les options d'affichage, elle sera mise en surbrillance en fonction des balises choisies.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
    <title>Liste de membres</title>
    <link href="/li_gest_membre_labo_stylesheet.css"
        rel="stylesheet" type="text/css" />
</head>

<body>

<!-- ###TEMPLATE### begin-->
```

```
<!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
        <p>
            <!--###PageWebLien### begin -->
                <!--###MembresSurlignes### begin -->
                    ###NOMDUSAGE### ###Prenom###
                <!--###MembresSurlignes### end -->
            <!--###PageWebLien### end -->
            <!-- ###POSTES.DERNIER### begin -->
                ###Postes_LibelleWeb_Dernier_Separateur###
                ###Postes_LibelleWeb_Dernier###
            <!-- ###POSTES.DERNIER### end -->
        </p>
    <!-- ###ITEM### end -->
<!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->

</body>
</html>
```

Utilisez votre gabarit

Une fois votre gabarit terminé. Stockez-le sur le serveur à l'aide du menu "Media". Rendez-vous ensuite dans la configuration de l'extension dans l'Onglet Gabarit et sélectionné votre gabarit à l'aide en cliquant sur l'image à droite du champ.

Installation et Utilisation de l'extension de Gestion de Thèses

Cette extension permet d'afficher une liste de thèses et/ou d'Habilitations à Diriger des Recherches (HDR).

Pour utiliser cette extension, vous devez au préalable avoir installé l'extension de Gestion des Membres. L'utilisation de cette extension est similaire à celle de Gestion des Membres que ce soit pour l'installation, l'ajout ou la modification d'enregistrements dans la base de données, l'ajout du plugin sur une page,...

E.1 Installation

Pour installer cette extension, veuillez vous référer à l'installation de l'extension de Gestion des membres. En effet, son installation est totalement similaire.

E.2 Remplissage de la base de données

E.2.1 Ajout et modifications d'enregistrements

L'ajout et la modification d'enregistrements se font de la même façon que pour l'extension de Gestion des membres.

Ici les formulaires disponibles sont : "Informations sur la thèse/HDR" et uniquement lorsque l'on ajoute un enregistrement à un dossier : "Association : Direction de thèses/HDR" et "Association : Cotutelle", si l'on souhaite ajouter des enregistrements à une page, seul le premier formulaire ne sera accessible. En effet, comme pour l'extension de gestion de membres, des champs ont été ajoutés au formulaire "Informations sur la thèse/HDR" pour accéder à ces deux formulaires pour lier directement les enregistrements à une thèse/HDR.

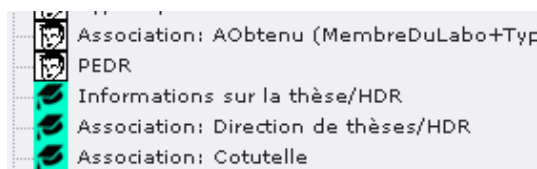


FIG. E.1 – Choix du formulaire

Voici la liste des tables associées à chaque formulaire :

Nom du formulaire	Table associée
Informations sur la thèse/HDR	tx_ligesttheses_TheseHDR
Direction de thèses/HDR	tx_ligesttheses_Dirige
Cotutelle	tx_ligesttheses_Cotutelle

TAB. E.1 – Tables associées à chaque formulaire

E.3 Insertion et configuration de plugin dans une page

E.3.1 Insertion du plugin dans une page

L'insertion du plugin dans une page se déroule de la même façon que l'extension de Gestion des membres.

E.3.2 Configuration du plugin

L'extension de Gestion des Thèses permet d'afficher toutes sortes de liste de thèses/HDR en sélectionnant des filtres ou en créant soit-même sa requête ou son gabarit. Pour cela, il faut cliquer sur l'onglet "plugin" lors de la configuration de celle-ci. Il y a alors plusieurs onglets disponibles : "Données", "Filtres", "Affichage", "Séparateurs", "Requête" et "Gabarit".

Onglet Données



FIG. E.2 – Onglet Données

L'onglet données permet de choisir un groupe de thèses/HDR à afficher en sélectionnant un dossier (en cliquant sur l'image représentant un dossier).

La sélection par dossiers comporte quelques particularités :

- Si aucun dossier n'est sélectionné, cela est considéré comme si tous les dossiers étaient sélectionnés.
- Il est possible de sélectionner plusieurs dossiers.
- Lorsqu'un dossier est sélectionné, tous ses sous-dossiers sont sélectionnés.

Onglet Filtres

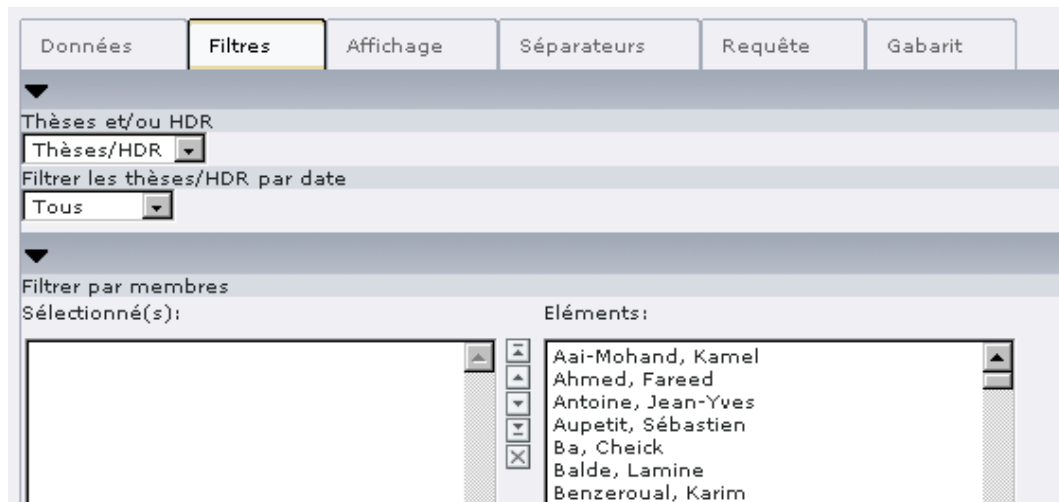


FIG. E.3 – Onglet Filtres

L'onglet filtres permet de sélectionner différentes catégories de thèses/HDR. Il est ainsi possible de sélectionner les thèses/HDR par rapport :

- à si on souhaite afficher uniquement les thèses, les HDR ou les deux. Par défaut, les deux seront sélectionnés.
- aux dates des thèses/HDR. Trois choix sont possibles : "En cours", "Terminés" ou "Tous". Par défaut, "En cours" est sélectionné.
- aux membres qui ont réalisés une thèses. Si aucun membre n'est sélectionné, cela est considéré comme si tous les membres étaient sélectionnés.
- aux membres qui ont dirigés une thèse. Si aucun membre n'est sélectionné, cela est considéré comme si tous les membres étaient sélectionnés. Il est possible d'ajouter une option sur le fait que les membres sélectionnés soient directeurs de thèses/HDR ("Oui", ne soient pas directeurs de thèses/HDR ("Non") ou que cela ne soit pas définit ("Oui ou Non"). Par défaut, l'option "Oui ou Non" est sélectionnée.
- aux équipes. Si aucune équipe n'est sélectionnée, cela est considéré comme si toutes les équipes étaient sélectionnées.
- aux résultats. Si aucun résultat n'est sélectionné, cela est considéré comme si tous les résultats étaient sélectionnés.
- aux structures. Si aucune structure n'est sélectionnée, cela est considéré comme si toutes les structures étaient sélectionnées.
- aux cotutelles. Il est possible de sélectionner uniquement les thèses/HDR possédant des cotutelles ou uniquement les thèses ne possédant pas de cotutelles ou le deux.

Onglet Affichage

Données Filtres **Affichage** Séparateurs Requête Gabarit

Format d'affichage des dates (Correspond au premier paramètre de la fonction date() de php)
d/m/Y

Texte en remplacement d'une date 0000-00-00 pour une date début de thèse/HDR
??-?-??

Texte en remplacement d'une date 0000-00-00 pour une date de fin de thèse/HDR
??-?-??

Texte en remplacement d'une date 0000-00-00 pour une date de début de direction de thèse/HDR
??-?-??

Texte en remplacement d'une date 0000-00-00 pour une date de fin de direction de thèse/HDR
??-?-??

FIG. E.4 – Onglet Affichage

L'onglet Affichage permet de configurer la façon dont vous voulez afficher certaines informations comme les dates "0000-00-00" dans le Frontend ou les libellés des résultats. Le premier champ permet de personnaliser l'affichage des dates en choisissant le format de celles-ci. La syntaxe utilisée par ce champ est la même que pour le premier paramètre de la fonction date() de php (<http://fr.php.net/date>). Par défaut, la date sera affichée au format français (d/m/Y).

Onglet Séparateurs

Données Filtres Affichage **Séparateurs** Requête Gabarit

Séparateur pour le titre de la thèse/HDR
,

Séparateur pour la description de la thèse/HDR
,

Séparateur pour la date de début de la thèse/HDR
,

Séparateur pour la date de fin de la thèse/HDR
,

Séparateur pour le jury de la thèse/HDR
,

FIG. E.5 – Onglet Séparateurs

L'onglet séparateur permet de personnaliser l'affichage des séparateurs dans le Frontend. Les séparateurs sont des marqueurs utilisés dans la construction d'un gabarit (template) qui permettent d'afficher un "séparateur" (une virgule, un point, deux points ou n'importe quel texte) uniquement si la balise contenant le texte provenant de la base de données n'est pas vide. Tous les marqueurs de séparation sont de la même forme. Par exemple pour un marqueur ###NomDuChamp### on a un marqueur ###NomDuChamp.Séparateur### associé.

Onglet Requête

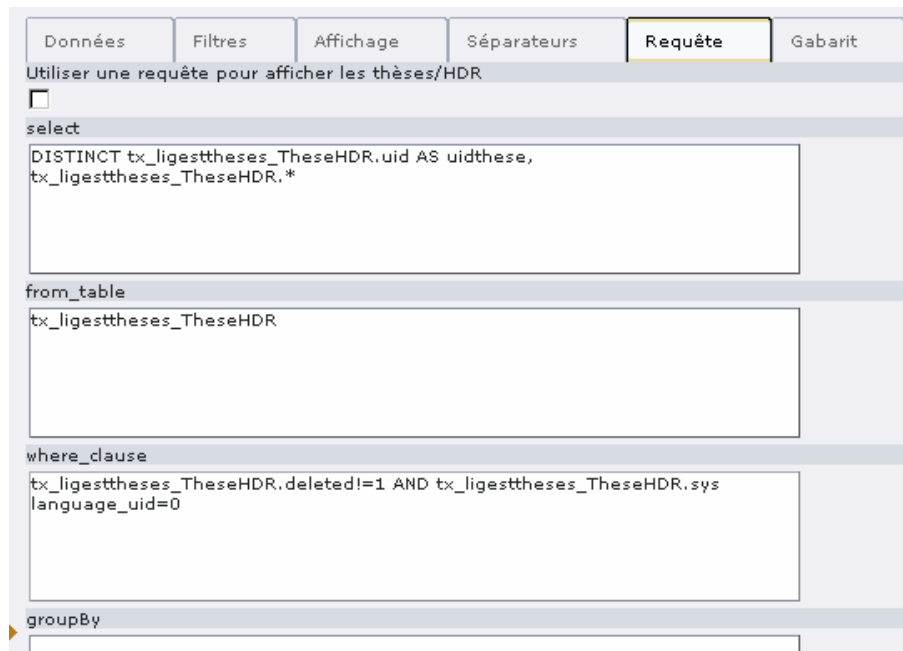


FIG. E.6 – Onglet Requête

L'Onglet Requête permet de créer sa propre requête. Pour que la requête soit effectivement utilisée, il faut tout d'abord cocher la case "Utiliser une requête pour afficher les thèses/HDR".

La construction de la requête se fait de la même manière que toute requête SQL, à l'exception près, qu'elle est séparée en plusieurs parties. Ainsi vous devez écrire la partie SELECT de votre requête dans le champs SELECT, la partie FROM dans le champs FROM,... Les mots-clés SELECT, FROM, WHERE, ORDER BY,... ne doivent pas être indiqué.

Pour que cette requête fonctionne, certains informations sont obligatoires. Vous devez absolument prendre pour requête de départ celle qui est sur l'exemple ci-dessus, à savoir :

- "DISTINCT tx_ligesttheses_TheseHDR.uid AS uidthese, tx_ligesttheses_TheseHDR.*" dans le champ SELECT.
- "tx_ligesttheses_TheseHDR" dans le champ FROM.
- "tx_ligesttheses_TheseHDR.deleted<>1 AND tx_ligesttheses_TheseHDR.sys_language_uid=0" dans le champ WHERE.

Cette requête permet d'afficher toutes les thèses/HDR qui n'ont pas été effacées. Vous pouvez ensuite rajouter vos propres conditions...

Onglet Gabarit



FIG. E.7 – Onglet Gabarit

L'Onglet gabarit permet de sélectionner le gabarit que vous souhaitez utiliser pour afficher votre page. Il est possible d'utiliser soit l'un des gabarits proposés dans l'extension, soit d'en créer un. Pour plus de détails veuillez vous référer à la partie "Création et utilisation de gabarits".

E.4 Création et utilisation de gabarits

Comme pour l'extension de Gestion des Membres, il est possible de réaliser ses propres gabarits pour personnaliser l'affichage dans le Frontend à l'aide de balises et de marqueurs.

Lors de la configuration du plugin, pour indiquer le gabarit à utiliser, il faut remplir le champs de "Référence vers le fichier du gabarit HTML" de l'onglet "Gabarit".

E.4.1 Créer son gabarit

Un gabarit est créé sur la base d'une page HTML (en-tête, corps,...). Dans le corps de cette page (body), on peut ensuite rajouter différentes balises pour obtenir un affichage personnalisé. Pour l'extension de Gestion des Thèses certaines balises sont obligatoires, il s'agit des balises TEMPLATE (début du gabarit), CONTENT (début d'un contenu) et ITEM (liste des informations à afficher) qui permettent de structurer l'affichage (Ce sont les mêmes balises de base que l'extension de Gestion des Membres).

Les balises s'utilisent toutes de la même façon :

- une balise de départ : `<!-- ###NOM### begin-->`
- une balise de fin : `<!-- ###NOM### end-->`

La structure de base d'un gabarit est :

```
<!-- ###TEMPLATE### begin-->
  <!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
    <!-- ###ITEM### end -->
  <!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->
```

Nous pouvons maintenant ajouter dans ce code différentes balises, qui permettent un affichage de plusieurs éléments (quand la base de données renvoie plusieurs résultats) ou des marqueurs, qui n'affiche qu'un seul texte.

Affichage des informations générales d'une Thèse/HDR

L'affichage des informations générales d'une thèse/HDR se fait essentiellement à l'aide de marqueurs (notés **###NomDuMarqueur###** à insérer dans la balise ITEM. Des marqueurs concernant le membre qui a réalisé la thèse/HDR ont été ajoutés. Ces marqueurs sont :

Nom du marqueur	Caractéristique
Titre	le titre de la thèse/HDR
Description	la description de la thèse/HDR
DateDebut	date de début de la thèse/HDR
DateFin	date de fin de la thèse/HDR
Jury	composition du jury de la thèse/HDR
Resultat	résultat de la thèse/HDR
Type	donne le type de thèse/HDR (T pour thèse ou H pour HDR)
NomDUsage	nom d'usage de l'auteur
NOMDUSAGE	nom d'usage de l'auteuren majuscule
NomMarital	nom marital de l'auteur
NOMMARITAL	nom marital de l'auteuren majuscule
NomPreMarital	nom prémarital de l'auteur
NOMPREMARITAL	nom prémarital de l'auteur en majuscule
Prenom	prénom de l'auteur
PRENOM	prénom de l'auteur en majuscule
InitialePrenom	première lettre du prénom de l'auteur
InitialeNom	première lettre du nom de l'auteur
Genre	genre de l'auteur (M pour masculin ou F pour féminin)
DateNaissance	date de naissance de l'auteur
Nationalite	nationalité de l'auteur
NATIONALITE	nationalité de l'auteur en majuscule
DateArrivee	date d'arrivée de l'auteur au laboratoire
DateSortie	date de sortie de l'auteur du laboratoire
NumINE	numéro INE de l'auteur
SectionCNU	numéro(s) de section CNU de l'auteur
CoordonneesRecherche	coordonnées recherche de l'auteur
CoordonneesRecherche_Ligne	coordonnées recherche de l'auteur sur une seule ligne
CoordonneesEnseignement	coordonnées enseignement de l'auteur
CoordonneesEnseignement_Ligne	coordonnées enseignement de l'auteur sur une seule ligne
CoordonneesPersonnelles	coordonnées personnelles de l'auteur
CoordonneesPersonnelles_Ligne	coordonnées personnelles de l'auteur sur une seule ligne
email	email de l'auteur
PageWeb	page web de l'auteur

TAB. E.2 – Marqueurs pour les thèses/HDR

Une balise **###PageWebLien###** permet de créer un lien autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

Par exemple pour afficher un lien autour du nom et du prénom d'un membre on écrit :

```
<!--###PageWebLien### begin -->###NOMDUSAGE### ###Prenom###<!--
###PageWebLien### end -->
```

Affichage des informations concernant les membres qui dirigent la thèse/HDR

Les informations concernant les membres qui dirigent une thèse/HDR peuvent être affichées par l'intermédiaire des balises `###DIRIGE###` et/ou `###DIRIGE_EST_DIRECTEUR###` et/ou `###DIRIGE_NESTPAS_DIRECTEUR###`.

`###DIRIGE###` permet d'afficher **tous** les membres qui dirige une thèse. `###DIRIGE_EST_DIRECTEUR###` permet d'afficher tous les membres qui dirigent une thèse **ET** qui sont directeur de la thèse. `###DIRIGE_NESTPAS_DIRECTEUR###` permet d'afficher tous les membres qui ne sont pas directeur de la thèse.

Les marqueurs sont les mêmes pour ces trois balises. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Dirige_CoefficientDEncadrement	coefficient d'encadrement exprimé en pourcentage
Dirige_EstDirecteur	indique si le membre est directeur de la thèse/HDR ou non
Dirige_DateDebut	date de début
Dirige_DateFin	date de fin
Dirige_NomDUUsage	nom d'usage du membre
Dirige_NOMDUSAGE	nom d'usage du membre en majuscule
Dirige_NomMarital	nom marital du membre
Dirige_NOMMARITAL	nom marital du membre en majuscule
Dirige_NomPreMarital	nom prémarital du membre
Dirige_NOMPREMARITAL	nom prémarital du membre en majuscule
Dirige_Prenom	prénom du membre
Dirige_PRENOM	prénom du membre en majuscule
Dirige_InitialePrenom	première lettre du prénom du membre
Dirige_InitialeNom	première lettre du nom du membre
Dirige_Genre	genre du membre (M pour masculin ou F pour féminin)
Dirige_DateNaissance	date de naissance du membre
Dirige_Nationalite	nationalité du membre
Dirige_NATIONALITE	nationalité du membre en majuscule
Dirige_DateArrivee	date d'arrivée du membre au laboratoire
Dirige_DateSortie	date de sortie du membre du laboratoire
Dirige_NumINE	numéro INE du membre
Dirige_SectionCNU	numéro(s) de section CNU du membre
Dirige_CoordonneesRecherche	coordonnées recherche du membre
Dirige_CoordonneesRecherche_Ligne	coordonnées recherche du membre sur une seule ligne
Dirige_CoordonneesEnseignement	coordonnées enseignement du membre
Dirige_CoordonneesEnseignement_Ligne	coordonnées enseignement du membre sur une seule ligne
Dirige_CoordonneesPersonnelles	coordonnées personnelles du membre
Dirige_CoordonneesPersonnelles_Ligne	coordonnées personnelles du membre sur une seule ligne
Dirige_email	email du membre
Dirige_PageWeb	page web du membre

TAB. E.3 – Marqueurs pour les dirigeants de thèses/HDR

Une balise `###Dirige_PageWebLien###` permet de créer un lien autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

Affichage des informations concernant les cotutelles

Les informations concernant les cotutelles d'une thèse/HDR peuvent être affichées par l'intermédiaire des balises `###COTUTELLE###`.

Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Cotutelle_Nom	Nom du cotuteur
Cotutelle_Prenom	Prénom du cotuteur
Cotutelle_Adresse	Adresse du cotuteur

TAB. E.4 – Marqueurs pour les cotutelles des thèses/HDR

Exemple de gabarit

Voici un exemple de gabarit permettant d'afficher une liste de thèses/HDR avec le nom et le prénom de l'auteur de la thèse/HDR, le titre de la thèse/HDR, la date de soutenance et la liste de tous les encadrants de la thèse appartenant au laboratoire.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Liste de thèses/HDR</title>
</head>
<body>

<!-- ###TEMPLATE### begin-->
<!-- ###CONTENT### begin -->
<!-- ###ITEM### begin -->

<p>
####NomDUsage#### ####Prenom####
####Titre_Separateur#### ####Titre####
####DateSoutenance_Separateur#### ####DateSoutenance####<br />
<!-- ###DIRIGE### begin -->
####Dirige.NOMDUSAGE#### ####Dirige_Prenom####<br />
<!-- ###DIRIGE### end -->
</p>

<!-- ###ITEM### end -->
<!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->

</body>
</html>
```

Utilisez votre gabarit

Une fois votre gabarit terminé. Stockez-le sur le serveur à l'aide du menu "Media". Rendez-vous ensuite dans la configuration de l'extension dans l'Onglet Gabarit et sélectionné votre gabarit à l'aide en cliquant sur l'image à droite du champ.

Installation et Utilisation de l'extension de Gestion de Publications

Cette extension permet d'afficher une liste de publications.

Pour utiliser cette extension, vous devez au préalable avoir installé l'extension de Gestion des Membres. L'utilisation de cette extension est similaire à celle de Gestion des Membres que ce soit pour l'installation, l'ajout ou la modification d'enregistrements dans la base de données ou pour l'ajout du plugin sur une page.

F.1 Installation

Pour installer cette extension, veuillez vous référer à l'installation de l'extension de Gestion des membres. En effet, son installation est totalement similaire.

F.2 Remplissage de la base de données

F.2.1 Ajout et modifications d'enregistrements

L'ajout et la modification d'enregistrements se font de la même façon que pour l'extension de Gestion des membres.

Ici les formulaires disponibles sont : "Publication", "Thème", "Fichier", "Type de Publication", "Auteur" et uniquement lorsque l'on ajoute un enregistrement à un dossier : "Association : Lier une publication à un thème", "Association : Lier une publication à une équipe" et "Association : Lier une publication avec un auteur" si l'on souhaite ajouter des enregistrements à une page, seul le premier formulaire ne sera accessible. En effet, comme pour l'extension de gestion de membres, des champs ont été ajoutés au formulaire "Publication" pour accéder à ces formulaires pour lier directement les enregistrements à une publication.

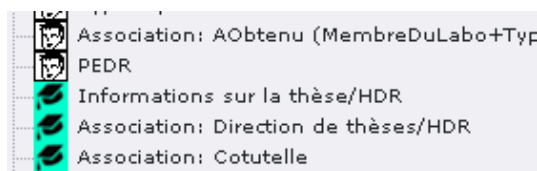


FIG. F.1 – Choix du formulaire

Voici la liste des tables associées à chaque formulaire :

Nom du formulaire	Table associée	
Publication	tx_ligestpublications_Publication	Informations sur la publication
Thème	tx_ligestpublications_Theme	Tous les thèmes de publication possibles
Lier une publication à un thème	tx_ligestpublications_Theme_Publication	Permet d'ajouter un thème à une publication
Fichier	tx_ligestpublications_Fichier	Permet d'ajouter un fichier à une publication (gestion des droits sur les fichiers)
Type de Publication	tx_ligestpublications_TypePublication	Tous les types de publications possibles
Auteur	tx_ligestpublications_Auteur	Ajouter un nom d'auteur à un membre du labo
Lier une publication avec un auteur	tx_ligestpublications_Publication_Auteur	Ajouter un auteur à une publication
Lier une publication à une équipe	tx_ligestpublications_Appartenir	Ajouter une équipe à une publication

TAB. F.1 – Tables associées à chaque formulaire

F.3 Insertion et configuration de plugin dans une page

F.3.1 Insertion du plugin dans une page

L'insertion du plugin dans une page se déroule de la même façon que l'extension de Gestion des membres.

F.3.2 Configuration du plugin

L'extension de Gestion des Publications permet d'afficher toutes sortes de liste de publications en sélectionnant des filtres ou en créant soit-même sa requête et/ou son gabarit. Pour cela, il faut cliquer sur l'onglet "plugin" lors de la configuration de celle-ci. Il y a alors plusieurs onglets disponibles : "Données", "Filtres", "Affichage", "Séparateurs", "Requête" et "Gabarit".

Onglet Données



FIG. F.2 – Onglet Données

L'onglet données permet de choisir un groupe de publications à afficher en sélectionnant un dossier (en cliquant sur l'image représentant un dossier).

La sélection par dossiers comporte quelques particularités :

- Si aucun dossier n'est sélectionné, cela est considéré comme si tous les dossiers étaient sélectionnés.
- Il est possible de sélectionner plusieurs dossiers.
- Lorsqu'un dossier est sélectionné, tous ses sous-dossiers sont sélectionnés.

Onglet Filtres

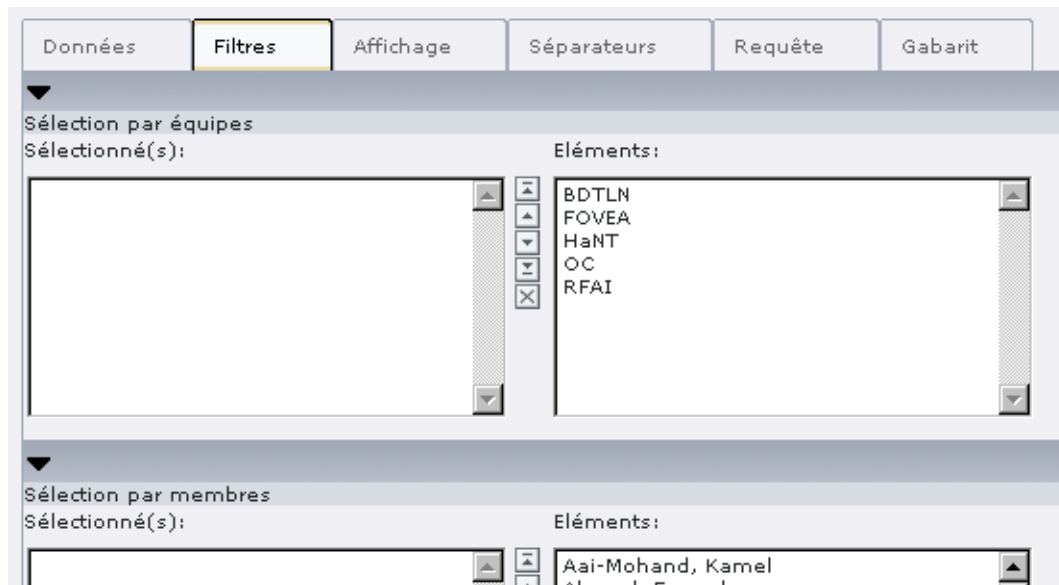


FIG. F.3 – Onglet Filtres

L'onglet filtres permet de sélectionner les publications à afficher par rapport :

- aux équipes concernées par les publications. Si aucune équipe n'est sélectionnée, cela est considéré comme si toutes les équipes étaient sélectionnées.
- aux membres concernés par les publications. Si aucun membre n'est sélectionné, cela est considéré comme si tous les membres étaient sélectionnés.
- aux auteurs concernés par les publications. Si aucun auteur n'est sélectionné, cela est considéré comme si tous les auteurs étaient sélectionnés.
- aux thèmes concernés par les publications. Si aucun thème n'est sélectionné, cela est considéré comme si tous les thèmes étaient sélectionnés.
- aux types de publications concernés par les publications. Si aucun type de publication n'est sélectionné, cela est considéré comme si tous les types de publications étaient sélectionnés.
- aux dates de publications "En cours", "Terminées" ou "Toutes". Par défaut toutes les publications sont sélectionnées.

- à une année de publication. On ne sélectionne que les publications publiées depuis cette année. L'année est au format aaaa.
- au fait que les publications soient internationales ("Est internationale") ou non ("N'est pas internationale"). Par défaut, toutes les publications sont sélectionnées ("Peu importe").
- au fait que les publications soient présentées au cours d'une conférence normale ("Conférence normale") ou en tant qu'invité ("Invité"). Par défaut, toutes les publications sont sélectionnées ("Peu importe").
- au fait que les publications soient des ouvrages ("Est un livre entier"), des chapitres ("Est un chapitre") ou ne soient pas des ouvrages ("N'est pas un ouvrage"). Par défaut, toutes les publications sont sélectionnées ("Peu importe").
- au fait que les publications soient de la vulgarisation ("Oui") ou non ("Non"). Par défaut, toutes les publications sont sélectionnées ("Peu importe").
- au fait que les publications soient parues ("Oui"), en cours de parution ("En cours de parution") ou si aucunes parutions ne sont encore prévues ("Non"). Par défaut, toutes les publications parues ou en cours de parution sont sélectionnées ("Parue ou en cours de parution"). Il est possible de sélectionner toutes les publications en choisissant l'option "Peu importe".

Onglet Affichage

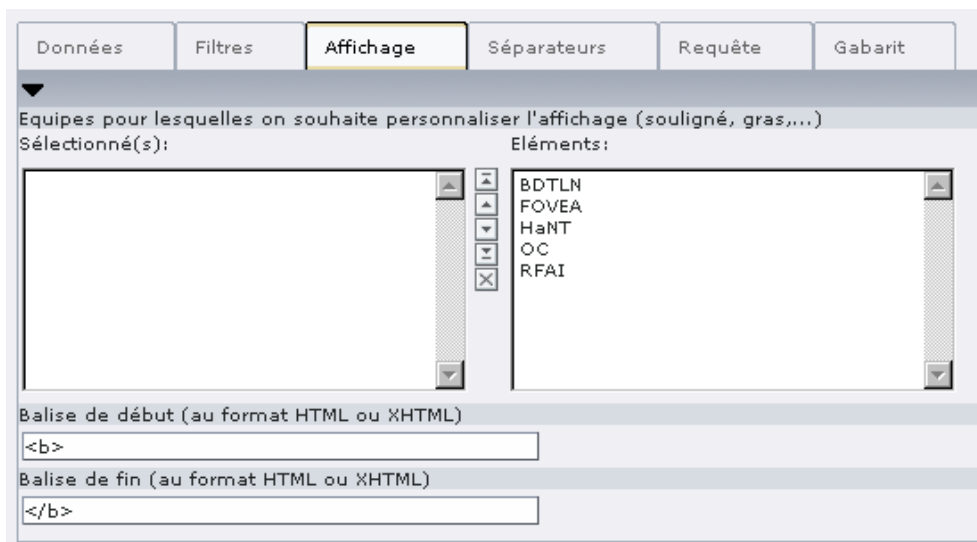


FIG. F.4 – Onglet Affichage

L'onglet Affichage permet de configurer la façon dont vous voulez afficher certaines informations comme les dates "0000-00-00" ou différents libellés dans le Frontend.

Lorsque l'on a une liste de publications, dans certains cas, il peut être utile de vouloir mettre en évidence les auteurs d'une équipe en particulier. Dans cet onglet, il est possible de sélectionner les équipes que l'on veut surligner ainsi que définir les balises que l'on souhaite utiliser pour marquer cette différence (en utilisant des balises HTML ou XHTML). Cette option est utilisée dans le gabarit à partir de la balise

"Auteurs_Surlignes".

Le champ "Format d'affichage des dates" permet, comme son nom l'indique, de personnaliser l'affichage des dates en choisissant le format de celles-ci. La syntaxe utilisée par ce champ est la même que pour le premier paramètre de la fonction `date()` de php (<http://fr.php.net/date>). Par défaut, la date sera affichée au format français (d/m/Y).

Les champs "Tri des titres" et "Tri des années" permettent de définir l'ordre de tri des publications.

Tous les autres champs permettent de personnaliser l'affichage de certains champs de la base de données.

Onglet Séparateurs

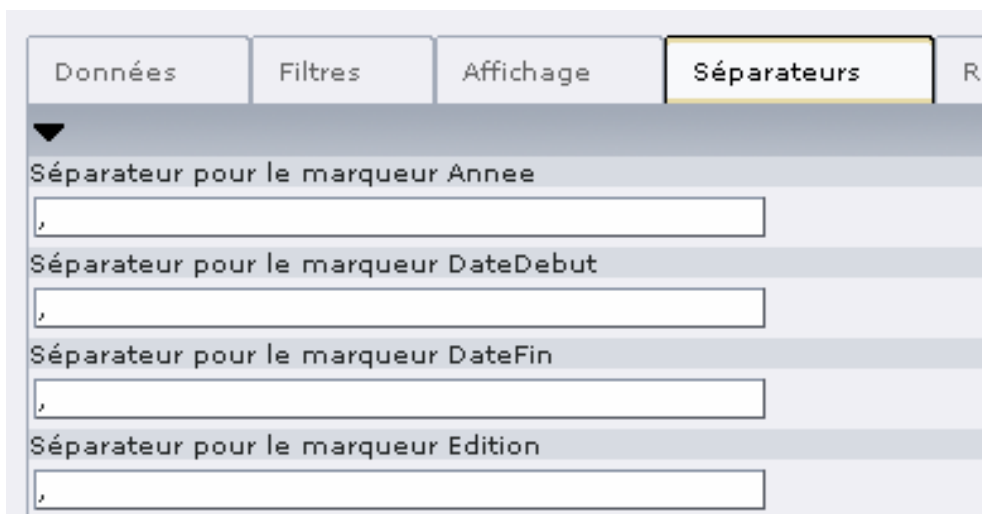


FIG. F.5 – Onglet Séparateurs

L'onglet séparateur permet de personnaliser l'affichage des séparateurs dans le Frontend. Les séparateurs sont des marqueurs utilisés dans la construction d'un gabarit (template) qui permettent d'afficher un "séparateur" (une virgule, un point, deux points ou n'importe quel texte) uniquement si la balise contenant le texte provenant de la base de données n'est pas vide. Tous les marqueurs de séparation sont de la même forme. Par exemple pour un marqueur `###NomDuChamp###` on a un marqueur `###NomDuChamp_Separateur###` associé.

Onglet Requête

Données Filtres Affichage Séparateurs **Requête** Gaba

☐ Utiliser une requête pour afficher les publications

select

```
DISTINCT tx_ligestpublications_Publication.uid AS uidpublication,
tx_ligestpublications_Publication.*
```

from_table

```
tx_ligestpublications_Publication
```

where_clause

```
tx_ligestpublications_Publication.deleted!=1
```

FIG. F.6 – Onglet Requête

L'Onglet Requête permet de créer sa propre requête. Pour que la requête soit effectivement utilisée, il faut tout d'abord cocher la case "Utiliser une requête pour afficher les publications".

La construction de la requête se fait de la même manière que toute requête SQL, à l'exception près, qu'elle est séparée en plusieurs parties. Ainsi vous devez écrire la partie SELECT de votre requête dans le champs SELECT, la partie FROM dans le champs FROM,... Les mots-clés SELECT, FROM, WHERE, ORDER BY,... ne doivent pas être indiqué.

Pour que cette requête fonctionne, certains informations sont obligatoires. Vous devez absolument prendre pour requête de départ celle qui est sur l'exemple ci-dessus, à savoir :

- "DISTINCT tx_ligestpublications_Publication.uid AS uidpublication, tx_ligestpublications_Publication.*" dans le champ SELECT.
- "tx_ligestpublications_Publication" dans le champ FROM.
- "tx_ligestpublications_Publication.deleted<>1" dans le champ WHERE.

Cette requête permet d'afficher toutes les publications qui n'ont pas été effacées. Vous pouvez ensuite rajouter vos propres conditions...

Onglet Gabarit



FIG. F.7 – Onglet Gabarit

L'Onglet gabarit permet de sélectionner le gabarit que vous souhaitez utiliser pour afficher votre page. Il est possible d'utiliser soit l'un des gabarits proposés dans l'extension, soit d'en créer un. Pour plus de détails veuillez vous référer à la partie "Création et utilisation de gabarits".

F.4 Création et utilisation de gabarits

Comme pour l'extension de Gestion des Membres, il est possible de réaliser ses propres gabarits pour personnaliser l'affichage dans le Frontend à l'aide de balises et de marqueurs.

Lors de la configuration du plugin, pour indiquer le gabarit à utiliser, il faut remplir le champs de "Référence vers le fichier du gabarit HTML" de l'onglet "Gabarit".

F.4.1 Créer son gabarit

Un gabarit est créé sur la base d'une page HTML (en-tête, corps,...). Dans le corps de cette page (body), on peut ensuite rajouter différentes balises pour obtenir un affichage personnalisé. Pour l'extension de Gestion des Thèses certaines balises sont obligatoires, il s'agit des balises TEMPLATE (début du gabarit), CONTENT (début d'un contenu) et ITEM (liste des informations à afficher) qui permettent de structurer l'affichage (Ce sont les mêmes balises de base que l'extension de Gestion des Membres).

Les balises s'utilisent toutes de la même façon :

- une balise de départ : `<!-- ###NomDeLaBalise### begin-->`
- une balise de fin : `<!-- ###NomDeLaBalise### end-->`

La structure de base d'un gabarit est :

```
<!-- ###TEMPLATE### begin-->
  <!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
    <!-- ###ITEM### end -->
  <!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->
```

Nous pouvons maintenant ajouter dans ce code différentes balises, qui permettent un affichage de plusieurs éléments (quand la base de données renvoie plusieurs résultats) ou des marqueurs, qui n'affiche qu'un seul texte.

Ordre d'affichage des publications

Cette extension a la particularité de permettre la possibilité de choisir différents tri de publications. La balise PUBLICATIONS_PAR_TITRES permet de trier les publications par titre soit dans l'ordre alphabétique, soit dans l'ordre inverse suivant le choix effectué dans l'onglet affichage des options du plugin. la balise PUBLICATIONS_PAR_ANNEES permet de trier les publications suivant leur année de publication soit dans le sens chronologique, soit dans l'ordre inverse suivant le choix effectué dans l'onglet affichage des options du plugin. Cette seconde balise possède le marqueur `###Annee###` pour afficher les différentes années de publications sous forme de titre. Lorsque plusieurs publications ont été publiées la même année, ce titre "Année" n'est affiché qu'une seule fois. Pour ces deux balises, on utilise ensuite la sous-balise `###PUBLICATIONS###` et ses marqueurs associés pour afficher les informations sur les publications.

Voici la structure du template avec la balise PUBLICATIONS_PAR_TITRES :

```
<!-- ###TEMPLATE### begin-->
<!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
        <!-- ###PUBLICATIONS_PAR_TITRES### begin -->
            <!-- ###PUBLICATIONS### begin -->
                ###Titre### ###Titre_Separateur###<br />
            <!-- ###PUBLICATIONS### end -->
        <!-- ###PUBLICATIONS_PAR_TITRES### end-->
    <!-- ###ITEM### end -->
<!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->
```

Voici la structure du template avec la balise PUBLICATIONS_PAR_ANNEES :

```
<!-- ###TEMPLATE### begin-->
<!-- ###CONTENT### begin -->
    <!-- ###ITEM### begin -->
        <!-- ###PUBLICATIONS_PAR_ANNEES### begin -->
            <h1>###Annee###</h1>
            <!-- ###PUBLICATIONS### begin -->
                ###Titre### ###Titre_Separateur###<br />
            <!-- ###PUBLICATIONS### end -->
        <!-- ###PUBLICATIONS_PAR_ANNEES### end-->
    <!-- ###ITEM### end -->
<!-- ###CONTENT### end -->
<!-- ###TEMPLATE### end-->
```

Affichage des informations générales d'une Publication

L'affichage des informations générales d'une publications se fait essentiellement à l'aide de marqueurs (notés `###NomDuMarqueur###` à insérer dans la balise PUBLICATIONS. Ces marqueurs sont :

Nom du marqueur	Caractéristique
Titre	le titre de la publication
Annee	l'année d'édition de la publication
DateDebut	date de début de la publication
DateFin	date de fin de la publication
EstEditeur	éditeur ou non ?
EstDeLaVulgarisation	la publication est de la vulgarisation ou une publication scientifique ?
EstInternationale	publication internationale ou nationale ?
EstInvite	conférence normale ou invité de la conférence ?
EstParu	la publication est paru ou est en cours de parution ?
TypeContribution	indique si la contribution est partielle ou totale.
TypeDiffusion	indique le type de diffusion Large (ISBN/Impact Factor) ou Restreinte (pas ISBN/pas Impact Factor).
TauxAcceptation	Le taux d'acceptation.
ISBN	numéro ISBN-13 de la publication.
DOI	DOI de la publication.
TitreSupport	Titre du support de la publication.
MediaDePublication	média utilisé pour la publication (nom du journal, nom de la conférence,...)
Notes	Notes sur la publication.
Numero	numéro de la revue.
Pages	nombre de pages ou plage de pages de la publication.
PublisherOrSchool	L'organisation éditrice telle que Springer, Hermes,... ou l'école telle que Université de Tours.
Serie	le numéro de la série.
Edition	L'édition (ex : seconde édition).
VilleEtPays	La ville et le pays où s'est déroulé l'événement.
Volume	Le numero du volume.
LienDur	Lien vers une page interne ou externe au site.
TypePublication_Libelle	Libellé du type de publication
TypePublication_Code	Code du type de publication

TAB. F.2 – Marqueurs pour les publications

Une balise `###Lien###` permet de créer un lien vers la page interne ou externe indiqué pour la publication autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

Par exemple pour afficher un lien autour du titre de la publication on écrit :

```
<!--###Lien### begin -->###Titre### <!--###Lien### end -->
```

Affichage des informations concernant les auteurs de la publication

Les informations concernant les auteurs de la publication peuvent être affichées par l'intermédiaire de la balise `###AUTEURS###`. Les auteurs sont triés selon le champ ordre indiqué lors de l'ajout d'un

auteur pour une publication (dans l'ordre croissant). Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Auteurs_Nom	Nom de l'auteur
Auteurs_NOM	Nom de l'auteur en majuscule
Auteurs_Prenom	Prénom de l'auteur
Auteurs_PRENOM	Prénom de l'auteur en majuscule
Auteurs_InitialeNom	Première lettre du nom de l'auteur
Auteurs_InitialePrenom	Première lettre du prénom de l'auteur
Auteurs_PageWeb	Page web de l'auteur (page web du membre lié à l'auteur)

TAB. F.3 – Marqueurs pour les auteurs de la publication

La balise `###Auteurs_PageWebLien###` permet de créer un lien vers la page web de l'auteur autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

La balise `###Auteurs_Surlignes###` permet de surligner le texte que l'on veut à partir des balises définie dans l'option du plugin et des équipes sélectionnées. Ainsi les balises ne seront ajoutées que pour les auteurs appartenant aux équipes sélectionnées.

Affichage des informations concernant les fichiers de la publication

Les informations concernant les fichiers de la publication peuvent être affichées par l'intermédiaire de la balise `###FICHIERS###`. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Fichiers_Nom	Nom du fichier
Fichiers_LienDur	Lien vers le fichier

TAB. F.4 – Marqueurs pour les fichiers de la publication

La balise `###Fichiers_Lien###` permet de créer un lien vers la fichier autour du texte que l'on veut. Ce texte peut être formé de marqueur ou non.

Les informations concernant les fichiers sont affichées uniquement pour les utilisateurs possédant les droits nécessaires à leur affichage.

Affichage des informations concernant les équipes liées à la publication

Les informations concernant les équipes liées à la publication peuvent être affichées par l'intermédiaire de la balise `###EQUIPES###`. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Equipes_Nom	Nom de l'équipe
Equipes_Abbreviation	Abréviation de l'équipe

TAB. F.5 – Marqueurs pour les équipes liées à la publication

La balise `###Equipes_Surlignes###` permet de surligner le texte que l'on veut à partir des balises définie dans l'option du plugin et des équipes sélectionnées. Ainsi les balises ne seront ajoutées que pour les équipes sélectionnées.

Affichage des informations concernant les thèmes de la publication

Les informations concernant les thèmes de la publication peuvent être affichées par l'intermédiaire de la balise `###THEMES###`. Les marqueurs disponibles sont :

Nom du marqueur	Caractéristique
Themes_Libelle	Libellé du thème

TAB. F.6 – Marqueurs pour les thèmes de la publication

Utilisez votre gabarit

Une fois votre gabarit terminé. Stockez-le sur le serveur à l'aide du menu "Media". Rendez-vous ensuite dans la configuration de l'extension dans l'Onglet Gabarit et sélectionné votre gabarit à l'aide en cliquant sur l'image à droite du champ.

Applicatifs métiers basés sur le framework Typo3 pour la gestion du Laboratoire d'Informatique

Département Informatique
5^{eme} année
2008-2009

Rapport de projet de fin d'études

Résumé: L'objectif de ce projet de fin d'études était de réaliser plusieurs extensions pour le framework Typo3, un système de gestion de contenu permettant de créer des sites web, afin d'automatiser la mise à jour de certaines pages du Laboratoire d'Informatique présent à Polytech'Tours. Ce rapport présente le déroulement du projet, la création d'une extension pour le CMS Typo3 ainsi que l'utilisation des trois extensions que j'ai réalisé. Ces trois extensions permettent d'afficher respectivement des listes de membres, de thèses/HDR et de publications à partir d'une base de données, créée lors de l'installation et de paramètres définies au cours de la configuration du plugin.

Mots clefs: Typo3, extensions, Membre, Thèses/HDR, Publications, base de données, formulaires

Abstract : The purpose of this project was to realize several extensions for the Typo3 framework, a free Open Source content management system allowing to create Web site, to automate the update of certain pages of the Laboratory of Computing to Polytech'Tours. This report presents the progress of the project, the creation of an extension for CMS TYPO3, the use of three extensions that I realized. These three extensions allow to show respectively lists of members, thesis/HDR and publications from a database, created during the installation and the parameters defined during the configuration of the plugin.

Keywords : Typo3, extension, members, thesis, publications, database, form

Étudiant :

Bruno GALLET

bruno.gallet@etu.univ-tours.fr

Encadrants :

Sébastien AUPETIT

sebastien.auvetit@univ-tours.fr

Jean-Charles BILLAUT

jean.billaut@univ-tours.fr

Université François-Rabelais, Tours