



HAL
open science

ASKER : un outil auteur pour la création d'exercices d'auto-évaluation

Marie Lefevre, Nathalie Guin, Baptiste Cablé, Brice Buffa

► **To cite this version:**

Marie Lefevre, Nathalie Guin, Baptiste Cablé, Brice Buffa. ASKER : un outil auteur pour la création d'exercices d'auto-évaluation. Atelier EAEI (Évaluation des Apprentissages et Environnements Informatiques) - Conférence EIAH 2015, Jun 2015, Agadir, Maroc. hal-01177830

HAL Id: hal-01177830

<https://hal.science/hal-01177830>

Submitted on 17 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ASKER : un outil auteur pour la création d'exercices d'auto-évaluation

Marie Lefevre, Nathalie Guin, Baptiste Cablé, Brice Buffa
Université de Lyon, CNRS - Université Lyon 1, LIRIS, UMR5205, F-69622, France
prenon.nom@liris.cnrs.fr

Résumé. Cet article présente ASKER, une plateforme permettant à un enseignant de créer et diffuser des exercices d'auto-évaluation pour ses élèves. Actuellement utilisée en première année de licence à l'Université Lyon 1, elle permet aux étudiants d'effectuer des exercices dans le but d'évaluer leur acquisition des notions jugées importantes par l'enseignant. ASKER permet la création d'exercices (appariements, groupements, QROC, QCM) qui peuvent être utilisés pour évaluer les apprentissages dans de nombreux domaines. Pour créer des exercices permettant d'évaluer une notion, l'enseignant définit un modèle d'exercices qui permettra la génération d'exercices variés, utilisant des ressources de type texte ou image. La possibilité pour l'apprenant de demander la génération de plusieurs exercices issus du même modèle lui permet de s'évaluer à plusieurs reprises sur les mêmes notions, sans que l'enseignant n'ait à définir de manière répétitive de nombreux exercices.

Mots-clés. Auto-évaluation, génération semi-automatique d'exercices, outil auteur, contexte universitaire

1 Introduction

L'objectif des travaux décrits dans cet article est de permettre à un apprenant d'évaluer ses connaissances dans le cadre d'un parcours de formation pour lequel un enseignant définit des objectifs d'apprentissage. Notre approche consiste à permettre à l'enseignant de proposer à l'apprenant des exercices portant sur les notions à acquérir, exercices que l'apprenant pourra utiliser s'il le souhaite pour évaluer sa maîtrise des notions mises en jeu par les exercices. Nous nous plaçons donc ici dans une évaluation à caractère formatif.

Nous proposons un outil auteur qui laisse à l'enseignant la liberté de fixer les notions sur lesquelles les apprenants pourront s'évaluer, et qui lui permet de créer des exercices destinés à évaluer la maîtrise de ces notions. Afin de répondre au besoin du plus grand nombre, nous avons choisi de considérer des types d'exercices qui pourront s'appliquer à de nombreux domaines, tels que les QCM, appariements, groupements, ordonnancements, textes à trou, etc.

La section suivante explique pourquoi nous avons choisi une démarche fondée sur des générateurs d'exercices pour répondre à cette problématique de la création d'exercices d'auto-évaluation. Nous apportons ensuite un éclairage sur les processus d'acquisition des connaissances nécessaires à la mise en œuvre de cette démarche,

avant de présenter l'architecture de la plateforme que nous avons développée : ASKER (Authoring tool for aSsessing Knowledge genErating exeRcises). Enfin, après avoir montré comment cette plateforme peut être utilisée à la fois par l'enseignant et par les apprenants, nous effectuons un bilan de l'utilisation d'ASKER dans le cadre d'un enseignement en première année de Licence à l'Université Lyon 1.

2 Un outil auteur fondé sur des générateurs d'exercices

Notre objectif était de permettre à un auteur de proposer aux apprenants des exercices d'auto-évaluation en ligne. De tels exercices permettent à l'apprenant de tester en autonomie son niveau de maîtrise de ce qu'il a appris dans le cours, que ce cours soit en ligne ou en présentiel. Les apprenants peuvent échouer lors des premières tentatives de réponse aux exercices si les connaissances ne sont pas maîtrisées. Il est donc possible qu'un apprenant soit amené à répondre plusieurs fois à un même exercice avant d'obtenir un succès. Afin que l'apprenant ne se laisse pas influencer par ses précédentes résolutions, il est nécessaire que l'exercice d'auto-évaluation soit différent d'une fois sur l'autre, tout en évaluant la même connaissance. Toutefois, il ne semble pas raisonnable de demander à l'auteur de rédiger de nombreuses versions de l'énoncé. C'est pourquoi nous avons choisi de recourir à des générateurs d'exercices que l'auteur peut facilement utiliser quel que soit le domaine. Il s'agit cependant d'un processus *semi-automatique* de génération d'exercices, afin de laisser l'enseignant intervenir sur le choix des critères que devront respecter les exercices.

Pour répondre aux besoins d'auto-évaluation des apprenants, les propriétés attendues de l'outil auteur étaient les suivantes :

- l'exercice est différent d'une fois sur l'autre ;
- l'auteur est maître du contenu de l'exercice et a la garantie qu'il correspond à ses attentes en matière de contenu pédagogique ;
- les générateurs d'exercices sont utilisables dans un grand nombre de domaines et de niveaux scolaires ;
- le diagnostic de la réponse est fait automatiquement et en temps réel ;
- la construction des exercices avec les générateurs est un gain de temps pour l'auteur par rapport à une création exercice par exercice ;
- la création d'exercices ne demande pas de compétences techniques.

Les générateurs d'exercices de la littérature peuvent se répartir en trois catégories : les générateurs manuels, automatiques et semi-automatiques.

Souvent utilisés dans le cadre des outils auteurs, les *générateurs manuels* laissent une grande liberté à l'auteur, qui définit précisément le contenu de l'exercice ainsi que toutes les options de mise en forme. Certains outils auteurs issus du commerce tels que Articulate Quizmaker [1] ou Hot Potatoes [2] sont couramment utilisés pour créer, de manière assistée, des exercices sur papier ou informatisés. Les plateformes d'apprentissage en ligne Claroline [3] et Moodle [4], couramment utilisées dans l'enseignement supérieur, proposent également leurs propres outils d'édition d'exercices. Avec ce type de générateurs, l'auteur est assuré d'avoir un exercice correspondant précisément à ses attentes, ce qui répond à l'un de nos besoins. Par contre, l'auteur doit créer chaque instance d'exercice une par une en précisant son

contenu. Ce type de générateur n'est pas capable de créer automatiquement un grand nombre d'exercices évaluant la même compétence.

Les *générateurs automatiques* ont eux cette capacité, mais laissent malheureusement peu de place à l'auteur dans le processus de création. Avec ce genre de générateur, un grand nombre d'exercices est créé automatiquement sans que l'auteur puisse influencer les choix du système. Il peut simplement choisir la catégorie de l'exercice (forme, thème, connaissances abordées) mais ne peut pas agir sur le contenu ni sur des contraintes précises. On peut citer par exemple Alfalex [5], le générateur du Reading Tutor [6], ou celui d'Aplusix [7].

Afin de profiter des fonctionnalités des générateurs automatiques mais en laissant à l'auteur la liberté éditoriale sur les exercices créés, nous avons opté pour les *générateurs semi-automatiques*, qui combinent les avantages des deux catégories précédentes. Ces générateurs proposent à l'auteur de définir un modèle d'exercices qui est ensuite instancié pour donner lieu à un grand nombre d'exercices [8, 9]. Certaines plateformes de e-learning proposent des exercices faisant intervenir des variables, qui se rapprochent de la notion de modèle d'exercices [4, 10, 11, 12]. Ce type de générateurs répond partiellement à notre besoin mais se limite aux domaines nécessitant du calcul.

Pour disposer de générateurs semi-automatiques adaptés à de nombreux domaines et incluant d'autres types d'exercices, nous avons choisi d'utiliser l'approche GEPPETO [13]. Cette approche consiste à permettre à l'enseignant d'exprimer des contraintes sur les exercices à générer. Pour cela, il est nécessaire de disposer d'un modèle des types d'exercices qu'il est possible de générer, afin de connaître le type de contraintes qu'il faut permettre à l'enseignant d'exprimer. La section suivante précise ainsi les modèles guidant l'acquisition des connaissances nécessaires à la génération d'exercices selon l'approche GEPPETO.

3 Acquisition de connaissances pour la génération d'exercices

La Figure 1 présente l'approche GEPPETO (GEneric models and processes to Personalize learners' PEdagogical activities according to Teaching Objectives). Cette approche permet d'acquérir des connaissances de plusieurs niveaux, auprès d'experts et auprès d'enseignants, pour générer des exercices.

Dans GEPPETO, un méta-méta-modèle d'exercices a été défini par l'équipe de recherche [14]. Il précise quelles sont les connaissances qu'un expert devra définir pour créer un méta-modèle d'exercices d'un type donné (cf. A sur la Figure 1), par exemple un méta-modèle des exercices de type QCM, ou un méta-modèle d'exercices de type appariement.

Ce méta-modèle d'exercices de type X ou Y permet ensuite à l'enseignant de préciser des contraintes définissant un modèle d'exercices (cf. B sur la Figure 1). Selon le type d'exercices, les contraintes ne seront pas les mêmes, c'est donc la raison d'être du méta-modèle d'exercices. L'enseignant pourra par exemple utiliser le méta-modèle des exercices de type QCM pour définir un modèle d'exercices de type QCM portant sur un sujet donné et comportant N questions avec pour chacune M propositions avec exactement une réponse juste par question.

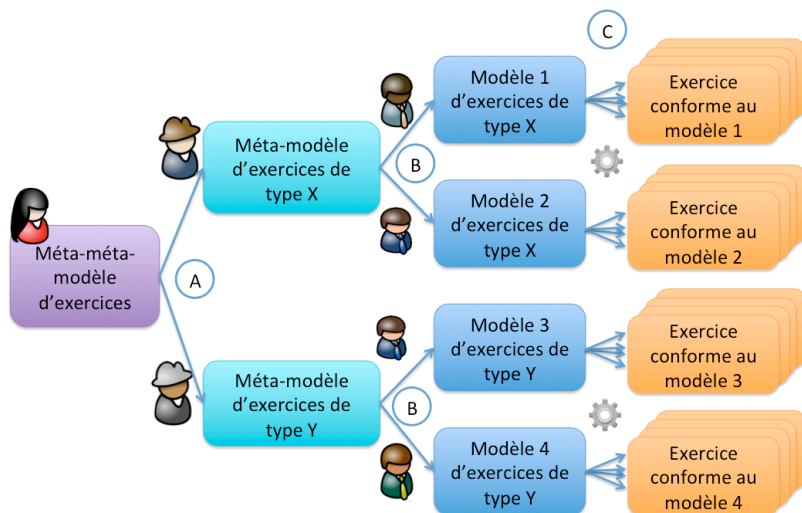


Fig. 1. L'approche GEPPETO.

À partir de tels modèles d'exercices, des générateurs d'exercices peuvent construire plusieurs exercices conformes à ces modèles (cf. C sur la Figure 1). Les générateurs d'exercices permettant d'exploiter ces modèles dépendent du type d'exercices, et donc du méta-modèle d'exercices de type X ou Y.

On constate donc que l'approche GEPPETO requiert deux processus d'acquisition des connaissances. Une acquisition des connaissances de l'expert pour la création de méta-modèles d'exercices (cf. A sur la Figure 1). Cette acquisition est effectuée une seule fois pour chaque type d'exercices, et nécessite une interface fondée sur le méta-méta-modèle d'exercices. Une acquisition des connaissances de l'enseignant pour la création des modèles d'exercices (cf. B sur la Figure 1). Cette acquisition est effectuée de manière régulière pour la construction de modèles d'exercices variés, et nécessite également une interface qui, elle, est fondée sur le méta-modèle d'exercices d'un type donné (les contraintes que l'enseignant doit définir dépendant du type d'exercices).

Les méta-modèles ainsi que le méta-méta-modèle sont indépendants du domaine pour lequel un exercice sera généré. Par exemple le méta-modèle "Identification des parties du texte" précise comment formuler la consigne, comment caractériser les différentes ressources de type texte, et comment décrire les actions à effectuer sur ces ressources pour générer les exercices.

À chacun de ces méta-modèles est associé un générateur d'exercices. Une interface associée au générateur et fondée sur le méta-modèle permet à l'enseignant de définir des contraintes sur les exercices à générer. C'est à ce moment là, lors de la création du modèle d'exercices (par exemple un texte à trous portant sur la conjugaison des verbes du second groupe à l'imparfait), que l'application à un domaine et à un niveau

d'études sera effectuée. Les exercices générés à partir du modèle d'exercices sont donc bien entendu dépendants du domaine.

Les méta-modèles étant tous conformes au méta-méta-modèle, l'ensemble des générateurs d'exercices partage une architecture commune [15].

4 Architecture de la plateforme ASKER

Nous avons donc choisi pour l'outil auteur un fonctionnement inspiré de l'approche GEPPETO : après avoir choisi un type d'exercices (QCM, appariement, texte à trous...), l'auteur peut créer un modèle d'exercices qui décrit le contenu et la forme de l'exercice qu'il souhaite créer, mais sans forcément le contraindre totalement. En exploitant ce modèle, un générateur d'exercices est capable de mettre à disposition de l'apprenant un grand nombre d'exercices différents évaluant la même compétence. Chaque instance d'exercice ainsi générée sera interactive : l'apprenant y répondra en ligne et obtiendra un diagnostic de sa réponse.

Les types d'exercices que nous avons retenus sont les suivants [15] :

- identification de parties du texte (inclut le texte à trous) ;
- organisation d'éléments (ordonnancement, groupement, appariement) ;
- annotation d'illustrations ;
- questionnaire à choix multiples (QCM) ;
- questionnaire à réponses ouvertes et courtes (QROC).

La Figure 2 présente l'architecture de l'outil auteur ASKER. Le bloc du haut est constitué des différents niveaux de représentation des exercices, issus de l'approche GEPPETO : les méta-modèles de types d'exercices, les modèles d'exercices, et les instances d'exercices. Dans le bloc central se trouvent les trois mécanismes manipulant ces représentations des exercices. Le bloc inférieur contient les ressources et connaissances utilisées dans le processus de création des exercices.

Les *ressources* sont les éléments de base qui sont utilisés pour construire les exercices. Il s'agit, par exemple, de textes, d'images ou de séquences d'éléments. Chaque ressource dispose de métadonnées la caractérisant pour faciliter les recherches (thème, niveau, etc.) ainsi que de métadonnées enrichissant la ressource pour définir des exercices, comme la légende d'une image ou des annotations sur des zones de l'image.

Les *connaissances du domaine* sont des connaissances relatives à une matière et indépendantes du type d'exercices. Il s'agit par exemple d'une formule, de la liste des mots-clés du langage C, d'une méthode pour détecter le genre d'un nom, de la valeur d'une constante, etc.

L'auteur choisit un type d'exercices T. Un expert ayant décrit le méta-modèle MMT des exercices de type T, l'auteur crée un modèle d'exercices MT conforme à MMT à l'aide d'un outil dédié s'appuyant sur les connaissances du type d'exercices T contenues dans MMT. L'auteur peut générer quelques exercices ExoT instances de MT pour tester si le modèle donne bien lieu aux exercices attendus. Cet outil facilite le choix des ressources, et permet à l'auteur d'en créer de nouvelles.

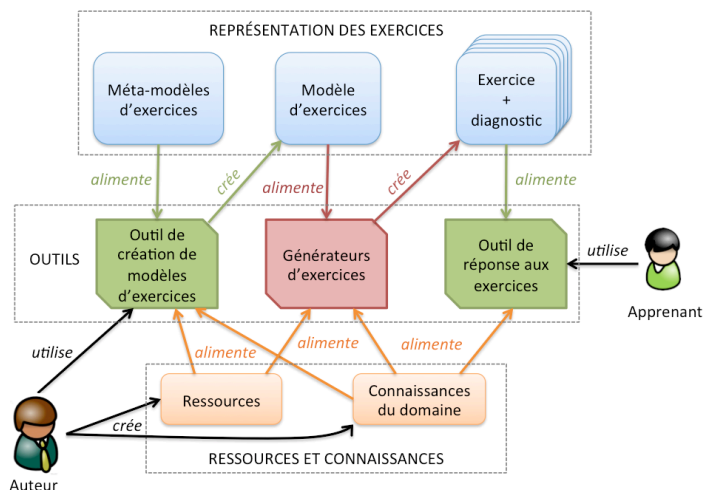


Fig. 2. Architecture de la plateforme ASKER.

Le générateur d'exercices de type T associé à MMT reçoit donc en entrée un modèle d'exercices MT qu'il instancie pour donner lieu à un exercice ExoT en sortie. L'exercice est conforme au modèle MT et donc aux choix de l'auteur qui l'a créé. Le générateur ne nécessite aucune intervention humaine. Il dispose de toutes les informations nécessaires dans le modèle d'exercices MT et fait appel, si nécessaire, aux ressources et aux connaissances du domaine. Le générateur est utilisé à chaque fois que l'on veut obtenir une nouvelle instance d'exercice ExoT (qui contient le diagnostic) à partir du modèle MT.

L'exercice est alors présenté à l'apprenant *via* un outil de résolution qui met en forme l'exercice, recueille la réponse de l'apprenant et lui en fournit le diagnostic.

5 La plateforme ASKER

La plateforme ASKER peut d'une part être utilisée par un enseignant pour créer des modèles d'exercices, et d'autre part par l'apprenant pour effectuer des exercices générés à partir de ces modèles et obtenir un diagnostic dans le but de s'auto-évaluer.

Un outil auteur pour l'enseignant. ASKER permet actuellement à un enseignant de créer des modèles d'exercices de type QCM, QROC, appariement, groupement. L'enseignant peut créer des ressources de type texte, image, question de QCM, question de QROC. Sur chacune de ces ressources, il peut ajouter des méta-données qui seront utilisées par les modèles d'exercices exploitant la ressource. Une ressource peut être utilisée pour plusieurs modèles d'exercices de types différents. Ainsi, une même méta-donnée sur une ressource pourra être utilisée à la fois pour un exercice d'appariement, de groupement ou d'ordonnancement.

La Figure 3 présente l'interface auteur destinée à un créer un modèle d'exercices d'appariement. L'auteur a sélectionné des ressources (partie droite de la Figure 3) de type texte, et a filtré celles destinées au CC (contrôle continu) numéro 3 de l'UE LIF3. Plusieurs textes de fonctions en langage de programmation Scheme peuvent être utilisés. Sur ces textes, des méta-données indiquent la spécification de chaque fonction, son (ou ses) type(s) d'entrée et son type de sortie.

Pour créer un modèle d'exercices, l'auteur définit (partie gauche de la Figure 3) qu'il souhaite créer des exercices dans lesquels 4 fonctions doivent être associées à leur spécification. Le générateur d'exercices utilise ensuite ce modèle d'exercices pour créer des exercices répondant à ces contraintes, en exploitant les ressources disponibles décrivant des fonctions.

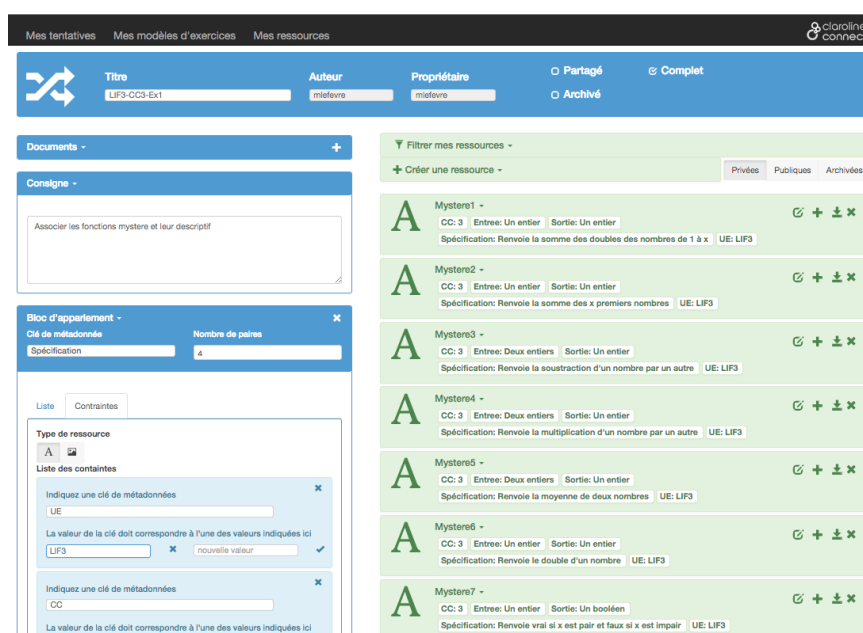


Fig. 3. Interface auteur pour la création d'un modèle d'exercices d'appariement.

Un outil d'auto-évaluation pour l'apprenant. L'enseignant propose à ses étudiants des modèles d'exercices correspondant aux notions étudiées en cours. Pour chaque modèle d'exercices, l'apprenant peut demander la génération de plusieurs instances d'exercices. Il résout tout d'abord un premier exercice issu du modèle d'exercices, le système effectue un diagnostic de ses réponses et lui affiche un retour (en vert et rouge) sur ses réponses ainsi que la correction (en bleu) de l'exercice (cf. Figure 4).

Un commentaire prévu par l'enseignant et expliquant une erreur courante ou rappelant une notion importante peut également être affiché. L'apprenant peut alors revoir si nécessaire son cours et demander un nouvel exercice issu du même modèle d'exercices.

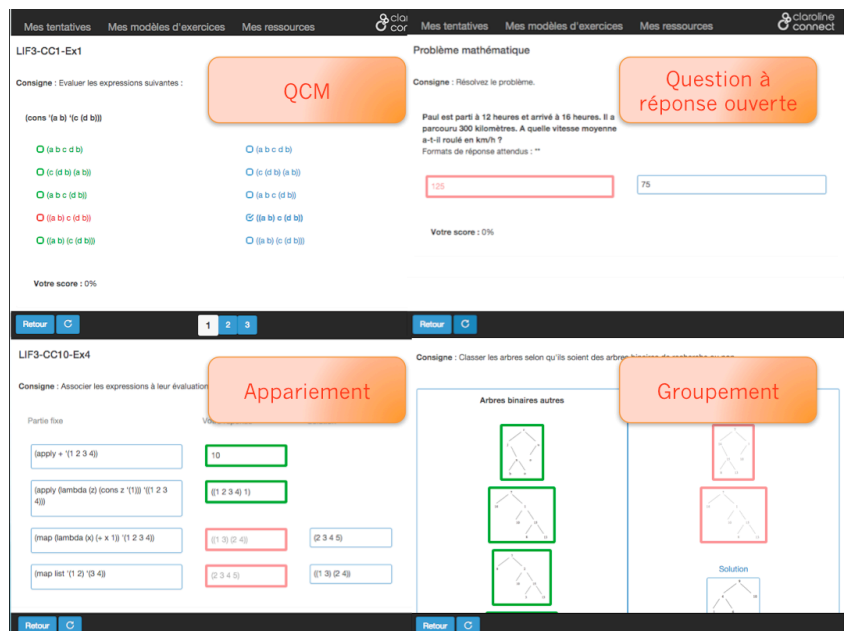


Fig. 4. Rétroactions fournies par ASKER à l'apprenant sur sa résolution de plusieurs types d'exercices.

6 Utilisation de la plateforme ASKER à l'université

La plateforme ASKER est utilisée à l'Université Lyon 1 dans le cadre d'une unité d'enseignement (UE) en première année de licence Mathématiques et Informatique. Cette UE présente les concepts de la programmation fonctionnelle et récursive. Elle est dispensée soit en tant que première UE d'informatique au cours du semestre d'automne, soit en tant que seconde UE d'informatique au semestre de printemps. Il s'agit donc ici d'une utilisation d'ASKER pour un enseignement qui n'est pas informatisé, la plateforme étant un outil complémentaire à l'enseignement présentiel.

Dans le cadre du plan *réussite en licence*, nous avons, grâce à la plateforme ASKER, proposé aux étudiants de s'auto-évaluer sur la compréhension des concepts présentés en cours avant de venir en Travaux Dirigés. Cela permet ainsi aux étudiants de se situer par rapport à l'acquisition des notions abordées dans l'UE, de préparer les interrogations effectuées chaque semaine en Travaux Dirigés, et d'effectuer un diagnostic de leurs difficultés pour préparer les séances de soutien.

Pour cela, nous avons pour chacun des neuf cours magistraux proposé une série de modèles d'exercices. En instanciant les méta-modèles d'exercices, nous avons fourni aux étudiants 18 modèles d'appariement portant sur l'association d'objets de type textes et/ou images, comme celui présenté sur la Figure 3 ; 9 modèles d'exercices de groupements où les étudiants doivent construire des familles d'objets, par exemple

regrouper des expressions selon les résultats de leur évaluation ; et 8 modèles de QCM portant sur 38 ressources de type questions.

La création de ces 35 modèles et de leurs 121 ressources a représenté entre 1h et 2h de travail chaque semaine, durant les 12 semaines d'enseignement, pour l'enseignant responsable du semestre d'automne. C'est une charge de travail conséquente mais qui ne concernait que la phase d'initialisation. Ces différents modèles et leurs ressources sont à présent exploités et complétés par l'enseignant du semestre de printemps en un temps plus raisonnable : 1/2h par semaine.

Coté étudiant, sur les 124 étudiants inscrits à l'UE en septembre 2014, 92 ont créé un compte sur la plateforme. Au cours du semestre, 7069 exercices ont été générés et 6940 réponses ont été soumises pour vérification. Afin de mesurer l'apport de la plateforme pour les étudiants, nous avons diffusé un questionnaire auprès des 93 étudiants ayant terminé l'UE. Nous avons obtenu 35 réponses. Parmi les étudiants ayant répondu, 34 ont créé un compte et utilisé la plateforme ainsi :

- Fréquence d'utilisation : 70% des étudiants se sont connectés chaque semaine, 21% l'ont fait au début du semestre mais plus après, et 9% la moitié du temps ;
- Durée : 44% ont travaillé entre 5 et 10 mn par séance, 38% entre 10 et 20 mn ;
- 82% des étudiants ont répondu à plusieurs instances de chaque exercice ;
- Parmi les objectifs visés par les étudiants, 44% se connectaient pour comprendre les concepts du CM, 41% pour identifier les notions non comprises, et 91% avant les TD afin de préparer les interrogations.

7 Conclusion et perspectives

Nous avons dans cet article présenté ASKER, une plateforme permettant à un enseignant de créer des exercices d'auto-évaluation pour ses élèves. Cet outil auteur permet la création d'exercices (appariements, groupements, QROC, QCM) qui peuvent être utilisés pour évaluer les apprentissages dans de nombreux domaines. Pour créer des exercices permettant d'évaluer une notion, l'enseignant définit un modèle d'exercices qui permettra la génération d'exercices variés, utilisant des ressources de type texte ou image. La possibilité pour l'apprenant de demander la génération de plusieurs exercices issus du même modèle lui permet de s'évaluer à plusieurs reprises sur les mêmes notions, sans que l'enseignant n'ait à définir de manière répétitive de nombreux exercices.

Cette plateforme peut être utilisée dans le cadre d'un enseignement à distance ou en complément d'un enseignement présentiel. Elle peut être associée à des environnements numériques de travail variés. À l'Université Lyon 1, ASKER est un plugin de Claroline Connect [16]. La création d'un plugin ASKER pour Moodle est prévue à court terme. Nous travaillons aussi à enrichir les types d'exercices qui peuvent être créés avec ASKER. Un générateur d'exercices sur des textes à trous va ainsi être intégré à ASKER à court terme. À l'Université Lyon 1, après l'utilisation dans l'UE d'introduction à la programmation fonctionnelle et récursive, nous envisageons de l'utiliser dans d'autres UE d'informatique de première année de licence. Une utilisation dans le cadre des MOOCs développés à Lyon 1 et par Open Classrooms [17] est également à l'étude.

Les générateurs d'exercices de l'outil ASKER étant indépendants du domaine, les connaissances du domaine n'existent pas au départ. Leur acquisition représente donc un enjeu important. L'utilisateur principal d'ASKER étant l'auteur, il serait intéressant que ce soit lui qui construise les connaissances du domaine, comme il le fait déjà pour des formules. Le plus convivial pour lui serait qu'il crée les connaissances du domaine requises au fur et à mesure qu'il en a besoin pendant la définition des modèles d'exercices. Le système pourrait l'assister dans cette tâche en lui proposant une généralisation des informations qu'il renseigne pour créer ses modèles d'exercices. Nous travaillons actuellement sur l'exploitation des traces d'activités de l'enseignant utilisant ASKER pour permettre au système d'assister l'enseignant dans cette élicitation des connaissances du domaine.

Remerciements. La plateforme ASKER a été développée grâce au soutien des Investissements d'Avenir (PIA e-éducation 1, projet CLAIRE) et de la mission CNRS COAT.

References

1. Site web de Articulate Quizmaker. <http://www.articulate.com/products/quizmaker.php> (consulté en avril 2015)
2. Site web de Hot Potatoes. <http://hotpot.uvic.ca/> (consulté en avril 2015)
3. Consortium Claroline. <http://www.claroline.net/> (consulté en avril 2015)
4. Documentation Moodle. <http://docs.moodle.org/2x/fr> (consulté en avril 2015)
5. Selva, T. : Génération automatique d'exercices contextuels de vocabulaire. *Traitement Automatique du Langage Naturel* (2002) 185–194
6. Mostow, J., Beck, J.-E., Bey, J., Cuneo, A., Sison, J., Tobin, B., Valeri, J.: Using automated questions to assess reading comprehension, vocabulary and effects of tutorial interventions. *Technology, Instruction, Cognition and Learning*, Vol. 2 (2004) 97–134
7. Bouhineau, D., Nicaud, J.-F. : Aplusix, un EIAH de l'algèbre. In: M. Grandbastien et J-M Labat (eds.): *Environnements Informatiques pour l'Apprentissage Humain*, Chap. 15, Hermes (2006) 333–350
8. Duclosson, N., Jean-Daubias, S., Riot, S. : AMBRE-enseignant : un module partenaire de l'enseignant pour créer des problèmes. *EIAH* (2005) 353–358
9. Prévité, D., Delozanne, E., Grugeon, B. : Génération d'exercices de diagnostic de compétences en algèbre. *EIAH* (2007)
10. WWW Interactive Multipurpose Server. <http://wims.unice.fr/wims/> (consulté en avril 2015).
11. EULER. <http://euler.ac-versailles.fr/> (consulté en avril 2015)
12. Auzende, O., Giroire, H., Le Calvez, F. : Propositions d'extensions à IMS-QTI 2.1 pour l'expression de contraintes sur les variables d'exercices mathématiques. *EIAH* (2007)
13. Lefevre, M., Guin, N., Jean-Daubias, S. : A Generic Approach for Assisting Teachers During Personalization of Learners' Activities. *Workshop PALE, UMAP* (2012) 35-40
14. Lefevre, M. : Processus unifié pour la personnalisation des activités pédagogiques : méta-modèle, modèles et outils. Thèse de doctorat, Université Claude Bernard Lyon 1 (2009)
15. Lefevre, M., Jean-Daubias, S., Guin, N. : Generation of pencil and paper exercises to personalize learners' work sequences: typology of exercises and meta-architecture for generators. *E-Learn* (2009) 2843-2848
16. Claroline Connect. http://claco.univ-lyon1.fr/icap_wiki/360 (consulté en avril 2015)
17. OpenClassroom. <http://openclassrooms.com/> (consulté en avril 2015)