

Compilation en tex.....	2
Scripts	2
Format du CSV.....	2
En-tête LaTeX	2
Correction	3
Processus	3
Difficultés et solutions apportées	3
Documentation.....	4
Moodle2amc	4
Ajouter des options à la page web	4
Bibliographie	4

Compilation en tex

Prérequis (tout peut être installé via “pip install amc2moodle”) :

- ImageMagick
- LaTeXML

Pour compiler en .tex on prend le fichier .xml et on utilise la librairie python (importée avec “from amc2moodle.moodle2amc import Quiz”).

Scripts

“creer_projet.sh <nom_projet>” qui permet d’instancier un dossier avec tous les dossiers nécessaires au bon fonctionnement d’AMC

“creer_copies.sh <nom_tex.tex> <liste_etudiants.csv>” après avoir glissé le .tex dans le dossier du projet, permet de créer les copies dans le dossier “copies”.

“corriger_copies.sh <liste_etudiants.csv>” après avoir glissé les copies remplies dans “scans/” et la liste d’étudiants qui permet de corriger et d’associer les notes à chaque étudiant.

Format du CSV

Le CSV peut contenir n’importe-quelle colonne pouvant donner des informations pertinentes (nom, prénom, etc). En revanche, il est **PRIMORDIAL** d’utiliser la même colonne pour l’identifiant pour la correction.

Dans mon cas de figure, j’utilise une colonne “id” qui donne le numéro d’étudiant catho (sur 10 chiffres). L’id est donné par “--liste-id” et “--notes-id” dans la commande d’association “association-auto”.

En-tête LaTeX

L’en-tête a deux composantes : la grille pour l’identifiant et le champ pour remplir le nom.

Seule la grille est importante. La grille est spécifiée via “\AMCcodeGridInt[h]{id}{10}” où “h” est la façon d’afficher les chiffres (ici à l’horizontale, les autres options peuvent être trouvées [ici](#)), “id” est le nom de la colonne du CSV pour le numéro catho de l’étudiant et “10” est le nombre de chiffres (nombre de colonnes/lignes).

Le reste de l'en-tête est du texte qui peut servir de préface à l'épreuve, instructions/consignes des enseignants, un titre pour le nom de l'épreuve ou autre.

Il est également possible de spécifier le nombre de pages différentes qu'il faut imprimer avec "`\exemplaire{nombre}`"

Correction

Le niveau de couleur nécessaire pour qu'une case soit considérée comme cochée est liée à l'option "`--seuil`".

La note peut être ajustée avec "`--notemin`" et "`--notemax`" (si on veut faire des QCM sur 5, 10, 20 ou autre)

Processus

1. On crée le projet avec le script "`creer_projet.sh <nom_projet>`" ou bien via la page web après s'être connecté
2. On insère le `qcm.tex` et la `liste_etudiants.csv` à la racine du dossier du nouveau projet. Ou bien upload via le site.
3. On lance la commande "`creer_copier.sh <nom_tex.tex> <liste_etudiants.csv>`" à la racine du projet qui va générer les `.pdf` dans le dossier "copies" du projet. Le bouton "générer copies" gère ça tout seul sur le site puis téléchargera un `.tar.gz` avec les copies.
4. Insérer les copies scannées et remplies dans "scans/". Ou bien via l'upload.
5. On lance la commande "`corriger_copies.sh <liste_etudiants.csv>`" qui corrigera les copies automatiquement les copies. Ou bien avec le bouton "corriger" de la page web.

Difficultés et solutions apportées

> Installation d'AMC

AMC a du mal à s'installer tout seul donc il faut très probablement lancer successivement l'installation à la main et parfois chercher sur internet le nom des paquets.

> Scripts bash dans d'autres dossiers.

AMC fonctionne sur le répertoire courant d'exécution. Si on fait un script il va tout lancer dans le répertoire où on a lancé le script bash, même si ce dernier lance la commande "`cd`". Donc je lance tout dans le même dossier (même si c'est un peu sale).

> Acquisition de code

La documentation d'AMC est assez cryptique et divisée donc il faut fouiller sur d'autres pages que le PDF ou la documentation sur AMC en CLI, notamment au niveau de la page sur le formatage.

> Utilisation de moodle2amc

<Aucune solution à ce jour>

Documentation

Moodle2amc

Pour implémenter moodle2amc, il suffit de lancer la commande correspondante dans le script "compiler_qcm.sh". Et de modifier les options dans le formulaire et la route correspondante.

Ajouter des options à la page web

Pour permettre à l'utilisateur du site web de choisir lui-même les arguments pour chacune des commandes (comme le seuil de noir pour considérer qu'une case est cochée ou le nombre de copies au moment de compiler le fichier .tex), il faut :

- Modifier le script correspondant en rajoutant un argument et en remplaçant la valeur déjà intégrée par la nouvelle variable
- Ajouter le champ dans le formulaire correspondant sur la page html
- L'ajouter à la requête POST avec la fonction correspondante dans le <script> du html
- Modifier le contenu de la route correspondante dans le fichier "projetRoutes.js" au niveau de l'exécution du script et de la réception des données transmises dans la requête

Bibliographie

Librairie python pour compiler moodle xml en tex :

<https://github.com/nennigb/amc2moodle/>

Documentation AMC en ligne de commande : [https://project.auto-multiple-choice.net/projects/auto-multiple-choice/wiki/Using AMC in command line only](https://project.auto-multiple-choice.net/projects/auto-multiple-choice/wiki/Using_AMC_in_command_line_only)

Documentation AMC sur le formatage du fichier LaTeX <https://www.auto-multiple-choice.net/auto-multiple-choice.en/latex.shtml>

Documentation AMC sur la procédure à suivre (via le GUI) <https://www.auto-multiple-choice.net/auto-multiple-choice.en/graphical-interface.shtml>