
WE4A

Cahier des Charges du Site

« Moodle Simplifié »

Le but de ce document est de préciser les demandes du sujet, de la perspective de WE4A. Même si le Moodle original reste un excellent référentiel pour ce que nous vous demandons, les différences sont aussi considérables, et pour répondre à votre demande pour une clarification, ce document a été produit.

Ce cahier des charges » a été rédigé de la perspective de WE4A. Il ne va donc vous décrire que les features demandées au niveau du site, sans se préoccuper de la qualité de la BDD qui le fournit en données.

Sommaire

- 1) Technologies à Utiliser.....3
- 2) Rôles devant exister sur le Site.....4
- 3) Liste des pages principales nécessaires
 - A) Page de Login
 - B) Page d'Administration
 - C) Page de « choix de l'UE »
 - D) Page de Contenu pour une UE
 - E) Page de création de Post
 - F) Page de notation des Devoirs
 - G) Liste des inscrits à une UE
 - H) Page de gestion de Compte
- 4) Types de Posts pouvant être créés
- 5) Autres Nice-to-Have

1 - Technologies à utiliser

Le site doit **absolument** utiliser le framework **Symfony 7**.

Pour que ce dernier fonctionne, il vous faut des serveurs Apache et MySQL, et la manière la plus simple de les avoir sur sa machine est d'installer XAMPP ou un équivalent.

Lors des corrections, le dossier de votre site sera placé dans une machine avec XAMPP et Symfony. Les « migrations » de BDD seront faites à partir de vos entités Symfony, des données exemples importées à partir d'un fichier SQL généré via PHPmyAdmin, et le site sera alors censé **fonctionner immédiatement sans avoir besoin d'installer autre chose**.

ATTENTION : AUCUNE AUTRE LIBRAIRIE, PACKAGE OU PLUGIN N'EST AUTORISÉ (à l'exception des 2 mentionnés si dessous). Si vous désobéissez à cette règle, votre note sera pénalisée.

En effet, si les étudiants ajoutent des librairies et technos de leurs choix, la correction de ces projets, déjà très longue et complexe, devient juste ingérable de la perspective des enseignants, car cela va **interférer avec nos barèmes** et critères de notation.

Je mentionnais des exceptions. Ce sont deux outils/librairies communs dans le web et qui ont l'avantage de pouvoir être importés par un lien global et/ou placées dans le dossier du site.

Nous n'aurons pas le temps de les étudier en TD/TP, mais vous avez le droit d'utiliser si vous les connaissez :

- **Bootstrap.** Pour aider à concevoir le HTML/visuel de la page
- **JQuery,** pour aider sur les Javascript côté client

J'insiste que ce sont bien ces deux là et seulement ces deux là. Il existe des dizaines d'alternatives moins connues à Bootstrap et JQuery, qui, elles aussi, peuvent être importées par un lien et jouent des rôles similaires.

Mais le critère pour refuser les autres librairies n'est pas d'éventuels problèmes d'installation, mais bien le BARÈME, qui ne peut pas tenir en compte tous les cas possibles de librairies et d'outils, et doit donc vous restreindre sur les librairies facultatives que vous avez le droit d'utiliser.

Mais avec Bootstrap et JQuery, on peut quasi littéralement tout faire, alors...

2 – Rôles devant exister sur le Site :

Administrateur :

- **Ce rôle permet, après login, d'accéder à une zone spécifique du back-end, réservée aux seuls administrateurs.**
Dans ce back-end « admin », il est possible de créer les UE, et d'y assigner un ou plusieurs professeurs et étudiants.
- **L'administrateur a aussi le pouvoir de créer des comptes utilisateur** avec un mot de passe temporaire/par défaut, et de leur assigner les rôles de son choix (étudiant, professeur, ou même admin)
- **Permettre à n'importe qui de créer son compte via formulaire ne fait pas vraiment sens sur le site d'une université/école.**
C'est l'administration qui s'en occupe. Il faudra juste créer « artificiellement » un compte admin dans la BDD afin de démarrer la chaîne de création de comptes.
- **Il a des professeurs qui sont aussi administrateurs**, mais il y a des administrateurs qui ont seulement ce rôle et ne sont pas professeurs (secrétariat, service de gestion de emplois du temps, etc.)
- **Si un prof qui est aussi admin se login**, il peut, au choix, arriver sur la zone d'administration, ou bien sur la page d'accueil normale des professeurs. Ce qui compte, c'est qu'en temps qu'admin, il aie accès à des liens pour passer facilement de la « zone admin » à la « zone profs »
- **Un étudiant ne peut pas être administrateur**
- Dans le cadre des « objectifs minimum », un administrateur **n'a pas la possibilité d'accéder à l'intérieur des UE** pour modifier leurs posts. Sauf si il s'avère qu'il est aussi professeur sur cette UV.
- Un « Nice to Have » (voir fin de document) serait que les admins puissent modifier le contenu des UE, mais que les professeurs d'une UE modifiée reçoivent une notification qu'une action admin a eu lieu.

Etudiant :

- **Après son login**, un étudiant arrive sur la page de « choix de l'UV ». Cette page contient aussi des informations sur l'activité récente sur les UE en question (voir plus loin)
- **En cliquant une UE, un élève va pouvoir voir les posts existants dans l'UE en question.** L'étudiant, ne peut, pour l'essentiel, que consulter : il peut voir les messages textes, télécharger les fichiers déposés par les professeurs, etc...
- **L'élève ne peut interagir qu'avec les posts « devoir »,** pour lesquels il peut uploader un fichier ZIP contenant son travail. Lorsque le professeur a noté ce devoir, l'étudiant peut voir sa note.
- **Le cas du devoir est décrit plus en détail plus loin**, vous donnant des suggestions sur comment l'implémenter.

Professeur :

- **Ce rôle permet, après login, d'accéder aux mêmes pages que les étudiants.** ...Mais avec des liens/formulaires supplémentaires qui permettront **l'édition du contenu.**
- **Après son login**, un professeur (comme un étudiant) arrive sur la page de « choix de l'UV ». Cette page contient aussi des informations sur l'activité récente (voir plus loin)
- En cliquant une UE, un professeur va pouvoir voir les posts existants dans l'UE en question. Comme les étudiants. Mais lui pourra voir **des boutons supplémentaires afin d'ajouter/modifier/effacer les posts.**
- **Selon votre ambition/niveau/temps disponible**, les formulaires d'édition de contenu pourront être ou bien sur une page séparée, ou bien révélés/ajoutés dynamiquement à la page via Javascript et AJAX.
Conseil : commencez par une version où les formulaires sont sur des pages différentes (objectif minimum) et essayez plus tard de les transformer en versions plus dynamiques et ergonomiques
- **Pour les posts « devoir » sur une de ses UE**, un professeur a accès à un lien qui va ouvrir une page différente, où il peut voir tous les fichiers uploadés. Et noter ces travaux sur 20.
Le devoir a sa propre section avec des suggestions sur comment aborder son implémentation en termes de site web (voir plus loin)

3 – Listes des Pages principales nécessaires

Vous aurez probablement besoin de pages supplémentaires pour accueillir, par exemple, le formulaire de création d'un nouveau post.

Mais dans la théorie, les pages ci-après sont les seules à être strictement nécessaires. ...Pour peu que vous utilisiez beaucoup, beaucoup de Javascript afin de mettre la page à jour « en direct ».

Ces Javascript destinés à éviter les pages supplémentaires sont des « Nice to Have ». Ils ne sont pas obligatoires.

Mais il y a une certaine quantité de Javascript obligatoire, qui vous sera demandé explicitement dans la description des pages ci-après. Ils seront présentés ainsi, dans un encadré bleu clair facile à repérer.

A) Page de Login (*tout le monde*)

La page d'accueil consiste essentiellement en un formulaire de login.

Elle est la seule page qui peut être accédée sans login, et si un utilisateur tente d'accéder une autre page sans s'être identifié, il sera **redirigé** ici.

Javascript demandé : le formulaire de login doit proposer l'option de montrer/cacher le mot de passe. Il doit aussi refuser d'envoyer les données si jamais un champ a été laissé vide.

Nice to Have : Vous pouvez, si vous le souhaitez, rendre cette page un peu plus intéressante, par exemple avec une esthétique plus poussée ou en affichant des statistiques, par exemple sur le nombre de professeurs, d'élèves, de posts, etc, etc...

B) Page(s) d'administration (*admin seulement*)

Accessible uniquement aux administrateurs, cette page est même celle où ce type d'utilisateur va se retrouver immédiatement après son login.

Un professeur qui est aussi un admin n'arrive pas ici, mais sur la page de sélection de l'UE après son login (voir plus loin).

Un « prof admin » a en revanche en permanence accès à un lien qui l'amène sur cette page d'administration (*par exemple dans une barre de menu en haut*)

Javascripts demandés

UE et users sont sur deux « onglets » de la même page, avec la possibilité d'afficher un onglet ou l'autre via Javascript.

L'effacement des UE et étudiants doit se produire via AJAX, sans changer de page. Une pop-up demandant confirmation doit également être affichée avant que l'effacement via AJAX et la mise à jour de la liste aie lieu.

Nice To Have

Lorsque l'on veut ajouter ou modifier une UE ou un User, un usage intensif et intelligent de Javascript et AJAX permet de « faire apparaître les formulaires » dans la page, et d'envoyer les modifications et mettre à jour la page, le tout sans « aller sur une page de formulaire séparé ».

Si vous implémentez ça, le site sera plus dynamique à utiliser, et c'est un +

C) Page de choix de l'UE (*prof, étudiant*)

D) Page de contenu pour une UE (*prof, étudiant*)

E) Page de création de Post (*prof*)

F) Page de notation des Devoirs (*prof*)

G) Liste des inscrits à une UE (*prof, étudiant*)

H) Page de gestion de Compte (*tout le monde*)

4 – Types de Posts pouvant être créés

Un utilisateur « professeur » peut ajouter du contenu sur les pages d’UV.
Votre « Moodle simplifié » doit proposer, au minimum, ces trois types de posts.

A) Les messages Texte

Le type de poste le moins complexe. **Il est simplement constitué d’un titre, une date+heure de post, et d’un texte** (potentiellement long, plusieurs paragraphes).
A l’affichage, ces informations ressortent simplement pour afficher un texte avec un titre.

Un étudiant ne peut pas interagir avec le message texte, autrement qu’en le lisant.
De sa perspective, il n’est pas interactif.

Un professeur, en revanche, a accès à des boutons « éditer » et « supprimer » sur chaque « post message » d’une UE dont il est professeur (*même si un autre professeur, assigné à la même UE, est l’auteur du message*).

Cliquer le bouton éditer ouvre un formulaire d’édition. Comme déjà mentionné, ce formulaire peut se trouver sur une autre page (*et c’est conseillé de commencer ainsi*)

Le processus de création de posts est décrit plus haut, dans la « liste de pages indispensables ». **N’oubliez pas que chaque post, peu importe son type, va générer aussi un affichage sur la liste « activité récente » visible sur la page de choix d’une UE.**

Javascrpts demandés

Pour un professeur, la suppression d’un message doit pouvoir avoir lieu sans changer de page : Une pop-up apparaît pour demander confirmation, et si on valide, l’effacement et la mise à jour de la page ont lieu par AJAX.

Cette fonctionnalité est aussi demandée pour les autres types de posts.

Nice To Have

Si vous avez le temps/les moyens, vous pouvez essayer de concevoir un système Javascript/AJAX permettant d’éditer les posts sans quitter la page.

B) Les dépôts de Fichier(s)

Un post « dépôt de fichier » est assez similaire à un message texte : il contient un titre, un texte descriptif, une date et heure de post, et, bien sûr, un fichier uploadé par le professeur.

Le seul type de fichier « valide » absolument exigé/nécessaire est ZIP.

A l’affichage, les dépôts de fichiers s’affiche de façon similaire à un message texte, mais inclut en plus **un lien (très proéminent) qui, quand il est cliqué, lance le téléchargement du fichier associé.**

Un étudiant a une interaction assez limité avec un « post fichier », mais cette « petite interaction » est critique : il peut en voir le titre, lire le texte descriptif, et **cliquer le lien afin de télécharger le fichier.**

Un professeur, en revanche, a accès à des boutons « éditer » et « supprimer » sur chaque « post de fichier » d’une UE dont il est professeur (*même si un autre professeur, assigné à la même UE, est l’auteur du message*).

Cliquer le bouton éditer ouvre un formulaire d’édition. Comme pour les posts de messages textes, le formulaire peut être sur une autre page. Ce formulaire propose les mêmes options que pour le message texte, avec en plus l’option de remplacer le fichier associé, si on le désire.

Le processus de création de posts est décrit plus haut, dans la « liste de pages indispensables ».

Javascripts demandés

Pour un professeur, la suppression d’un message doit pouvoir avoir lieu sans changer de page : Une pop-up apparaît pour demander confirmation, et si on valide, l’effacement et la mise à jour de la page ont lieu par AJAX.

(Même demande que pour les « posts message »)

Nice To Have

Idem que les « post messages » : une édition « directe sur la page » comme dans le « vrai » Moodle, serait un plus.

Vous pouvez aussi introduire une gestion de différents types de fichiers en plus du ZIP, et éventuellement faire que cela résulte en des différences visuelles sur le post (icône de type sur le lien, etc...)

C) Les Devoirs

Le devoir est le troisième type de post, et est de très loin le plus complexe. J'ai donc décomposé la description de son comportement en plusieurs parties.

C1) Création

Un devoir, lors de sa création par un professeur, a beaucoup de points communs avec un dépôt de fichier : il a aussi un titre, un texte descriptif, une date et heure de post, et la possibilité d'uploader un fichier ZIP contenant les sujet/des documents de référence. *(Vous pouvez littéralement réutiliser le code)*

Un devoir offre néanmoins des options supplémentaires : il est possible de saisir une date limite, ainsi que le coefficient de l'examen *(un nombre fourni à titre indicatif, sauf si vous allez très loin dans les « nice to have » et proposez un calcul automatique de la moyenne des étudiants !)*

C2) Mise en avant des devoirs « non terminés »

Les devoirs impliquant des nécessités de réaction de la part des utilisateurs (les élèves doivent rendre leur travail à temps, les profs doivent noter), il convient de **les mettre en avant par rapport au autres posts**.

Sur la page d'une UE, les devoirs sont donc affichés en priorité, en haut de la page. Si il y a plusieurs devoirs, ils sont tous « épinglés » en haut de la page, et rangés en ordre chronologique du plus récent au plus ancien (comme les autres posts)

La « notification d'activité » visible dans les actualités (sur la page de choix de l'UE) va elle aussi être placée en haut de la liste. Si il y a plusieurs devoirs dans une UE ou dans les différentes UE combinées de l'utilisateur, la même logique que sur une page individuelle d'UE est appliquée : tous les devoirs sont en haut, et rangés par ordre chronologique entre eux.

Comme vous le verrez plus loin, il est possible de « désépingler » ces notifications et posts particuliers, afin qu'ils ne soient plus « placés en haut ».

Mais pour que cette option apparaisse, il faut que le travail lié à ce devoir soit terminé. Ce qui signifie :

- **Etudiant** : avoir uploadé son travail, et reçu sa note
- **Professeur** : avoir noté tous les étudiants ayant remis un travail et avoir « validé » l'examen *(ce qui n'est possible qu'après la date limite de rendu)*

C3) Sur les pages d'UE

Au delà du problème de « l'épinglage », décrit précédemment, voilà comment le post doit apparaître sur la page de son UE correspondante.

Globalement, son apparence est similaire à celle d'un dépôt de fichier. **Mais la date limite de remise du travail, ainsi que le coefficient, sont ajoutés avant le texte descriptif.** Faites en sorte que la date limite soit bien proéminente !

Pour un étudiant : Si le travail n'a pas encore été rendu, le post inclut un petit formulaire permettant d'**uploader un fichier ZIP avec son travail**.

Ce formulaire doit demander confirmation avant l'envoi, car, dans le contexte du «minimum » que nous vous demandons, la remise du travail est définitive.

Si un travail a été rendu, le formulaire n'est plus là, et est remplacé par un message « vous avez remis votre travail le XX/XX/20XX à XXhXX ».

Le post reste néanmoins « épinglé », afin que l'étudiant y accède facilement quand il aura été noté.

Si la date limite de rendu du devoir a été atteinte, mais que l'étudiant n'a rien rendu, le formulaire n'est pas montré, et est remplacé par un message de type « La date limite de rendu a été dépassé. Si vous avez une excuse valide, veuillez contacter un de vos professeurs par email ».

Nice To Have

Donnez l'option aux étudiants de modifier leur travail rendu, jusqu'à la date limite de l'examen. Faites évoluer l'interface graphique du site pour que l'utilisateur comprenne bien cette nouvelle possibilité.

Aussi, comme tous les autres formulaires, améliorer avec du Javascript/AJAX pour éviter les changements de page est un plus

Pour un professeur : Un message est ajouté au post : « X élèves sur Y ont déjà remis leur travail », généré dynamiquement.

Et surtout, un **bouton supplémentaire** est disponible pour le professeur : un qui lui permet d'accéder à la « page de notation ».

En raison du nombre d'élèves assez élevé dans une UE typique, **utiliser un formulaire intégré et du AJAX ne fait pas vraiment sens pour les notations.**

Si vous avez du temps à consacrer aux « Nice-to-have » qui consistent à éliminer les « formulaires séparés sur une autre page », faites le ailleurs ! XD

C4) Page de Notation

La page de notation d'un devoir est une page **exclusivement accessible aux professeurs assignés à l'UE dont ce devoir fait partie**. Si un autre utilisateur tente d'y accéder, il est redirigé sur sa « page d'accueil ».

Sur la page de notation, le professeur peut voir une liste de tous les étudiants assignés à l'UE. **Pour chaque étudiant, on peut voir si un « fichier de travail réalisé » a été déposé/uploadé ou non.**

Et si oui, télécharger ce fichier en cliquant sur un lien/bouton.

Nice To Have

Sauvegardez la date et l'heure de dépôt des fichiers, et affichez là ici, et sur le post du devoir, quand il est vu de la perspective de l'étudiant qui a déjà uploadé

Chaque étudiant sur cette page dispose d'un **champ de texte destiné à recevoir sa note**. Le professeur peut y saisir seulement des nombres.

Javascripts demandés

Lorsque le « focus » est retiré d'un de ces champs de note, un petit javascript doit s'exécuter pour vérifier qu'un nombre entre 0 et 20 a été saisi, et le sauvegarder automatiquement dans la BDD comme « la note pour cet étudiant »

Attention : un professeur ayant l'option de modifier ses notes en cas d'erreur, l'étudiant ne doit pas être notifié immédiatement de sa note.

Pour gérer cette « possibilité d'ajuster les notes plus tard », **les notes d'un examen doivent être validées.**

En bas de la « page de notation » se trouve un **bouton « valider les notes de ce devoir définitivement »**. Lorsque ce bouton est cliqué, le devoir est « validé » et les notes ne peuvent plus être changées.

Les étudiants pourront alors voir leur notes de leur côté de l'appli.

Javascripts demandés

Le bouton « valider les notes définitivement » doit demander confirmation avant de vraiment valider l'examen. Car, dans le contexte de ce Moodle simplifié, cette opération est irréversible.

Si un examen est validé alors que des étudiants n'ont aucune note indiquée, ces étudiants recevront automatiquement un zéro.

Nice To Have

Une amélioration « nice to have » possible est de permettre de personnaliser cette note « attribuée à tous les absents » et/ou d'ajouter la possibilité de placer un étudiant en statut « absence justifiée », dans lequel ils n'auront pas de note pour ce devoir.

C5) Affichage pour l'étudiant quand il a été noté

Lorsqu'une note a été attribuée à un étudiant, cet étudiant est notifié de plusieurs manières :

- **Dans « l'activité récente » visible sur la page de choix de l'UE, une notification est ajoutée pour l'étudiant**, avec un texte de type « vous avez reçu la note de X/20 au devoir (titre du devoir) »
Ces notifications apparaissent toujours en haut de la liste d'activité.
- **Sur la page de contenu de l'UE**, le post du devoir correspondant est modifié pour afficher la note. Un bouton « retirer l'examen du haut de page » (ou « j'ai compris » si vous voulez être moins formel) est alors ajouté aussi, donnant l'option à l'étudiant de « désépingler » le devoir du haut de la page.
Un devoir « désépinglé » sera affiché, à l'avenir, à sa position chronologique, comme les reste des posts.
- **Notez que pour le professeur, l'option de dés-épingler les devoirs terminés peut exister aussi.** Mais pour eux, il est nécessaire d'avoir validé » un devoir avant de pouvoir le faire.
Mais cela fait aussi sens qu'un devoir « validé » soit automatiquement « dés-épinglé »

Nice To Have

Vous pouvez améliorer le système pour gérer des cas comme les étudiants absents, les devoirs rendus en retard, etc, etc.

Le système « d'épingler des posts en haut de la page » pourrait également être généralisé, permettant à un professeur de remonter vers le haut de la page des posts de son choix, et de les « redescendre » quand il le souhaite (voir plus loin)

5 – Autres « Nice To Have »

Vous avez pu constater que ce document est rempli d'encadrés « Nice to Have », suggérant l'implémentation de fonctionnalités facultatives.

Les « Nice to have » font, en réalité, partie intégrale de la note, avec 4 points qui leur sont consacrés. Pour être vraiment « noté sur 20 », vous êtes donc censés en implémenter quelques uns !

Le but de ce système est de permettre aux étudiants de « personnaliser » un peu ce sur quoi ils travaillent, et de capitaliser sur leurs points forts.

Javascript vous parle ? Alors choisissez des Nice to have qui utilisent cet outil. Vous, c'est plutôt le PHP et la BDD ? Alors ajoutez des possibilités telles que la possibilité de remplacer une note par une « absence justifiée », etc, etc.

Pour ceux qui sont particulièrement à l'aise avec le web et veulent un challenge, voilà 3 « nice to have » particulièrement ambitieux. Un seul, bien réalisé, vous rapportera déjà une bonne fraction des 4 points. Le dernier, en particulier, vous les rapportera tous.

Ayez bien conscience que ces nice to have sont bien plus complexes que les autres suggestions du document. Et que vous serez moins guidés sur la manière de les accomplir aussi.

Il est plus « rentable » de se focaliser d'abord sur ce qui est exigé, et de voir ENSUITE si vous avez le temps de tenter un de ces objectifs.

1) Possibilité d'épingler/désépingler n'importe quel post

- Lorsqu'un professeur crée/modifie un post qui n'est pas un devoir, il aurait l'option de « l'épingler ».
- Un post épinglé apparaît en haut de la page. Comme un « devoir » le fait dans ce qui est exigé. Dans ce contexte, on peut considérer qu'un devoir est « épinglé automatiquement ».
- Les devoirs en attente sont affichés au dessus des autres posts épinglés. On a donc trois groupes, dans l'ordre : les devoirs en attente, les posts épinglés, et les autres posts.
Chacun des trois groupes est, de manière interne, rangé en ordre chronologique décroissant (du plus récent au plus ancien)
- La BDD doit pouvoir stocker si un post est « pinned », et pour quel utilisateur
- Le plus simple de ces trois « gros nice to have », et de loin. Il vaut en conséquence, moins de points.

2) Les Admins peuvent jouer un rôle de « modérateur »

- Dans la fonctionnalité de base, un administrateur ne peut pas rentrer dans les UE, sauf si il est aussi un professeur sur cette UE.
- Ce « Nice to Have » transformerait les admins en modérateurs : ils peuvent rentrer dans n'importe quelle UE, et ajouter/modifier/effacer des posts comme le ferait un professeur.
- Il convient, en revanche, que les professeurs soient bien notifiés qu'une action admin a eu lieu sur une de leurs UE : des alertes doivent donc être générées et affichées dans « l'activité récente » sur la page de choix d'UE.
- A la façon des devoirs non terminés, ces « alertes » doivent être priorisées en haut de la liste d'activité, et mises en valeur (couleur, image de panneau « ! »...) pour attirer l'attention
- L'alerte doit inclure un lien qui amène sur la page de l'UE concernée, et, si possible, place le focus immédiatement sur le post modifié.
- Un professeur peut retirer une alerte en cliquant sur un bouton « J'ai compris ». Elle ne sera alors plus affichée.
- Les « alertes action admin » ont besoin d'avoir une existence dans la BDD
- Ce Nice-to-Have est complexe, mais inclut des similitudes avec des choses exigées, et réutilise certains systèmes (alertes de devoir en attente, entrer dans les UE pour pouvoir les modifier, etc.), il n'est donc pas aussi ambitieux que le suivant.

3) Donner aux profs le contrôle de la mise en page des UE

- L'objectif le plus ambitieux que je vous propose : reproduire une partie des fonctionnalités du « mode édition » que propose Moodle.
- **Pour cela, la logique d'affichage du contenu dans les UE doit être modifiée :** au lieu que le rangement dans les « groupes » (devoirs en cours, posts éponglés et/ou posts normaux) soit basé sur l'ordre chronologique, ce sont les professeurs qui vont totalement contrôler l'ordre d'affichage des posts à l'intérieur de ces groupes.
- Auprès de chaque post, un professeur de l'UE pourra voir une liste déroulante « Déplacer ce post vers... ». Les choix qui se trouvent dans cette liste sont de type « Avant le post (titre du post) », et une dernière option, « à la fin »
- Le post lui-même doit, idéalement, être exclu de sa liste déroulante associée.
- Selon votre préférence, il peut y avoir un bouton pour valider l'action de déplacement, ou alors elle se déclenche automatiquement une fois un choix pris dans la liste déroulante.
- Le déplacement peut avoir lieu via AJAX, pour que l'utilisateur le voie se réaliser sans recharger la page.
- Les déplacements devant être permanents, l'ordre des posts doit être stocké dans la BDD.

- Les « groupes particuliers » comme les devoirs en cours ou les posts épinglés (si vous les avez) peuvent ne pas proposer l'option de « changement de position », et garder la logique d'ordre chronologique...
- ...Mais vous pouvez aussi proposer l'option de les ranger « entre eux ».
- Attention à ce qui se passe quand un devoir est terminé, ou un post « dé-épinglé » : ces posts redeviennent alors des posts normaux, et il faudra établir une logique de où ils seront alors placés.
Le plus simple est sans doute de placer de tels posts en haut des « posts normaux », laissant ensuite la tâche de les ranger correctement au professeur.