

MA-004-150-1

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad DAHLEB de Blida



Faculté des sciences

Département Informatique



En vue d'obtenir le diplôme de Master

Option : Ingénierie de logiciel

**Sujet :**

**Correction automatique des réponses courtes  
dans la plate-forme MOODLE**

Mémoire réalisé par :

BOULAHIA Soumia Meriem.

HAROUS Lyna Racha.

Promoteur :

Dr.BENNOUAR Djamel.

Soutenu le : Mardi 02 Juillet 2013.

Devant le jury composé de :

Mm. CHERFA

Présidente.

Mm. MENACAER

Examinatrice.

Mm. JELLATOU

Examinatrice.

MA-004-150-1

Promotion 2012/2013

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



## ملخص:

في هذه الأطروحة، نقوم بتصميم و تنفيذ وحدة (مقياس) اختبار جديدة من أجل الإجابات المكتوبة من طرف الممتحن بحرية مع إضافة خيارات جديدة لتحسين التصحيح الأوتوماتيكي، لمساعدة الطالب في الحصول على العلامة الكاملة و إن وجدت أخطاء إملائية ناجمة عن النسيان أو عدم الانتباه. تتمثل الفكرة الأساسية في إعطاء الأستاذ دوراً مهماً خلال عملية تحضير الأسئلة بتحويلها إلى عملية تسمح بطريقة فعالة في توجيه عملية التقييم و ذلك بأخذ ضعف الطلبة اللغوي بعين الاعتبار و الإحاطة بأخطاء السهو.

## Résumé :

Dans ce travail, nous présentons la conception et l'implémentation d'un nouveau module de test pour les réponses librement rédigées avec de nouvelles options pour améliorer la correction automatique de façon à aider l'étudiant à avoir la note complète même dans le cas d'erreur d'orthographe, d'oubli ou d'inattention. L'idée fondamentale de cet outil consiste à donner un rôle important à l'enseignant durant le processus de préparation de question en transformant ce processus de préparation de question en une préparation qui permet de guider efficacement le processus d'évaluation pour que ce dernier puisse prendre en compte la faiblesse des examinés sur le plan linguistique et de prendre en charge les erreurs d'inattention.

## Abstract

In this work, we present the conception and implementation of a new test module for answers freely written with new options for improving the automatic correction in order to help the student to have the complete notes, even in the case of spelling error, forgetting or inattention.

The basic idea of this tool is to give an important role to the teacher in the process of preparation of the question by transforming this process into a preparation that can effectively guide the evaluation process so that the latter can take into account the weakness examined in terms of language and to support careless mistakes.



## *REMERCIEMENTS*

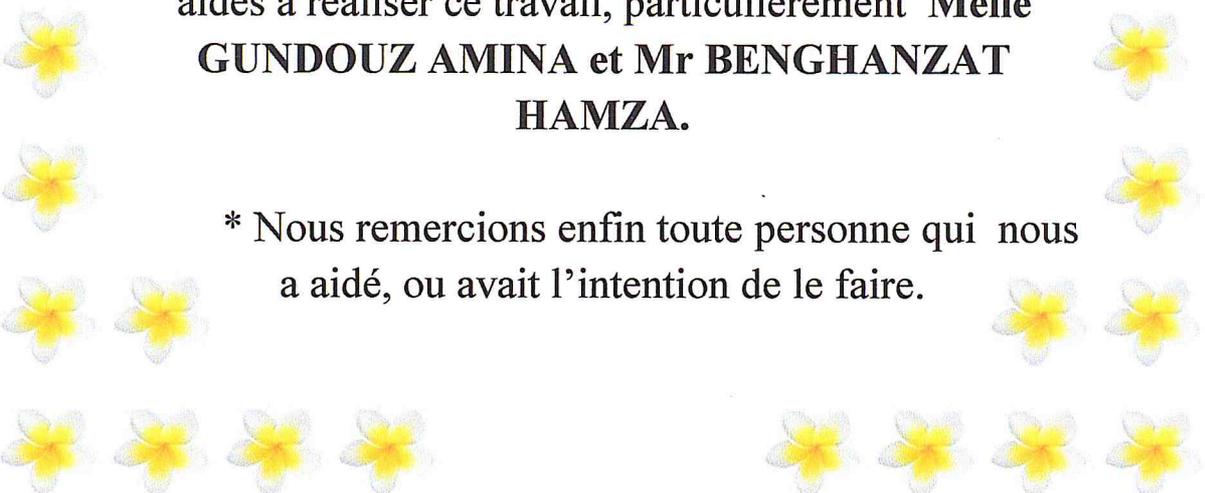
\* Nous tenons tout d'abord à remercier **ALLAH** qui nous a guidé vers le chemin du savoir et qui nous a aidé à réaliser ce modeste travail.

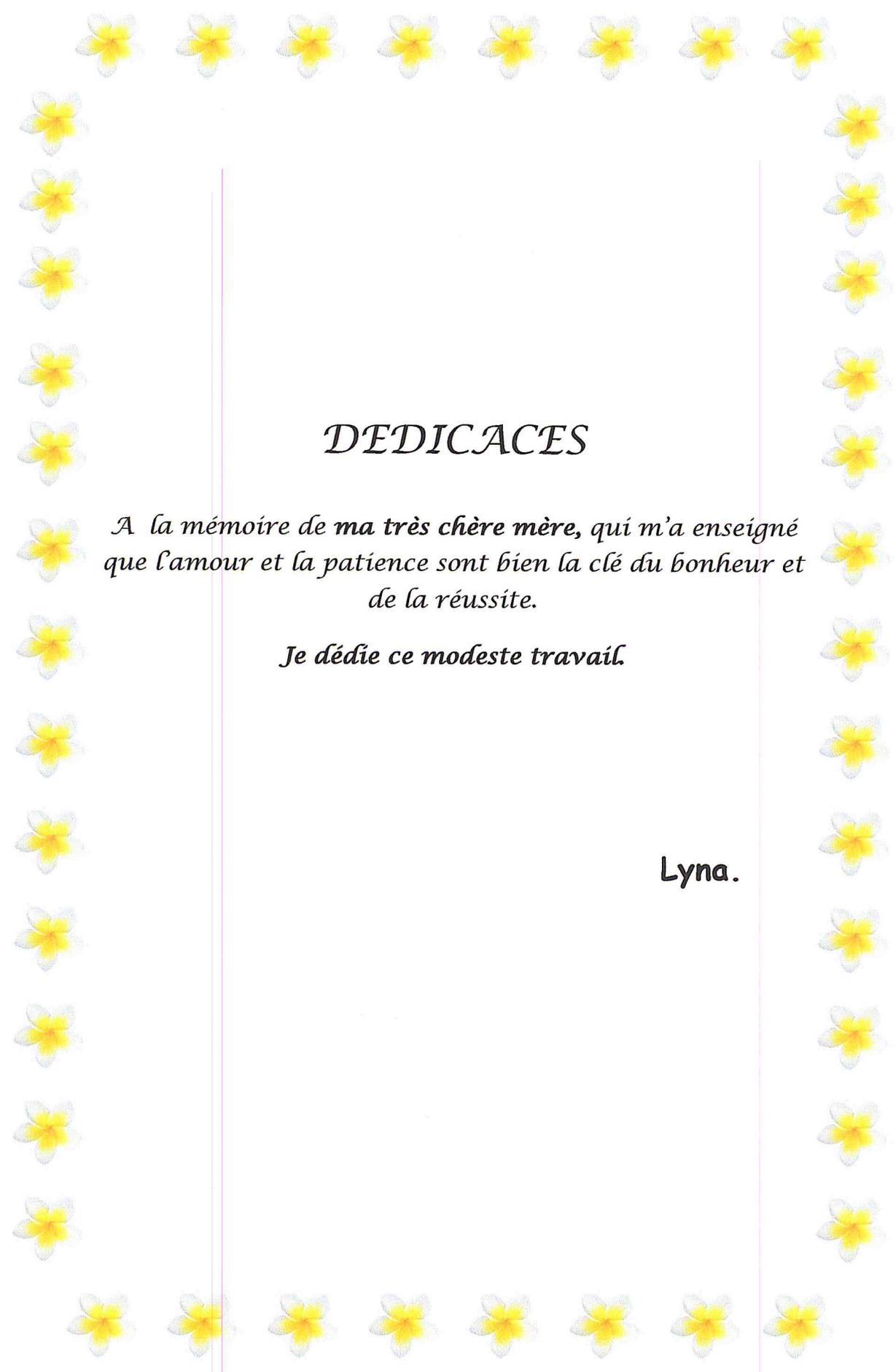
\* Nous remercions infiniment notre promoteur **Dr DJAMAL BENNOUAR** pour l'aide compétente qu'il nous a apportée, pour sa patience et son encouragement à finir ce travail, et qui été pour nous une grande source de motivation et d'inspiration pour continuer et accomplir cette recherche.

\* Nous remercions aussi les membres de jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions.

\* Nous remercions aussi tous ceux qui nous ont aidés à réaliser ce travail, particulièrement **Melle GUNDOUZ AMINA et Mr BENGHANZAT HAMZA.**

\* Nous remercions enfin toute personne qui nous a aidé, ou avait l'intention de le faire.



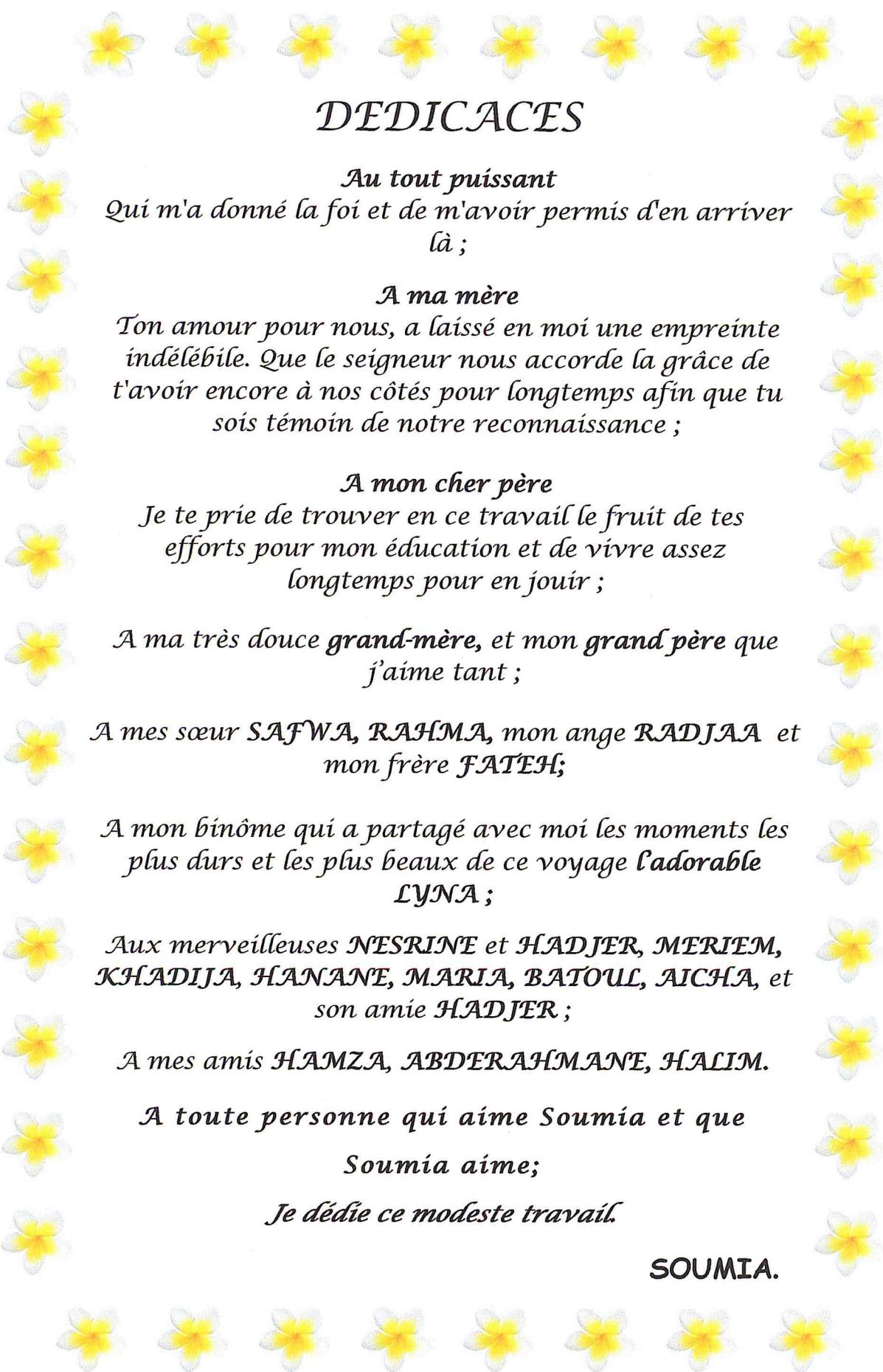


## *DEDICACES*

*A la mémoire de ma très chère mère, qui m'a enseigné  
que l'amour et la patience sont bien la clé du bonheur et  
de la réussite.*

*Je dédie ce modeste travail.*

**Lyna.**



## DEDICACES

*Au tout puissant  
Qui m'a donné la foi et de m'avoir permis d'en arriver  
là ;*

*A ma mère  
Ton amour pour nous, a laissé en moi une empreinte  
indélébile. Que le seigneur nous accorde la grâce de  
t'avoir encore à nos côtés pour longtemps afin que tu  
sois témoin de notre reconnaissance ;*

*A mon cher père  
Je te prie de trouver en ce travail le fruit de tes  
efforts pour mon éducation et de vivre assez  
longtemps pour en jouir ;*

*A ma très douce grand-mère, et mon grand père que  
j'aime tant ;*

*A mes sœur SAFWA, RAHMA, mon ange RADJAA et  
mon frère FATEH;*

*A mon binôme qui a partagé avec moi les moments les  
plus durs et les plus beaux de ce voyage l'adorable  
LYNA ;*

*Aux merveilleuses NESRINE et HADJER, MERIEM,  
KHADIJA, HANANE, MARIA, BATOUL, AICHA, et  
son amie HADJER ;*

*A mes amis HAMZA, ABDE RAHMANE, HALIM.*

*A toute personne qui aime Soumia et que  
Soumia aime;*

*Je dédie ce modeste travail.*

**SOUMIA.**

# Sommaire

Résumé.

Introduction générale.

## Chapitre 1 : Etat de l'art sur la correction automatique.

1. Etat de l'art sur la correction automatique .....	5
1.2. Les types de tests en ligne .....	5
1.3. Techniques utilisées dans les tests en ligne .....	6
2. l'évaluation automatique des réponses .....	7
3. Introduction brève aux logiciels utilisés .....	7
3.1. La plate forme Moodle .....	7
3.2. Hotpotatoes .....	7
3.3. La plate-forme «Claroline» .....	8
3.4. La plate-forme «eFront» .....	8
4. Les types de tests .....	8
4.1. Les tests basés sur les choix préétablis .....	9
4.1.1. Question à choix multiples .....	9
4.1.2. Question vrai ou faux .....	14
4.1.3. Question de correspondance .....	15
4.1.4. Question de type désordre .....	19
4.2. Réponses rédigées librement .....	21
4.2.1. Question à trous .....	21
4.2.2. Questions à réponses courtes .....	25
5. Evaluation de la correction automatique .....	28
5.1. Question à choix multiple .....	28
5.2. Question de correspondance .....	29

## Chapitre 2 : Les réponses librement rédigées.

1. La réponse courte dans « Moodle » .....	31
1.1. La question Cloze .....	31
1.2. La question à réponse courte .....	34
1.3. La question numérique .....	36
1.4 La question Calculée .....	38

1.5. La question « avec expressions régulières » .....	39
2. la réponse courte dans « HotPotatoes » .....	42
2.1. La question JCloze .....	42
2.2. La question « Quiz » .....	43
2.3. La question Hybride.....	44
3. la réponse courte dans « Claroline » .....	45
3.1. Texte à trous .....	45
4. La réponse courte dans « eFront » .....	46
4.1. Espaces vides .....	46
4.2. Texte libre .....	46
5. Evaluation des diverses approches pour les réponses.....	47
5.3. Les caractéristiques non considérés .....	49

### Chapitre 3 : Evaluation Flexible des réponses courte

1. Les objectifs .....	51
2. Détermination des éléments à considérer par l'enseignant.....	52
3. Architecture générale de l'outil à concevoir et réaliser .....	52
4. Conception et description des différentes fonctionnalités ajoutées .....	55
4.1 L'approche utilisée pour résoudre le problème des blancs .....	55
4.2 L'approche utilisée pour résoudre le problème des accents .....	56
4.3 L'approche utilisée pour résoudre le problème des doubles .....	57
4.4 L'approche utilisée pour résoudre le problème des déterminants.....	58
4.5 L'approche utilisée pour résoudre le problème des homophones .....	59
4.6 L'approche utilisée pour résoudre les problèmes des questions orientées vers un langage de programmation .....	60
4.7 L'approche utilisée pour réaliser des réponses acceptées à base de mots clés .....	60
4.8 L'approche utilisée pour réaliser des réponses acceptées à base de parties de mots clés ..	62
5. Les fonctions PHP utilisées.....	64

### Chapitre 4 : Réalisation et tests.

1. Introduction.....	66
2. Environnement de réalisation.....	66
3. Modules et plugins de « Moodle » .....	66
3.1. Définition d'un module ou plugin.....	66
3.2. Création d'un nouveau module Moodle .....	66

4. Réalisation des fonctionnalités du module .....	69
4.1 :L'option «Négliger les blancs» .....	69
4.2. L'option « Négliger les accents ».....	71
4.3. L'option « Négliger les caractères en doubles ».....	71
4.4. L'option« Négliger les déterminants » .....	72
4.5. L'option « Accepter les même homophones » .....	73
4.6. L'option « Orienter vers un langage de programmation » .....	75
4.7. L'option « Réponse à base des mots clés ».....	77
4.8. L'option «Réponse à base des partie de mots ».....	80
4.9. L'aide des options.....	82
5. Comparaison entre l'ancien et le nouveau système.....	82
Conclusion général.....	84
Annex.....	86
Bibliographie.....	88

## LISTE DES FIGURES

Figure 1.1 : schéma montrant le déroulement d'un test en ligne. ....	5
Figure 1.2 : Schéma générale de la correction automatique des questions à rédactions libres...7	7
Figure 1.3 : Exemple d'un QCU .....	9
Figure 1.4 : Exemple d'un QCM.....	10
Figure 1.5 : Exemple l'option QCM.....	11
Figure 1.6 : Exemple l'option multi-sélection.....	11
Figure 1.7 : Exemple QCU (Claroline).....	12
Figure 1.8 : Exemple QCM (Claroline).....	12
Figure 1.9 : Exemple d'un QCM (eFront).....	13
Figure 1.10 : Exemple d'un QCU corrigée (eFront).....	13
Figure 1.11 : Exemple vrai ou faux corrigée (Moodle).....	14
Figure.1.12 : Exemple vrai ou faux (Claroline).....	14
Figure.1.13 : Exemple vrai ou faux corrigée (eFront) .....	15
Figure.1.14 : Exemple appariement (Moodle).....	16
Figure.1.15 : Exemple appariement (une liste déroulante).....	17
Figure.1.16 : Exemple appariement (glisser-déposer) .....	17
Figure.1.17 : Exemple appariement (d'exercice des images) .....	17
Figure.1.18 : Exemple correspondance (Claroline).....	18
Figure.1.19 : Exemple correspondance avant et après (eFront).....	18
Figure.1.20 : Exemple Drag and drop (eFront).....	19
Figure.1.21 : Exemple JMix de type1 .....	19
Figure.1.22 : Exemple JMix de type2 .....	20
Figure.1.23 : Exemple1 JMix corrigé .....	20
Figure.1.24 : Exemple2 JMix corrigé .....	20
Figure.1.25 : Exemple JCross .....	21
Figure.1.26 : Exemple1 de Cloze corrigé.....	22
Figure.1.27 : Exemple de JCloze .....	23
Figure.1.28 : Exemple de JCloze corrigé1 .....	23
Figure.1.29 : Exemple de JCloze corrigé2 .....	23
Figure.1.30 : Exemple de JCloze avec menu déroulant.....	24
Figure.1.31 : Exemple de JCloze inclusion des mots.....	24

Figure.1.32 : Exemple dans le panneau de formateur.....	25
Figure.1.33 : Exemple de réponse courte (Moodle).....	26
Figure.1.34 : Exemple de la question Calculée.....	27
Figure.1.35 : Exemple de la question Quiz.....	27
Figure.1.36 : Exemple de la question texte libre.....	28
Figure.2.1 : Exemple1 de Cloze corrigé.....	33
Figure.2.2 : Exemple2 de Cloze corrigé.....	33
Figure.2.3 : Exemple3 de Cloze corrigé.....	33
Figure.2.4 : Exemple4 de Cloze corrigé.....	33
Figure.2.5 : Exemple1 de réponse courte corrigé .....	34
Figure.2.6 : Exemple2 de réponse courte corrigé .....	34
Figure.2.7 : Exemple3 de réponse courte corrigé .....	34
Figure.2.8 : Exemple4 de réponse courte corrigé .....	34
Figure.2.9 : Exemple5 de réponse courte corrigé .....	35
Figure.2.10 : Exemple6 de réponse courte corrigé .....	35
Figure.2.11 : Exemple «Numérique »dans l’interface de création.....	36
Figure.2.12 : Exemple1«Numérique » .....	37
Figure.2.13 :Exemple2«Numérique » .....	37
Figure.2.14 : Exemple3«Numérique ».....	37
Figure.2.15 : Exemple4«Numérique ».....	37
Figure.2.16 : Exemple «Calculée » .....	38
Figure.2.17 : Panneau de création de regexp .....	39
Figure.2.18 : Exemple 1 de regexp .....	40
Figure.2.19 : Exemple1 de JCloze corrigé.....	42
Figure.2.20 : Exemple2 de JCloze corrigé.....	42
Figure.2.21 : Exemple de JCloze avec menu déroulant.....	43
Figure.2.22 : Exemple de JCloze inclusion des mots.....	43
Figure.2.23 : Exemple1de Hybride .....	44
Figure.2.24 : Exemple2de Hybride .....	44
Figure.2.25 : Exemple de remplissage de zones de texte.....	45
Figure.2.26 : Exemple de liste déroulante.....	45
Figure.3.0 : Architecture du système. ....	54

Figure.3.1 : La fonction qui élimine les espaces.....	56
Figure.3.2 : La fonction qui élimine les accents .....	57
Figure.3.3 : La fonction qui élimine les doubles.....	58
Figure.3.4 : La fonction des déterminants.....	59
Figure.3.5 : une partie de la fonction des homophones.....	59
Figure.3.6 : la fonction qui traite les caractères spéciaux .....	60
Figure.3.7 : La fonction des mots clés .....	61
Figure.3.8 : la fonction qui élimine les symboles dans une chaîne.....	62
Figure.3.9 : les deux fonctions de vérification.....	63
Figure.3.10: la fonction de comparaison.....	63
Figure.4.1 : renommer les instances du module.....	67
Figure.4.2 : modification des tables .....	67
Figure.4.3 : succès d'installation.....	68
Figure.4.4 : l'ajout du module à la liste des modules.....	68
Figure.4.5 : liste des types de question .....	68
Figure.4.6 : schéma générale de la correction automatique dans le nouveau système .....	69
Figure.4.7 : création de question. ....	69
Figure.4.8 : choix de l'option «négliger les blancs » .....	70
Figure.4.9 : réponse de l'étudiant.....	70
Figure.4.10 : correction avec plusieurs blancs exemple1 .....	70
Figure.4.11 : correction avec plusieurs blancs exemple2. ....	71
Figure.4.12 : exemple de l'option « négliger les accents ».....	71
Figure.4.13 :exemple1 de l'option «Négliger les doubles » .....	72
Figure.4.14 : exemple 2 de l'option «Négliger les doubles » .....	72
Figure.4.15 : exemple de l'option « négliger les déterminants ».....	73
Figure.4.16 : choix de l'option « Accepter les même homophones ».....	73
Figure.4.17 : exemple de l'option « Accepter les même homophones » .....	73
Figure.4.18 : exemple de 'ke' au lieu 'que' .....	74
Figure.4.19 : exemple de 'o' au lieu 'eau'. ....	74
Figure.4.20 : exemple de 'wa' au lieu 'oi'. ....	74
Figure.4.21 : exemple de 'si' au lieu 'zi'. 'ge' se prononce 'je' mais pas 'gue' .....	75
Figure.4.22 : exemple de 'je' au lieu 'gue' .....	75
Figure.4.23 : Exemple d'une question et choix de l'option. ....	76
Figure.4.24 : Exemple de réponse de l'étudiant.....	76

Figure.4.25 : Exemple de réponse de l'étudiant avec réponse correcte.....	76
Figure.4.26 : Exemple1 de réponse de l'étudiant avec réponse fausse.....	77
Figure.4.27 : Exemple2 de réponse de l'étudiant avec réponse fausse.....	77
Figure.4.28 : Exemple de réponse de l'enseignant .....	78
Figure.4.29 : La saisit des mots clés .....	78
Figure.4.30 : Exemple de réponse de l'étudiant non corrigé. ....	79
Figure.4.31 : Exemple1 de réponse de l'étudiant corrigé .....	79
Figure.4.32 : Exemple2 de réponse de l'étudiant corrigé. ....	79
Figure.4.33 : choix de l'option et remplissage des champs .....	80
Figure.4.34 : Exemple1 avec l'option « Réponse à base des partie de mots» corrigé. ....	80
Figure.4.35 : Exemple2 avec l'option « Réponse à base des partie de mots» corrigé.....	81
Figure.4.36 : Exemple3 avec l'option « Réponse à base des partie de mots» corrigé. ....	81
Figure.4.37 : exemple d'aide.....	82

## LISTE DES TABLEAUX

Tableau 1.1- Comparaison entre les questions à Choix multiples. ....	28
Tableau 1.2- Comparaison entre les questions de correspondance.....	29
Tableau 2.1- Comparaison entre les questions à trous.....	47
Tableau 2.2- Comparaison entre les questions à réponses courte.....	48
Tableau 3.1-les fonctions PHP utilisées.....	64

---

---

000000\_000

# Introduction générale

---

---

000000

# Introduction générale

---

## ❖ Introduction :

Aujourd'hui, les systèmes de télé-enseignement semblent avoir atteint un bon degré de maturité qui permettrait de drainer un nombre de plus en plus important d'enseignants à l'utilisation de ces techniques pour d'une part la diffusion des ressources pédagogique, la communication plus efficace entre apprenant, enseignant et concepteur de cours et d'autre part le contrôle continu des étudiants.

Le contrôle continu à travers les tests en ligne semble être le point le plus attractif pour les enseignants. Cependant ce point ne semble pas avoir atteint un haut degré de maturité pour qu'il puisse devenir un facteur important d'attraction des enseignants à l'utilisation des plateformes de télé-enseignement. Au contraire, il semble être aujourd'hui le point le plus faible qui contribue plutôt au désintéressement d'un nombre important d'enseignant, surtout ceux qui ne trouvent pas, dans le contexte de leur matières enseignées, comment exprimer des questions dans le contexte des techniques traditionnelles telles que les diverses variantes de QCM et les contraintes importantes imposées aux étudiants dans le contexte des réponses trop courtes, réduites souvent à un mot ou une valeur.

## ❖ Problématique :

Les réponses courtes, ou l'étudiant est amené à écrire plusieurs mots semble être une technique qui permettrait d'évaluer de manière assez efficace l'acquisition des connaissances d'un cours, et permettre à l'enseignant de voir quels sont les points acquis et les points de difficulté pour les étudiants.

Cependant la situation actuelles des outils de correction automatique de réponse courte ne semble pas avoir atteint un haut degré de maturité et impose aux examinés des compétences et des contraintes importantes. Ainsi, ces outils ne permettent pas de traiter les faiblesses dans l'orthographe ni les erreurs d'inattention telle que la répétition erroné de lettre, l'oubli d'un accent, l'utilisation d'un mauvais déterminant etc.

# Introduction générale

---

## ❖ Objectifs:

L'objectif de notre travail consiste à étudier les diverses techniques de corrections automatique, notamment la correction de réponse courte et proposer une approche pour une correction automatique de réponse courte plus efficace que les solutions existante. L'idée principale de notre approche est de faire associer efficacement l'enseignant dans le processus de correction automatique. Cette association consiste à mettre à la disposition de l'enseignant diverses fonctionnalités qui lui permettront de guider à l'avance le processus de correction automatique. La conséquence directe de cette association est d'une part l'adaptation de la correction automatique au domaine (ou matière) concerné par l'évaluation et d'autre part ne nécessiterait pas de mettre en place des algorithmes très sophistiqué qui malgré leur sophistication ne donnerons pas un résultat satisfaisant.

Un autre objectif de taille, consiste à travailler dans le contexte de la communauté Moodle et ajouter à ce dernier un module. Ainsi, la maîtrise du code de ce dernier et de la méthode de mise en place et test de module est un travail très important que nous devons achever.

## ❖ Organisation du mémoire :

Pour réaliser ce projet de fin d'étude qui nous a été proposé, nous avons organisé notre mémoire en quatre chapitres répartis comme suit :

Le premier chapitre est consacré aux introductions et généralité sur la correction automatique, et présente aussi l'analyse et l'expérimentation des différents logiciels et plateformes pour voir comment ils traitent leurs réponses et les critiquer en plus on va faire une comparaison des traitements. Dans le deuxième chapitre nous avons introduit les détails d'évaluation automatique de la réponse rédigée librement avec un tableau récapitulatif des diverses caractéristiques des réponses courtes. Dans le troisième chapitre on explique le fonctionnement du système et les approches que nous avons suivies pour atteindre nos objectifs et les différentes méthodes et fonctions utilisées. Le dernier chapitre est consacré à l'implémentation et la vérification des résultats.

# CHAPITRE 1

---

## Etat de l'art sur la correction automatique

---

Ce chapitre est une étude théorique et expérimentale des différents logiciels et plateformes utilisant l'évaluation automatique des réponses dans un test en ligne. Dans cette étude nous allons évaluer ces diverses approches vis-à-vis de nos objectifs, et faire ressortir les points forts et faibles de chaque approche. Dans ce contexte nous avons étudié et expérimenté l'évaluation automatique dans le contexte des plateformes de Elearning et de logiciels dédiés uniquement aux tests. C'est ainsi que nous avons étudié et expérimenté les plateformes Moodle, Claroline, eFront, et le logiciel Hotpotatoes. Dans ce qui suit nous donnons un bref aperçu de chaque logiciel.

### 1.1 L'évaluation en ligne :

L'évaluation est une étape très importante dans la formation en générale, cela permet au formateur de juger le rendement de l'apprenant et le pourcentage d'acquisition des objectifs pédagogiques des connaissances.

Dans le télé-enseignement l'enseignant peut faire les évaluations en ligne en plusieurs types et formats, tout dépend de son but ou besoin, c'est ce qu'on appelle le test en ligne.

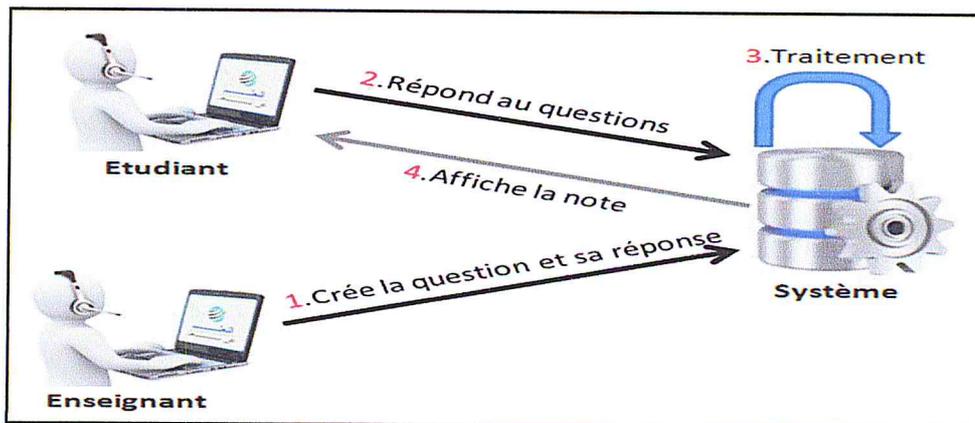


Figure 1.1 : schéma montrant le déroulement d'un test en ligne.

### 1.2. Les types des tests en lignes :

Traditionnellement, on distingue deux types d'évaluation, **formative** et **sommative** :

L'évaluation formative, est l'évaluation qui intervient au cours d'un apprentissage et vise à renseigner l'étudiant et l'enseignant de la progression de l'apprentissage et des difficultés rencontrées. Ce type d'évaluation peut facilement se mettre en place au moyen de plateformes d'enseignement en ligne tel que Moodle. Les plates-formes d'apprentissage incorporent de nombreux outils pour la création et la gestion de tests en ligne.

L'évaluation sommative, plus communément appelée **examen**, sert de bilan, de son résultat dépend la réussite du cours par l'apprenant. La pertinence pédagogique de la mise en place d'une évaluation sommative se pose dans les mêmes termes que pour l'évaluation formative ; la sécurité des données est cependant un enjeu majeur et doit être sérieusement considérée. [2]

### 1.3. Techniques utilisée dans les en lignes :

Les tests en ligne proposent divers formats de questions. On distingue les questions fermées (**réponses possibles strictement limitées**) et les questions ouvertes (**réponses rédigées librement**).

Si les questions **fermées** sont généralement automatiquement corrigées par le programme informatique, les questions ouvertes demandent souvent une intervention humaine dans le processus de correction. Pour construire son test, selon le logiciel utilisé, l'enseignant dispose d'un choix de possibilités :

- Texte à trous (avec ou sans indices).
- Question à choix multiples (une ou plusieurs réponses correctes, avec ou sans pénalité pour les réponses fausses).
- Vrai/faux.
- Question calculée (avec ou sans marge d'erreur).
- Listes à apparier.
- Réponse courte (avec modèle de réponse correcte pour auto-évaluation ou correction manuelle par l'enseignant).
- Enregistrement sonore de la réponse.
- Sélection d'une cible sur image.
- Sélection d'une portion de texte.
- Etc...[2]

**Remarque** : on va s'intéresser au « **réponses rédigées librement** » dans notre projet.

## 2. L'évaluation automatique des réponses:

Les tests en ligne sont évalués automatiquement par les logiciels, chaque logiciel de **Elearning** a sa propre manière de **correction automatique** tout dépend de ses algorithmes, et sa vision envers le test.

La figure suivante présente un schéma montrant le concept général de la correction automatique dans la plate-forme **Moodle** :

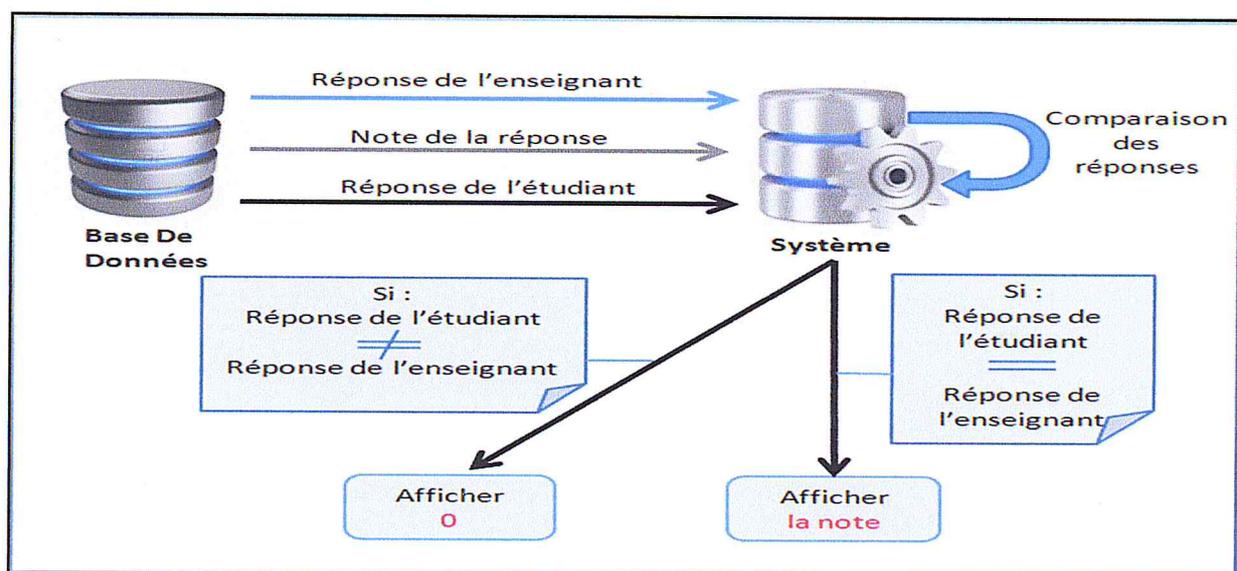


Figure 1.2 : Schéma générale de la correction automatique des questions à **rédactions libres**.

## 3. Introduction brève aux logiciels utilisés :

### 3.1. La plate-forme « MOODLE » :

Ce logiciel open source utilise les fonctionnalités de base d'une plate-forme de cours en ligne, il constitue un produit simple mais efficace. L'administrateur de la plate-forme peut s'occuper de tout: création des comptes utilisateurs, mise en ligne des cours et affectation des professeurs, modération des forums, analyse des logs contenant l'intégralité des actions effectuées au sein de la plateforme. [3]

### 3.2. «HotPotatoes» :

HotPotatoes est un créateur d'exercices qui jouit d'une très forte popularité dans le milieu de l'éducation. Il faut dire qu'il réussit la prouesse d'être une très grande richesse tout en restant simple d'emploi. Il est créé par l'équipe de recherche et développement du

Humanities Computing and Media Centre à l'Université de Victoria, Colombie-Britannique, Canada. Il permet à son utilisateur de créer cinq différents types d'exercices et de les publier sur internet. Il est permis d'utiliser gratuitement HotPotatoes pour une utilisation non-commerciale. Et Pour une utilisation commerciale il est payant. [4]

### 3.3. La plate-forme «Claroline»:

Est une plate-forme Open Source, distribuée sous licence GPL, qui permet à des centaines d'institutions issues de 93 pays de créer gratuitement des espaces cours en ligne. Pour chaque cours, le formateur dispose d'une série d'outils lui permettant d'élaborer des parcours pédagogiques, composer des exercices, structurer un agenda avec des tâches et des échéances, proposer des travaux à rendre en ligne... [5]

### 3.4. La plate-forme «eFront»:

**eFront** est une plate-forme de formation en ligne conçu pour aider à créer des communautés d'apprentissage en ligne avec des possibilités d'interaction. Il est livré avec une interface utilisateur basée sur des icônes distinctif qui est intuitif à utiliser. **eFront** est une plate-forme multilingue disponible en 40 langues. Plusieurs caractéristiques de la plate-forme (par exemple, la gestion, la structure organisationnelle, le rôle de superviseur des compétences), il est particulièrement adapté à l'utilisation de l'organisation interne, en particulier à la formation ou les services de gestion des ressources humaines. [6]

## 4. Les types de tests:

On peut initialement distinguer deux grandes catégories des tests en ligne :

### 1. Les tests basés sur le choix de réponses préétablies :

Un ensemble de réponses justes ou fausses sont présentés sous diverses formes à l'examiné. Celui-ci devra donc choisir la ou les bonnes réponses. Par cette technique le candidat examiné ne sera pas dans la contrainte d'une connaissance parfaite de la langue de réponse et sera à l'abri d'éventuelles erreurs d'inattention s'il était amené à écrire lui-même la réponse. Dans le contexte de cette catégorie nous retrouvons

1/ Question à Choix multiples.    3/ Question à correspondance.

2/ Question vrai ou faux.        4/ Le désordre.

## 2. Réponses rédigées librement :

Dans les réponses rédigées librement l'étudiant doit saisir la réponse. Il doit faire attention à bien veiller à respecter l'orthographe dans la langue utilisé pour la réponse et les styles imposés par la matière dans laquelle il est examiné. Dans cette catégorie nous retrouvons :

- 1/ Question à trous sans dictionnaire.
- 2/ Question à réponse courte.

### 4.1. Les tests basés sur le choix de réponses préétablies :

#### 4.1.1. Question à Choix multiples (QCM):

L'étudiant doit répondre à une question en choisissant une ou plusieurs réponses parmi un ensemble de réponses. Cet ensemble des réponses correctes et des réponses fausses. Les QCM peuvent être à réponse multiple ou à réponse unique. Toutes les plateformes de télé-enseignement que nous avons étudié supportent le QCM dans ses deux formes. Moodle offre en plus un QCM spécial appelé QCM à réponse calculée. Les questions calculées sont des questions numériques dont les nombres sont tirés aléatoirement d'un jeu de données lorsque le test est effectué. Dans ce qui suit nous allons étudier le QCM dans les plateformes Moodle, Claroline, eFront et dans le contexte du logiciel HotPotatoes

#### ➤ Les QCM dans Moodle :

##### 1. La question «à choix multiple» :

L'étudiant doit répondre à une question qui peut inclure une image à partir de choix multiples. Il existe deux types de questions à choix multiples - une réponse (QCU: unique réponse autorisée) ou - plusieurs réponses (QCM : réponses multiples autorisées).

Quelles est la raison principale à l'origine du développement du protocole IPV6?

Veuillez choisir une réponse :

- La simplification du format des en-tetes.
- l'extension des possibilités d'adressage. ✓
- La sécurité.
- la simplification de l'adressage.

Figure 1.3 : Exemple d'un QCU.

### 1. L'option QCM de JQUIZ :

L'option QCM permet de créer des questionnaires à choix multiple incluant éventuellement rétroactions, images, éléments multimédia, texte support...

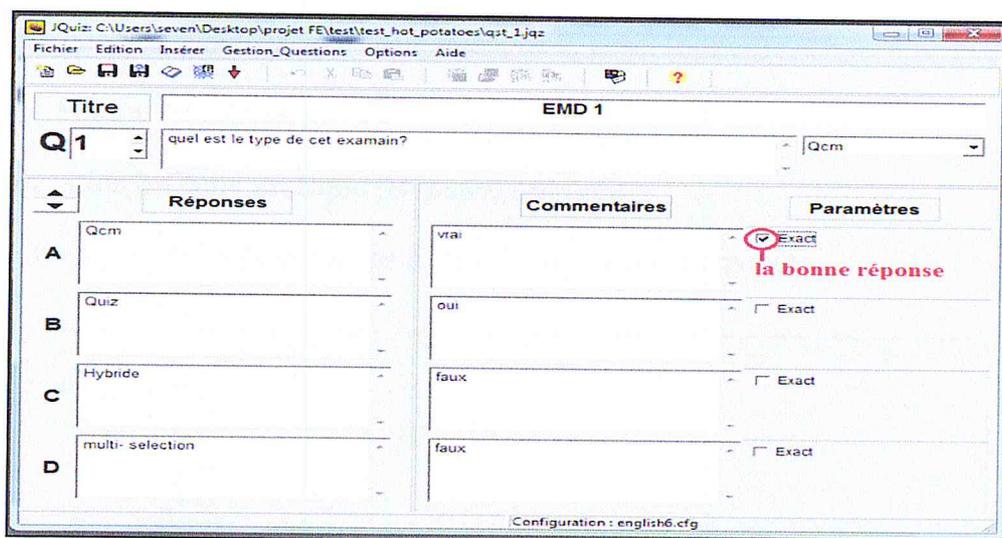


Figure 1.5 : Exemple l'option QCM.

- Si la bonne réponse se compose de plusieurs choix, et l'étudiant sélectionne un seul, le système lui donne une note complète.
- Si la réponse de l'étudiant est fausse, l'étudiant peut choisir de continuer jusqu'à l'obtention de la bonne réponse. Le score pour chaque question est basé sur le nombre de tentatives prise pour obtenir une réponse correcte.

### 2. L'option multi-sélection :

L'option multi-sélection permet de créer des exercices où plusieurs possibilités de réponses constituent la bonne réponse.

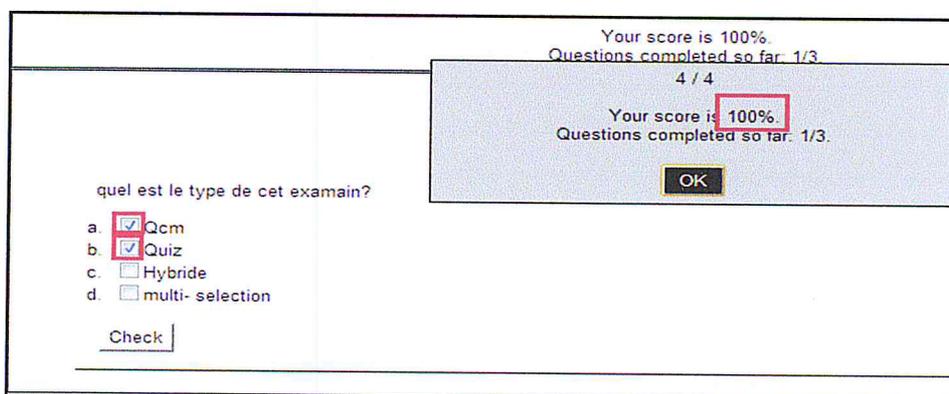


Figure 1.6 : Exemple l'option multi-sélection.

**Remarque :**

- Pour la notation, « **Claroline** » donne à la question la somme des notes attribuées pour chaque réponse attendue.

➤ **Les QCM dans eFront :****1. Choix multiple (Réponse unique):**

Il s'agit du classique QCM, où l'étudiant ne peut choisir qu'une seule réponse.

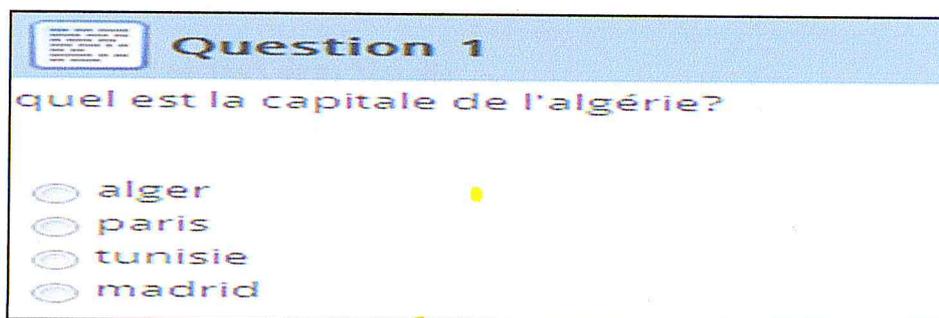


Figure 1.9 : Exemple d'un QCM (eFront).

Si l'étudiant sélectionne la bonne réponse il aura la note complète, sinon un 0.

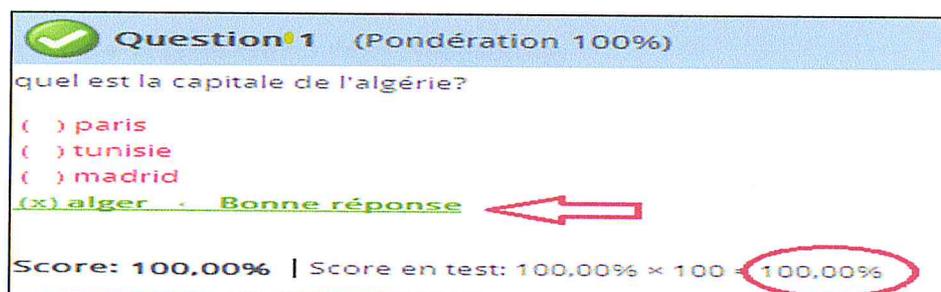


Figure 1.10 : Exemple d'un QCM corrigée (eFront).

**2. Choix multiple (réponses multiples) :**

Dans ce cas-ci, plusieurs réponses peuvent être correctes. efront pour ce type présente deux modes :

- **Le mode par défaut (ET) :** L'étudiant est obligé de donner toutes les réponses correctes, dans ce cas-ci, la note de l'exercice :

Note de chaque réponse = note globale (100%) /nbr de réponses correctes.

➤ La question « vrai ou faux » dans eFront :

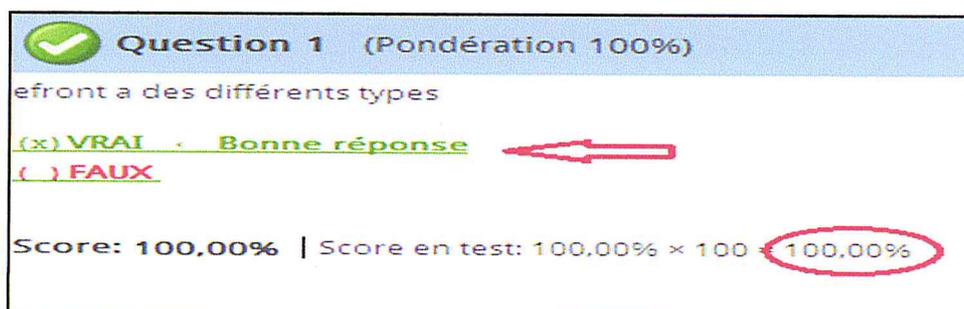


Figure 1.13 : Exemple vrai ou faux corrigée (eFront).

### 4.1.3. Question de correspondance:

L'étudiant est en face de deux listes d'éléments. Il doit établir la correspondance entre eux.

Selon la plateforme ou le logiciel, les questions par établissement de correspondance portent un nom particulier.

- **Moodle** : on trouve ce type dans Moodle sous le nom « **appariement** ».
- **Hotpotatoes** : on trouve ce type dans Hotpotatoes sous le nom « **JMatch** ».
- **Claroline** : on trouve ce type dans Claroline sous le nom « **correspondance** ».
- **Efront** : on trouve ce type dans Efront sous le nom « **correspondance** » et « **drag and Drop** ».

➤ La question de correspondance dans Moodle :

#### 1. La question « appariement »:

La question d'appariement demande au participant de faire correspondre une liste de termes (questions) avec une autre liste de termes (réponses).

Exemple :

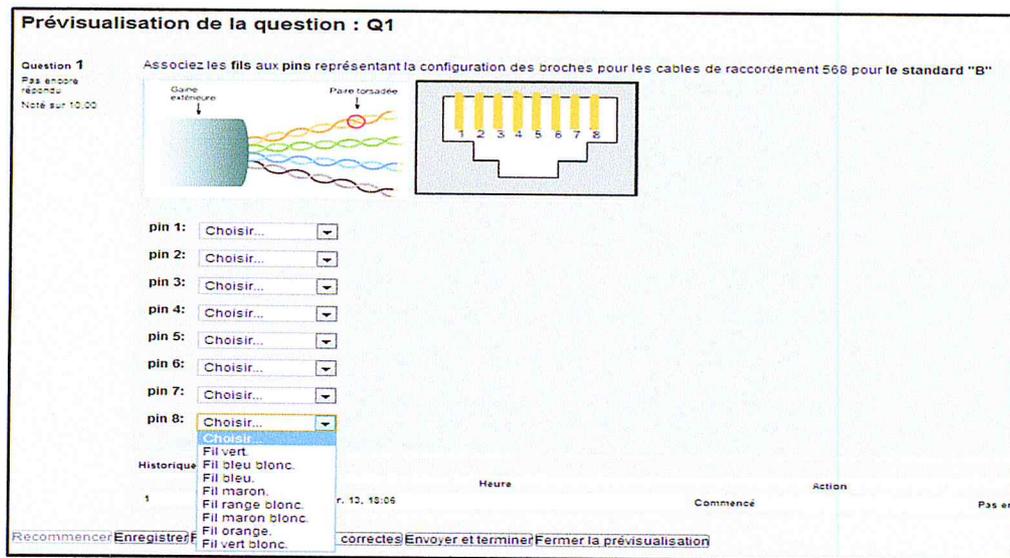


Figure 1.14 : Exemple appariement (Moodle).

**Remarques :**

- Désordre de sous questions après chaque tentative.
- Les réponses sont sous forme de listes, chaque liste contient les réponses des autres sous questions et d'autres réponses ajoutées.
- L'enseignant ne peut pas contrôler la liste des choix d'une réponse.
- On n'a pas une note pour chaque partie de la question, pour une fausse réponse d'une partie le système divise la note de la question globale sur le nombre de partie.
- Chaque sous-question (ou appariement) a un poids équivalent dans la note totale de la question. Par exemple, un étudiant qui apparie correctement 3 des 4 appariements possibles recevra une note de 3/4 ou 75 % de la note maximale pour cette question.
- Saisies répétées: si on répète des saisies à l'intérieur d'une liste, on doit nous assurer qu'elles soient vraiment identiques pour qu'elles n'apparaissent pas de multiples fois dans la liste, **par exemple** : « identifiez le type de ces créatures » les listes seraient : « fourmi, vache, chien, perroquet » et « insecte, mammifère, mammifère, oiseau » (cette dernière liste ne contiendra que 3 choix).

**2. Question Appariement aléatoire à réponses courtes:**

Ce type de question a été supprimé dans la version 2.1 et n'est donc disponible que dans la version 2.0. Du point de vue de l'étudiant, ce type de question ressemble en tout point à une question de type appariement. La différence entre les deux réside dans le fait que les sous-

questions sont tirées aléatoirement des questions de type réponse courte figurant dans la même catégorie de questions.

➤ **JMatch de HotPotatoes :**

Il est créé pour réaliser des exercices d'associations de 2 éléments. Et utilisé beaucoup plus pour les exercices simples.

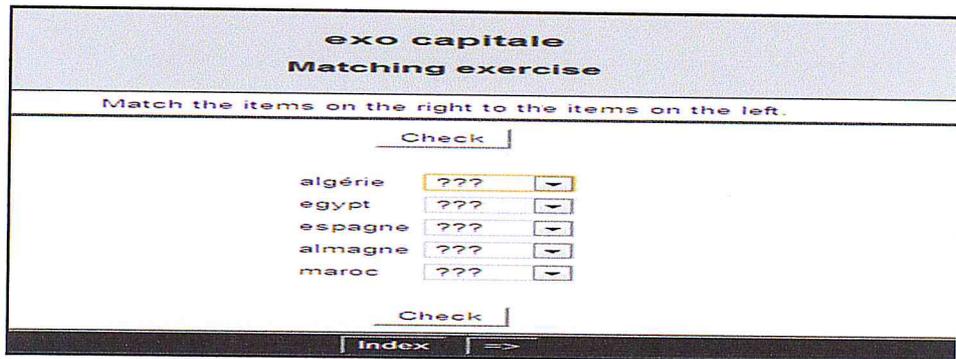


Figure 1.15 : Exemple **appariement** (une liste déroulante).

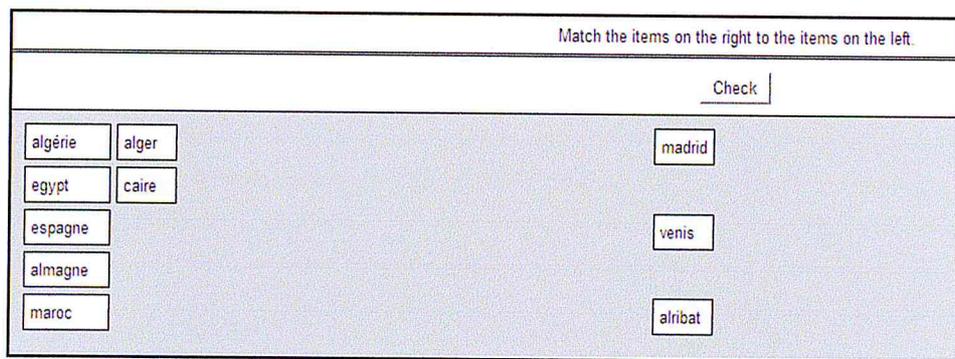


Figure 1.16 : Exemple **appariement** (glisser-déposer).

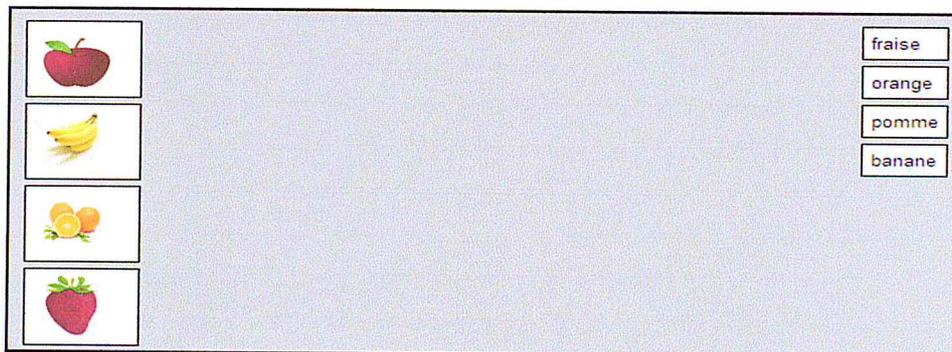


Figure 1.17 : Exemple **appariement** (d'exercice des images).

**Remarques :**

- Modifier l'ordre des propositions à chaque chargement de la page. (option)
- Limité le nombre de proposition à chaque chargement de la page. (option)

➤ **La question correspondance (Claroline) :**

L'étudiant devra faire correspondre les éléments d'une liste avec ceux d'une autre liste.

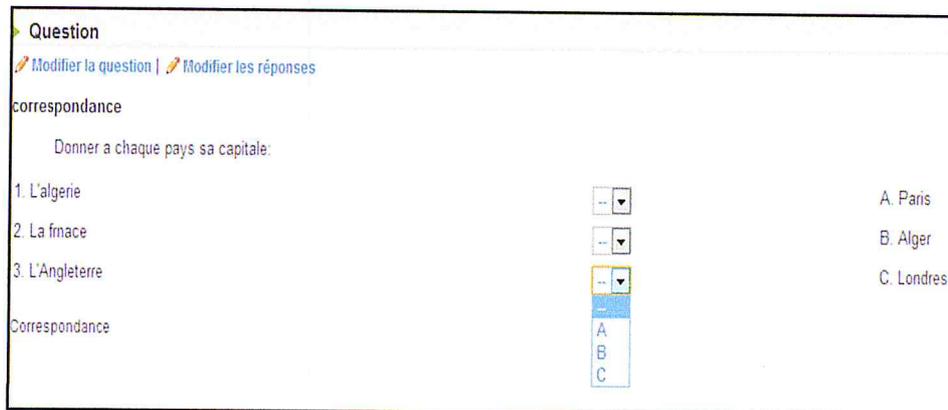


Figure 1.18: Exemple correspondance (Claroline).

**Remarque:**

Les éléments de la correspondance ont toujours le même ordre on ne pas leurs mettre en désordre.

➤ **Les correspondances de eFront :**

Dans eFront on trouve les questions à correspondance sous deux formes :

**1. Correspondances :**

Il est créé pour réaliser des exercices d'associations de 2 éléments ou les choix sont sous forme d'une liste déroulante :

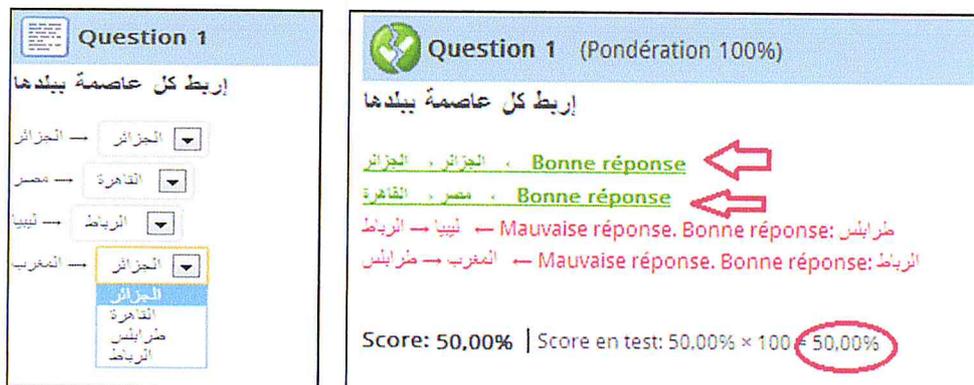


Figure 1.19 : Exemple correspondance avant et après (eFront)

**Drag and drop :**

La création du glisser-déposer et type de match de questions suit la même règle. La différence réside dans leur apparence pour les étudiants. Ce que nous avons à faire est de taper le texte de la question, puis insérer les paires.

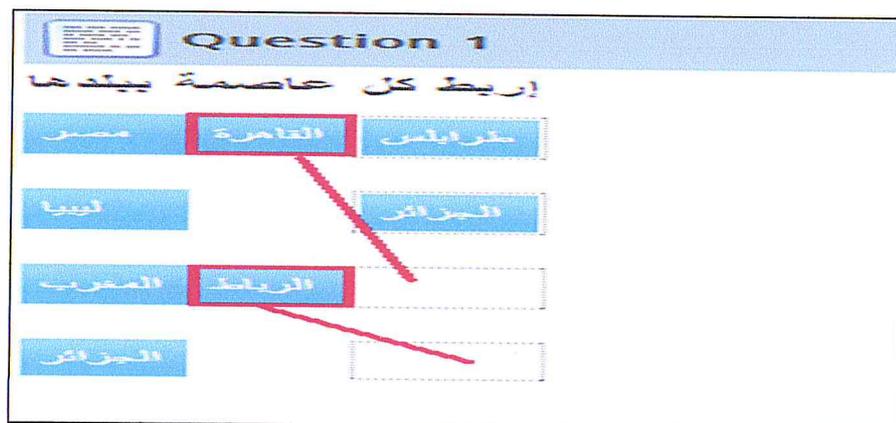


Figure 1.20 : Exemple Drag and drop (eFront).

**4.1.4. Les questions de type désordre :**

Dans ce type l'étudiant est censé de mettre en ordre un ensemble de mots ou de phrases, ce type se trouve **seulement** dans « Hotpotatoes » dans le contexte de « JMIX » et « JCROSS ».

**1. JMix :**

Permet de faire des exercices où l'apprenant doit remettre dans l'ordre des segments de phrases proposés dans le désordre. Les exercices ne peuvent porter que sur quelques courts segments et non sur des paragraphes ou unités plus grandes.

On peut afficher la page en deux types.

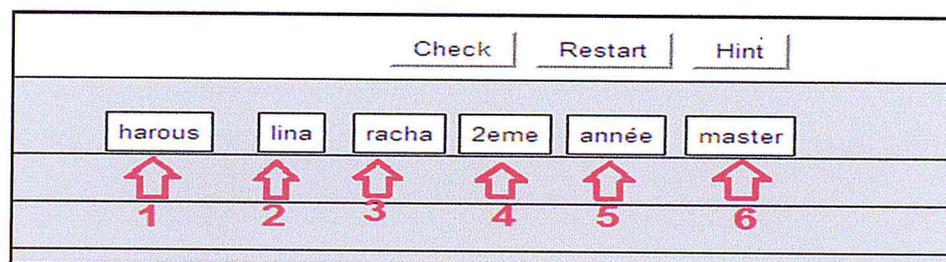


Figure 1.21 : Exemple JMix de type1.

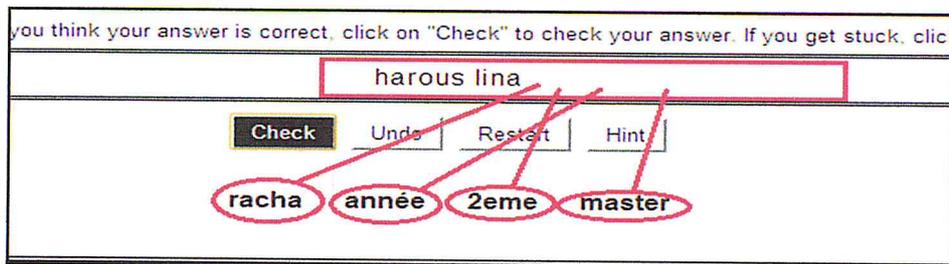


Figure 1.22 : Exemple JMix de type2.

- Si le début de la réponse est correct, et existe une erreur aux milieux ou à la fin, HotPotatoes ne donne pas 0, mais il insiste sur l'étudiant pour qu'il refaite sa réponse (il lui donne ou existe l'erreur au juste).

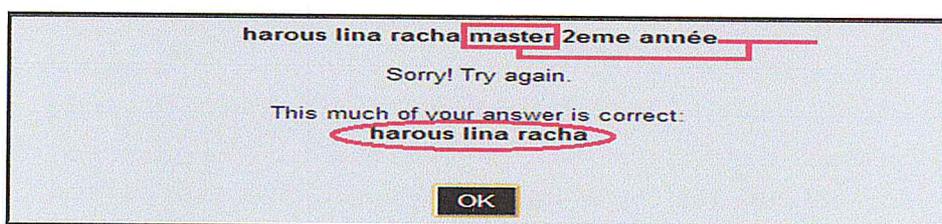


Figure 1.23 : Exemple1 JMix corrigé.

- Si le premier mot de la phrase est incorrect, HotPotatoes considère toute la phrase incorrecte.

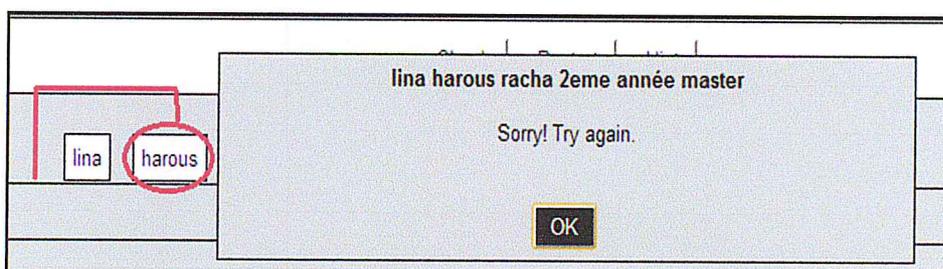


Figure 1.24 : Exemple2 JMix corrigé.

## 2. JCross :

Permet de créer des mots-croisés.

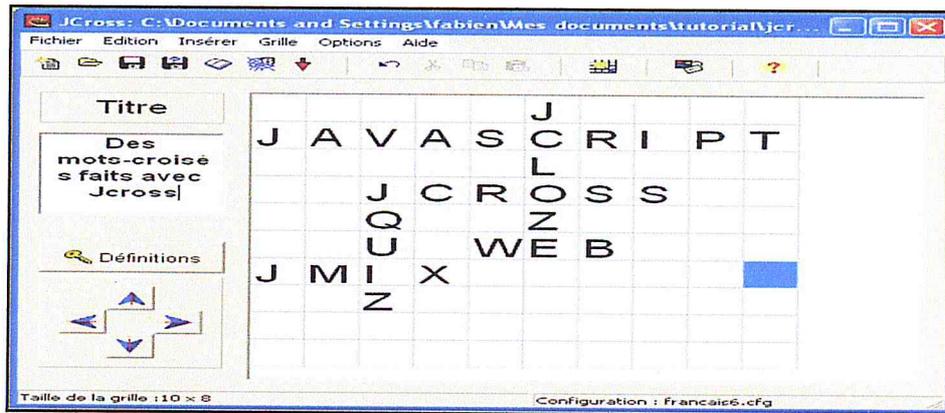


Figure 1.25 : Exemple JCross.

### Remarques:

- Si les mots entrés sont justes alors 100%. Sinon, le système donne un pourcentage qui dépend du nombre de lettres fausses.
- L'étudiant peut ré-exécuter et saisir de nouveau le mot exact.
- Le système pénalise l'étudiant pour chaque essai.

## 4.2. Réponses rédigées librement :

Dans ce type de questions, la réponse doit être explicitement écrite.

### 4.2.1. Question à trous :

L'étudiant est en face d'un texte contenant des emplacements vides et il est censé les remplir. Selon le logiciel ou la plateforme de télé-enseignement, ce type de question porte un nom particulier :

- **Moodle** : on trouve ce type dans Moodle sous le nom « Question Cloze ».
- **Hotpotatoes** : on trouve ce type dans Hotpotatoes sous le nom « JCloze ».
- **Claroline** : on trouve ce type dans Claroline sous le nom « Texte à trous ».
- **Efront** : on trouve ce type dans Efront sous le nom « Espaces vides ».

### ➤ La question « Cloze » de Moodle :

Ce type de question très flexible permet de proposer des textes à trous (en format Moodle) dans lesquels s'insèrent des réponses. Ces questions peuvent être de type **choix multiples**, **réponse courte** et **numérique**.

Les questions de ce type ne peuvent être créées qu'en tapant du texte suivant un format particulier avec des codes spécifiques.

Il n'existe à ce jour pas d'interface graphique pour créer de telles questions, il faut les écrire directement dans la boîte de texte, en format **Moodle** – suivant une **Syntaxe** et des règles de rédaction précises, en suite on teste si la syntaxe est correcte.

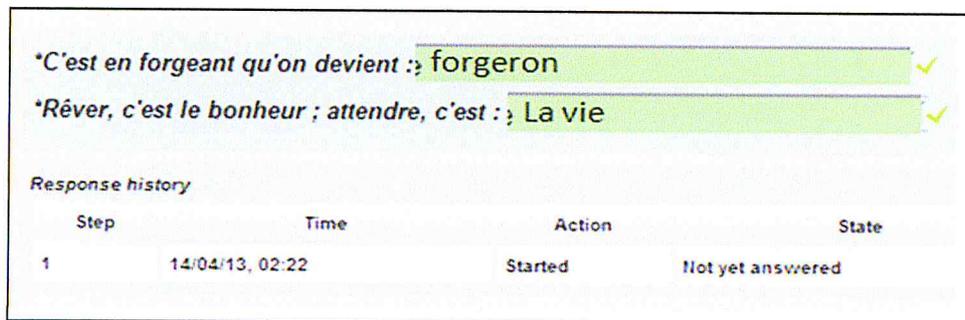


Figure 1.26 : Exemple1 de Cloze corrigé.

### ➤ Le JCloze de HotPotatoes :

Permet de créer des textes lacunaires, encore appelés exercices à trous ou tests de **Cloze**. L'utilisateur entre ses réponses dans les espaces vides, presse ensuite le bouton « Correction » pour découvrir ce qui est correct dans sa réponse et obtenir son score. Pour chaque mot manquant, d'autres réponses correctes peuvent être acceptées. Si l'utilisateur a besoin d'aide, il peut cliquer sur le bouton « indice » pour obtenir une lettre. [4]

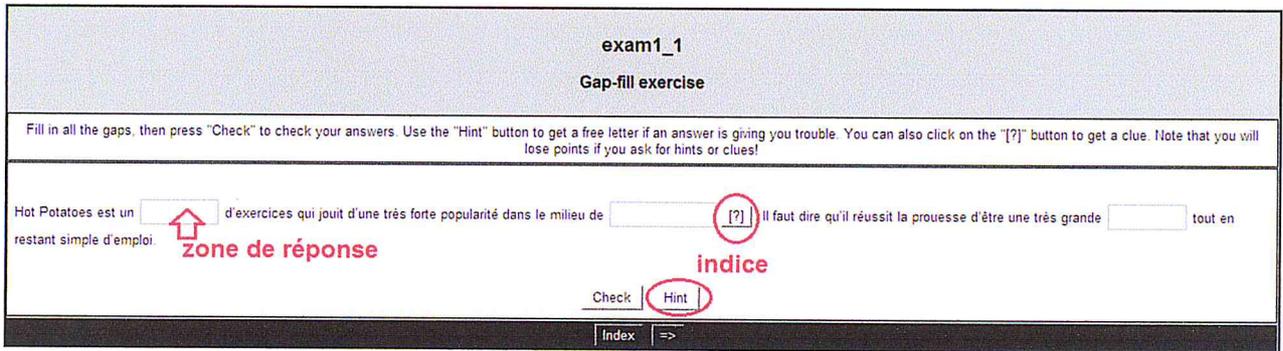


Figure 1.27 : Exemple de JCloze.

Remarques de correction automatique:

- Il ne prend pas en considération le nombre d'espaces.
- Il prend en considération toutes les réponses proposé, le reste est faut.
- Si l'étudiant écrit une réponse avec une lettre fausse alors il aura un 0 (même si un remplacement de « é » par « e », erreur de double frappe).

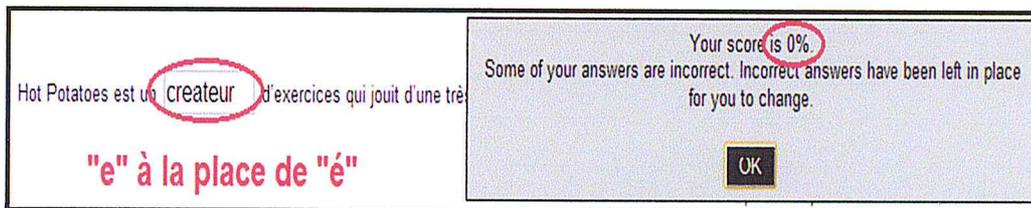


Figure 1.28 : Exemple de JCloze corrigé1.

- Cas de plusieurs trous dans le même exercice : si une réponse d'un trou est fausse, cela ne veut pas dire que toute la réponse de l'exercice est fausse (il n'aura pas un 0), mais HotPotatoes fait les statistiques et il calcule le pourcentage (%) de chaque trou, puis il soustrait la note du trou incorrect de la note globale.

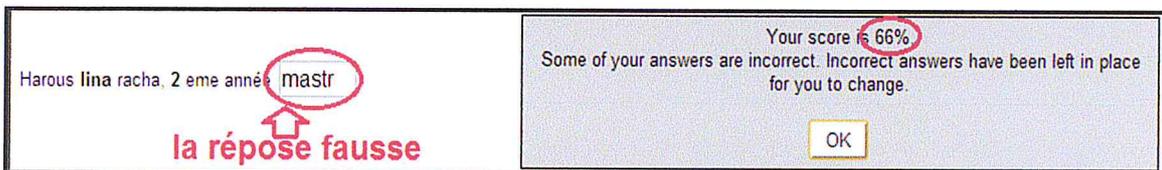


Figure 1.29 : Exemple de JCloze corrigé2.

- On peut utiliser les menus déroulants pour afficher une liste des mots contient tous les mots équivalent aux trous (et pas d'autre mots).

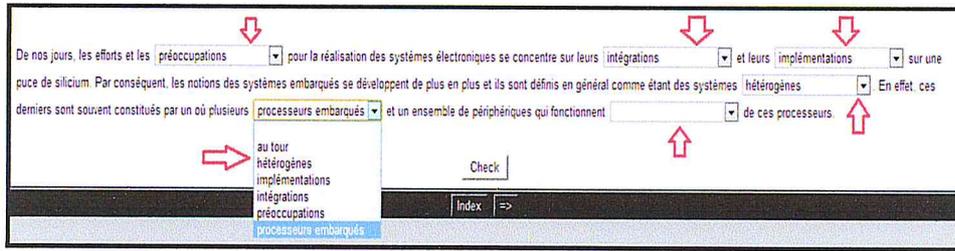


Figure 1.30 : Exemple de JCloze avec menu déroulant.

- Inclure une liste des mots avec le texte : pour aider l'étudiant de ne pas faire des fautes d'orthographe.

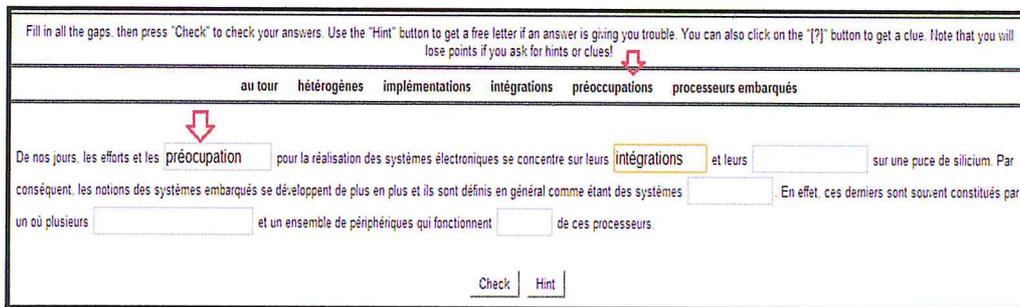


Figure 1.31 : Exemple de JCloze inclusion des mots.

- Respect des majuscules et minuscules.

➤ **Texte à trous de Claroline :**

Il s'agit du texte à trous. L'étudiant devra compléter un texte en trouvant les mots manquants. L'enseignant saisit le texte de question en utilisant les crochets [.....] pour les blancs.

➤ **Espaces vides de eFront :**

Il s'agit du texte qui contient des emplacements vides, le formateur le tape normalement dans la zone du texte, et à la place de chaque trou il insère ###, en suite un clique sur l'option "Créer des espaces vides" permet aux espaces d'apparaître dans une prochaine étape, on peut taper la bonne réponse dans les boîtes espace séparant plusieurs réponses correctes avec « | ».

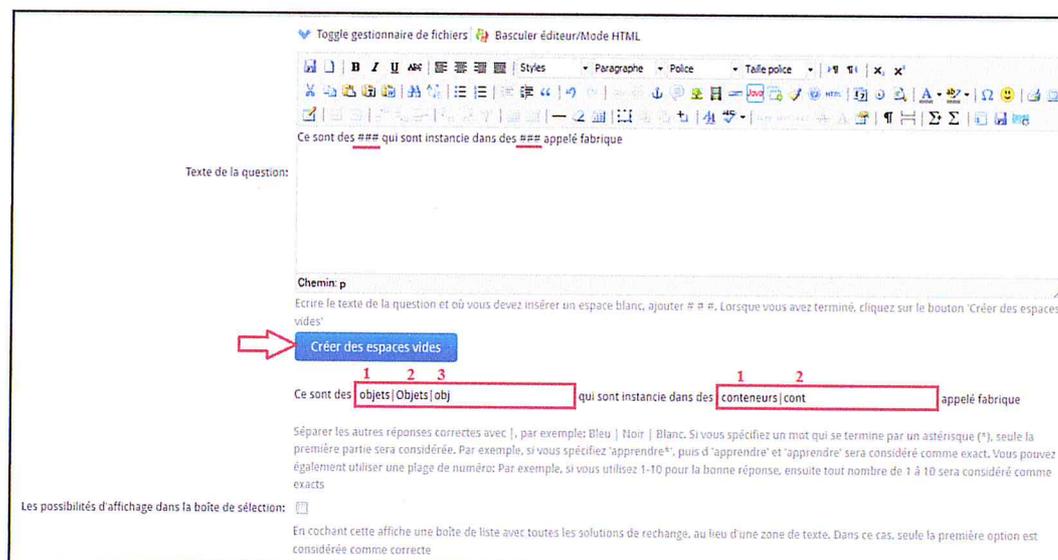


Figure 1.32 : Exemple dans le panneau de formateur.

#### 4.2.2. Les Question à réponse courte :

L'enseignant pose une question et l'étudiant répond par un ensemble de mots (un texte). Dépendant du logiciel ou de la plateforme de télé-enseignement, ce type de question porte un nom particulier

- **Moodle** : on trouve ce type dans Moodle sous le nom « **Question à réponse courte** », « **numérique** », « *Question Calculée* » et « *Question Calculée simple* ».
- **Hotpotatoes** : on trouve ce type dans Hotpotatoes sous le nom « **Quiz à réponse courte** » et « **hybride** ».
- **Claroline** : ce type n'existe pas dans Claroline.
- **Efront** : on trouve ce type dans Efront sous le nom « **Texte libre** ».

#### ➤ La question à réponse courte de « Moodle »:

##### 1. Les questions « à réponse courte »:

Permet une réponse d'un ou quelque mots, évaluée en comparant divers modèles de réponses, pouvant contenir des jokers.

Pour répondre à la question (qui peut contenir des médias), les participants doivent écrire un mot ou une phrase courte. Il peut y avoir plusieurs réponses correctes, chacune avec un pointage prédéterminé.

**Prévisualisation de la question : Q5**

Question 1  
Pas encore répondu  
Noté sur 1,00

Quelle est la capitale de l'Algérie ?

Réponse :

Historique des réponses

Étape	Heure	Ac
1	14 avr. 13, 14:07	Commencé

Figure 1.33 : Exemple de réponse courte (Moodle).

## 2. La question «Numérique :

Du point de vue de l'étudiant, une question numérique ressemble en tout point à une question de type réponse courte, le cas échéant avec des unités.

La différence entre les deux réside dans le fait que les réponses numériques permettent une marge d'erreur, donc un continuum de réponses possibles.

## 3. La Question Calculée:

Une question calculée est un moyen de créer des questions numériques et de choisir des **jokers** qui se substituent à des valeurs dans le test.

Cela permet de créer des questions numériques en utilisant des variables, placées entre deux accolades, qui sont remplacées par des valeurs lorsque le test est lancé.

Le but de la question calculée est de créer de multiples versions d'une même question avec des valeurs numériques différentes. Cela signifie que l'une des réponses doit contenir **au moins un joker (ou variable)**.

Quand un étudiant passe le test, **Moodle** choisira aléatoirement les valeurs pour les variables et évaluera la justesse de la réponse en utilisant la formule de calcul que l'enseignant a saisie. Le test a très peu de chance d'apparaître deux fois avec les mêmes valeurs.

Quelle est l'aire d'un rectangle de longueur 2.8 et de largeur 6.7 ?

Réponse :  ✓

La réponse correcte est : 18,76

Figure 1.34 : Exemple de la question **Calculée**.

#### 4. Les Question Calculée simples:

Version simplifiée des questions calculées (questions numériques dont les nombres sont tirés aléatoirement d'un jeu de données lorsque le test est effectué). Les **questions calculées simples** offrent les fonctionnalités les plus utilisées des questions calculées, mais avec une interface de création plus **simple**.

#### ➤ La question à réponse courte de « Hotpotatoes »:

##### 1. L'option Quiz de HotPotatoes :

Permet de créer un test à réponse courte. L'apprenant doit saisir la réponse au clavier.

après 3 essais

faux

Correct answers:  
Qcm  
Quiz ← la bonne rponse

Your score is 44%.

Questions completed so far: 1/3.

OK

quel est le type de cet examain?

quiz

Figure 1.35 : Exemple de la question **Quiz**.

##### 2. L'option Hybride de HotPotatoes :

Cette option combine le type de question Quiz (questions à réponse courte) avec le type de questions QCM. L'exercice se présente tout d'abord sous la forme d'un quiz, si après un certain nombre d'essais (2 par défaut), l'apprenant n'est toujours pas capable de saisir la réponse au clavier, la même question lui est présentée sous forme de QCM.

#### ➤ Textes libres de eFront :

Le professeur spécifie le type de correction, et les composants du texte attendu(les mots), en plus on peut donner un exemple de réponse aux étudiants.

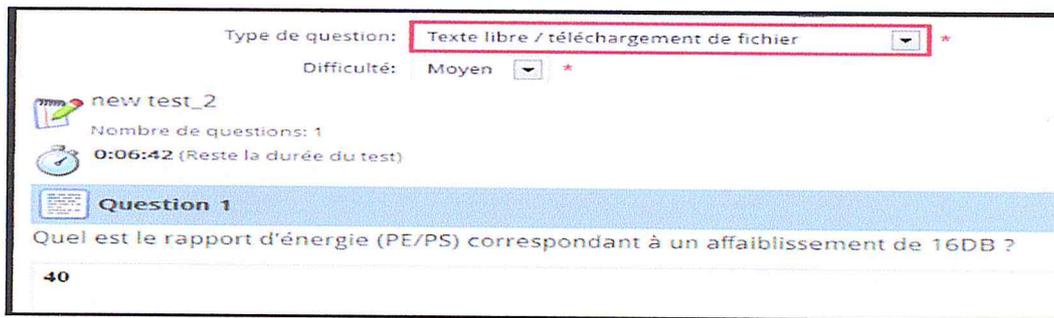


Figure 1.36 : Exemple de la question texte libre.

### 5. Evaluation de la correction automatique :

Dans cette section nous dressons des tableaux comparatifs entre les diverses approches de prise en charge de l'évaluation automatique des réponses dans les plateformes de Elearning et logiciels expérimentés.

#### 5.1. Question à Choix multiples :

Tableau 1.1- Comparaison entre les questions à Choix multiples.

Points de comparaison.	Moodle	Claroline	HotPotatoes	eFront
Mélanger les éléments.	*oui	*non	*oui	*non
Fixer un choix.	*non	*non	*oui	*non
Notation	*Un pourcentage pour chaque choix juste donné par l'enseignant.	*une note pour chaque choix.	*mode débutant : notation par défaut. *mode avancé : Un pourcentage pour chaque choix juste donné par l'enseignant.	*Il n'y a pas de note précisée mais un pourcentage par défaut.

## 5.2. Question de correspondance:

Tableau 1.2- Comparaison entre les questions de correspondance.

Points de comparaison.	Moodle	Claroline	HotPotatoes	eFront
Mélanger les éléments.	*oui	*non	*oui	*non
Fixer un choix.	*non	*non	*oui	*non
Existence de drag and drop.	*non	*non	*oui	*oui
Limiter le nombre de propositions affichées.	*non	*non	*oui	*non

## CHAPITRE 2

---

**Les réponses librement rédigées**

---

Dans ce chapitre on va introduire les détails d'évaluation automatique d'un type spécifique qui est la **réponse rédigée librement** dans les logiciels et plate-forme pour tester et expérimenter la façon de correction de ces réponses.

## 1. La réponse courte dans « Moodle » :

La réponse courte dans Moodle se présente sous les types suivants :

1. Question Cloze.
2. Question à réponse courte.
3. Question numérique.
4. Question Calculée.
5. Question avec expression régulière « regexp ».

### 1.1. La question Cloze :

Il n'existe à ce jour pas d'interface graphique pour créer de telles questions, il faut les écrire directement dans la boîte de texte, en format Moodle – suivant une **Syntaxe** et des règles de rédaction précises, en suite on teste si la syntaxe est correcte. Les différents éléments de la question décodés seront affichés et les erreurs de syntaxe seront mises en évidence si on clique sur le bouton « **décoder et vérifier le texte de la question** ».

**Remarque:** le texte de la question et les réponses attendues sont définies dans la même zone de texte.

#### **La syntaxe :**

##### **Voici les règles de syntaxe à respecter :**

- Au sein du texte, une réponse attendue (et ses paramètres) est toujours placée entre accolades { }
- Juste après l'accolade ouvrante, indiquer le nombre de points pour cette réponse (facultatif ; vous pouvez avoir {:})
- Ensuite indiquer le type de réponse attendue, encadré par des ":" :

**:SHORTANSWER:** -> Réponse courte indifférente à la casse.

**:SHORTANSWER\_C:** -> Réponse courte sensible à la casse.

**:NUMERICAL:** -> Réponse numérique.

Si le **plugin pour les questions de type Expressions régulières** est installé, on peut aussi indiquer le type de question : **REGEXP**:

- Ensuite arrivent les différentes propositions de réponses et leur feedback.

Le **feedback** est séparé de la proposition par un dièse #. Les groupes proposition+feedback sont séparés entre eux par une tilde ~.

Cela donne :

**proposition1#feedback1~proposition2#feedback2~proposition3~feedback3**

**N.B.** : en l'absence de feedback, le dièse # n'est pas nécessaire.

- Pour les réponses numériques, on peut définir une marge d'erreur acceptée dans la réponse. Ainsi, pour une réponse attendue de 5 plus ou moins 0,2 on met: **{:NUMERICAL:5:0.2#feedback...}**
- Pour les réponses courtes, on peut définir un feed-back par défaut pour les mauvaises réponses en utilisant le caractère joker \* afin de donner un même feedback à toute autres réponses que celles attendues. Par exemple : **{1:SHORTANSWER:=réponse attendue#bonne réponse~\*#feedback pour toute autre réponse}**
- Les notes sont indiquées en tête de la proposition considérée : = désigne la bonne réponse %50% désigne une réponse valant la moitié des points, on place %0% ou ne mettez rien pour désigner une mauvaise réponse. [7]

**Ainsi obtient-on par exemple :**

Dans le texte de la question:

**\*C'est en forgeant qu'on devient : {10:SHORTANSWER:=forgeron#bonne réponse~\*#feedback pour toute autre réponse}**

**\*Rêver, c'est le bonheur ; attendre, c'est : {10:SHORTANSWER:=la vie#bonne réponse~\*#feedback pour toute autre réponse}.**

**Remarques sur la correction automatique:**

On va supposer que la réponse correcte est «**réponse attendue** »:

**1** :L'étudiant saisit «**réponse attendue**» elle sera acceptée.

**2** :Ce type de test est **par défaut** insensible à la casse, si l'utilisateur saisit sa réponse en majuscule au lieu en minuscule (ou minuscules au lieu majuscules) elle sera acceptée comme

illustre la figure et l'enseignant peut le rendre sensible à la casse en insérant « \_C » dans le texte de la question:

\*C'est en forgeant qu'on devient: {10:SHORTANSWER\_C:=forgeron#bonne...

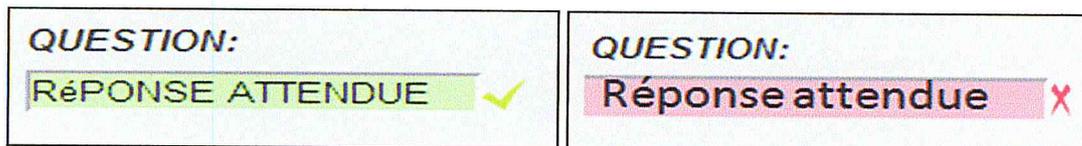


Figure 2.1 : Exemple1 de Cloze corrigé.

3 : La bonne réponse pourra être précédée ou suivie par un nombre illimité des espaces.

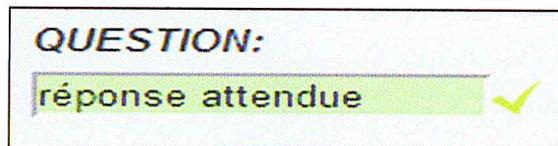


Figure 2.2 : Exemple2 de Cloze corrigé.

4 : On ne peut pas ajouter des espaces entre les termes de labonne réponse.

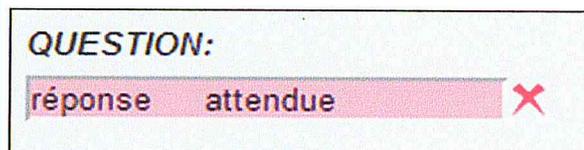


Figure 2.3 : Exemple3 de Cloze corrigé.

5 : On ne doit pas remplacer le « é » par « e » ou par « E ».

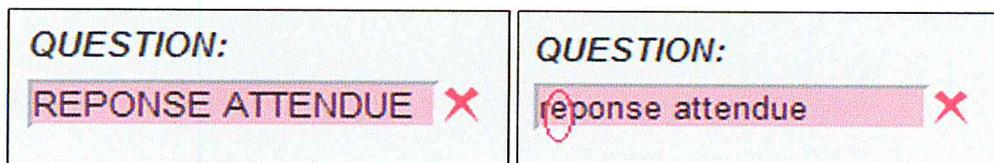


Figure 2. 4 : Exemple4 de Cloze corrigé.

1.2. La question à réponse courte :

**Remarques de correction automatique:**

On va supposer que la réponse correcte est «Alger»:

1 : L'étudiant saisit «Alger».

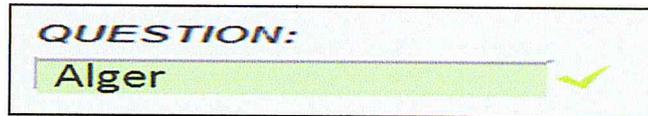


Figure 2.5 : Exemple1 de réponse courte corrigé.

2 : Les réponses peuvent ou non être sensibles à la casse :

Si l'option « Sensible à la casse » est activée:

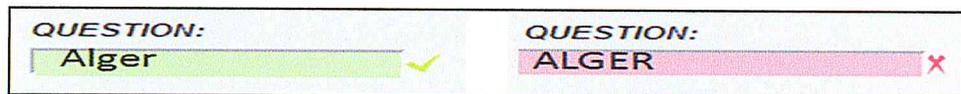


Figure 2.6 : Exemple2 de réponse courte corrigé.

Si l'option « Sensible à la casse » n'est pas activée:

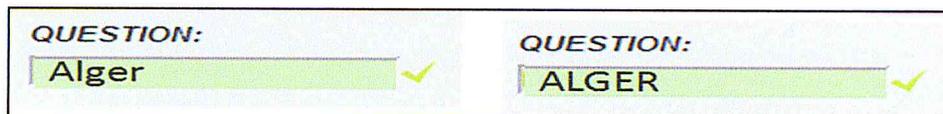


Figure 2.7 : Exemple3 de réponse courte corrigé.

3 : La bonne réponse pourra être précédée ou suivie par un nombre illimité des espaces.

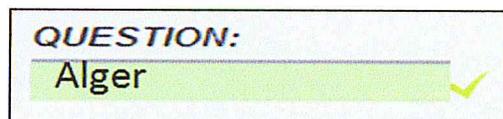


Figure 2.8 : Exemple4 de réponse courte corrigé.

4 : On ne peut pas ajouter des espaces entre les termes de la bonne réponse. Ici on suppose que la bonne réponse est « réponse attendue ».

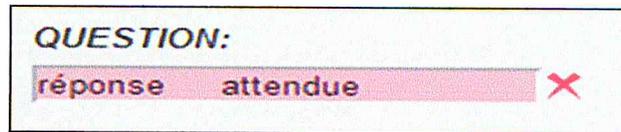


Figure 2.9 : Exemple 5 de réponse courte corrigé.

5 : On ne doit pas remplacer le « é » par « e » ou par « E ».

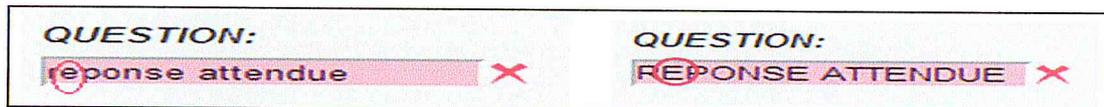


Figure 2.10 : Exemple 6 de réponse courte corrigé.

6 : Il est recommandé d'utiliser le **joker** \* pour la dernière réponse et de saisir une rétroaction et une note pour toute autre réponse donnée par les étudiants.

#### Utilisation du joker:

On peut utiliser « \* » comme joker pour indiquer que n'importe quel caractère est accepté à cet endroit.

**Par exemple:** in\*tion pour accepter tout mot ou toute proposition commençant par in et se terminant par tion.

Si nous voulons que la réponse soit vraiment un astérisque, nous devons utiliser la barre oblique comme ceci : \\*

**Exemple:** « Que consomme une fusée ? ».

Et la bonne réponse est: carburant et oxygène.

L'enseignant définit les réponses comme suit:

- oxygène\*carburant → note : 100 %.
- \*carburant\* → note : 50 %.
- \*oxygène\* → note : 50 %.
- \*air\* → note : 40 %.
- \* → note : 0 %.

✓ Cela acceptera les réponses « carburant oxygène », « carburant, oxygène », « carburant; oxygène », « carburant et oxygène », « carburant & oxygène », « carburant/oxygène », « carburant and oxygène », « carburant&oxygène ».

- ✓ Toutefois, cela acceptera aussi les réponses «**carburant ou oxygène**», «**carburant sans oxygène**» et «**carburant|oxygène** »,ce qui pourrait être moins bon, mais ce n'est pas un système absolument sans risques!
- ✓ L'ordre des réponses est important. Les réponses sont évaluées de la première à la dernière. Lorsqu'une réponse correspondante est trouvée, le processus s'arrête. Si aucune correspondance n'est trouvée, la note est de 0 et la rétroaction générale est utilisée.
- ✓ Il est bon de mettre simplement un joker comme **toute dernière réponse** pour que le processus d'évaluation sache quoi faire si aucune correspondance n'est trouvée.
- ✓ Sans joker, les réponses sont comparées exactement, donc il faut être prudents avec l'orthographe!

### 1.3. La question numérique :

Par exemple, si la réponse est 10 avec une marge d'erreur de 2, Alors, n'importe quel nombre entre 8 et 12 sera considéré comme correct.

Figure 2.11 : Exemple «Numérique »dans l'interface de création.

La réponse : 8, et erreur acceptée : 1 donc toute valeur comprise entre 7 et 9 sont acceptées.

#### Remarques :

- On peut y ajouter plusieurs réponses, et on attribue à un a chacune un pourcentage de la note.
- Si on saisit **n'importe quoi** après la bonne réponse elle sera considérée comme correcte, mais avant la bonne réponse **rien n'est acceptée sauf les espaces**.

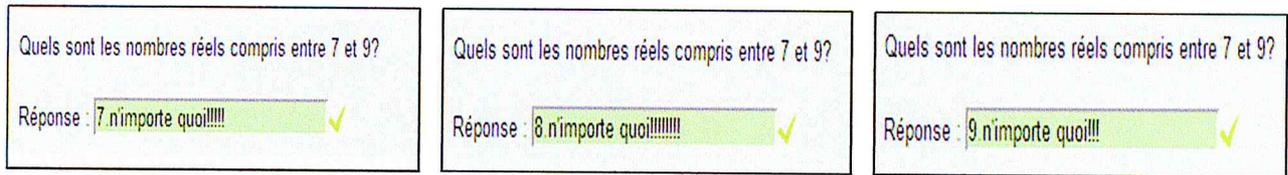


Figure 2.12 : Exemple1«Numérique ».

- Cette réponse est fausse car on a dépassé la marge.

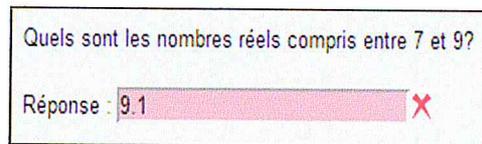


Figure 2.13 : Exemple2«Numérique ».

- Puisque les valeurs acceptées sont comprises entre 7 et 9, si on saisit 6.9990.... avec 13 chiffres (9) après la virgule la réponse ne sera pas correcte.

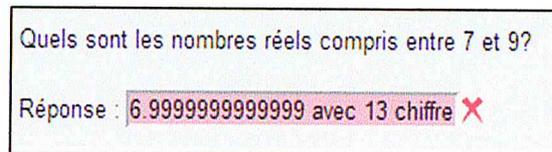


Figure 2.14 : Exemple3«Numérique ».

- Si l'étudiant saisit 6.9999.... avec 14 chiffres (9) après la virgule la réponse sera correcte.

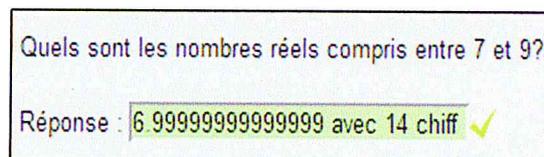


Figure 2.15 : Exemple4«Numérique ».

### 1.4. La question Calculée :

Quand un étudiant passe le test, **Moodle** choisira aléatoirement les valeurs pour les variables et évaluera la justesse de la réponse en utilisant la formule de calcul que l'enseignant a saisie. Le test a très peu de chance d'apparaître deux fois avec les mêmes valeurs.

- On peut mettre « , » ou « . ».
- Si on met deux points au lieu un seul, la réponse est fautive.

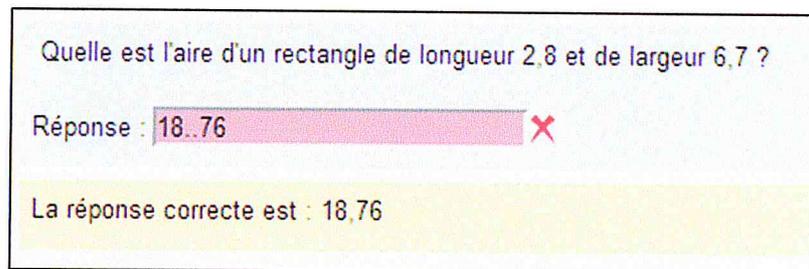


Figure 2.16 : Exemple «Calculée ».

- Si on met des espaces avant ou après la réponse correcte, la note est complète.

#### Jokers et jeux de données :

- Lorsque **Moodle** propose une question calculée à l'étudiant, les jokers sont remplacés par des valeurs aléatoires. Toutefois, ces valeurs ne sont pas complètement aléatoires - elles sont plutôt sélectionnées aléatoirement parmi un jeu de données prédéfini correspondant à des valeurs possibles.
- Ces jeux de données peuvent être privés ou partagés les jeux de données privés sont utilisés par un joker à l'intérieur d'une question calculée; les jeux de données partagés sont utilisés par un joker à l'intérieur de toutes les questions calculées qui l'utilisent.
- Le jeu de données : si on choisit plusieurs valeurs pour les variables on aura un changement à chaque tentative.
- Le test a très peu de chance d'apparaître deux fois avec les mêmes valeurs car **Moodle** choisit aléatoirement les valeurs pour les variables et évalue la justesse de la réponse en utilisant la formule de calcul saisie.

### 1.5. La question « avec expressions régulières » :

#### Définition :

La question à réponse courte « avec expressions régulières » ou « **regexp** » est un **plugin** additionnel de Moodle, qui permet de créer des questions pour l'activité Test.

Tout comme la question à réponse courte, la question à réponse courte de type "**Expression régulière**" attend de la part de l'élève une réponse à une question "**ouverte**", consistant en un ou plusieurs mots ou une courte phrase. Cependant, l'utilisation des expressions régulières nous donne accès à un puissant système d'analyse de la réponse des élèves.

La figure suivante montre le panneau de création de la question **regexp** :

Casse des caractères : La casse des caractères indiffère

Montrer les réponses alternatives à l'étudiant : Non

Note.- Réponse 1 doit être correcte (Note = 100%) et ne sera pas analysée en tant qu'expression régulière.

**Réponse 1**

Réponse : l'enseignant,l'étudiant,l'administrateur

Note : 100%

Feedback : [Rich text editor with toolbar]

Chemin: p

**Réponse 2**

Réponse : l'enseignant(,) l'étudiant(,et) l'administrateur

Note : 100%

Feedback : [Rich text editor with toolbar]

Figure 2.17 : Panneau de création de **regexp**.

**Exemple 3 :**(en anglais encore)

**Question :** "Donnez en anglais le nom d'un animal qui fait 3 lettres et dont la lettre du milieu est la voyelle a".

**Réponse :** "[bcr]at". Ce qui acceptera les 3 réponses : **bat**, **cat** et **rat**.<sup>[7]</sup>

- Dans la syntaxe des expressions régulières, l'inclusion de caractères entre crochets carrés signifie que n'importe lequel de ces caractères peut être utilisé pour l'évaluation de l'expression. Ainsi, dans l'exemple ci-dessus, l'expression "[bcr]at" est l'équivalent exact de "(b|c|r)at". Il faut faire bien attention de ne pas utiliser le caractère séparateur | dans une expression régulière entre crochets [...]. **Par exemple** l'expression "[b|c|r]at" ne fonctionnera pas comme attendu.
- La première réponse attendue correcte doit être sous forme standard sans expression régulière et doit avoir la note 100%.
- Pour échapper les caractères spéciaux : Si la réponses prévues à besoin de caractères qui font partie de la liste des caractères spéciaux utilisés dans la syntaxe des expressions régulières, on doit les "échapper", et les faire précéder d'une barre oblique inverse. Par exemple si la réponse est "Cet ordinateur vaut 1 000\$", on écrit ainsi l'expression régulière: "Cet ordinateur vaut 1 000\\$". Voici la liste complète des caractères spéciaux qui doivent être échappés : ^ \$ \* ( ) [ ] + ? | { } \ /
- Dans **Moodle 1.9**, par défaut une expression régulière est sensible à la casse (minuscules/majuscules). Pour rendre l'expression insensible à la casse, on ajoute le paramètre/i tout à la fin de l'expression.
- Dans **Moodle 2.0 et les versions suivantes**, on a en mode édition de la question le paramètre **Casse des caractères**, qui s'applique à toutes les réponses (expressions régulières) de la question en cours. On ne doit plus ajouter le paramètre /i à la fin de vos expressions régulières.

## 2. la réponse courte dans « HotPotatoes » :

La réponse courte dans **Hotpotatoes** se présente sous les types suivants :

1. JCloze.
2. Quiz.
3. Hybride.

### 2.1. La question JCloze:

- Il ne prend pas en considération le nombre d'espaces.
- Il prend en considération toutes les réponses proposé, **le reste est faut.**
- Si l'étudiant écrit une réponse avec une lettre fausse alors il aura un 0 (même si un remplacement de « é » par « e », erreur de double frappe).

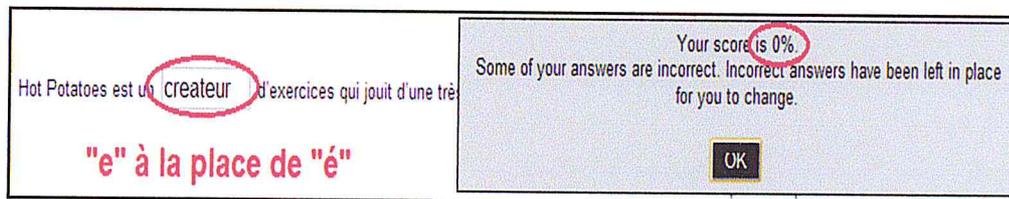


Figure 2.19 : Exemple1 de JCloze corrigé.

- Cas de plusieurs trous dans le même exercice : si une réponse d'un trou est fausse, cela ne veut pas dire que toute la réponse de l'exercice est fausse (il n'aura pas un 0), mais **Hot Potatoes** fait les statistiques et il calcule le pourcentage (%) de chaque trou, puis il soustrait la note du trou incorrect de la note globale.

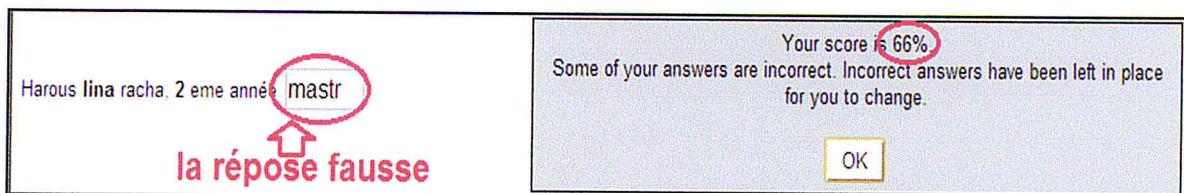


Figure 2. 20 : Exemple2 de JCloze corrigé.

- On peut utiliser les menus déroulants pour afficher une liste des mots contient tous les mots équivalent aux trous (et pas d'autre mots).

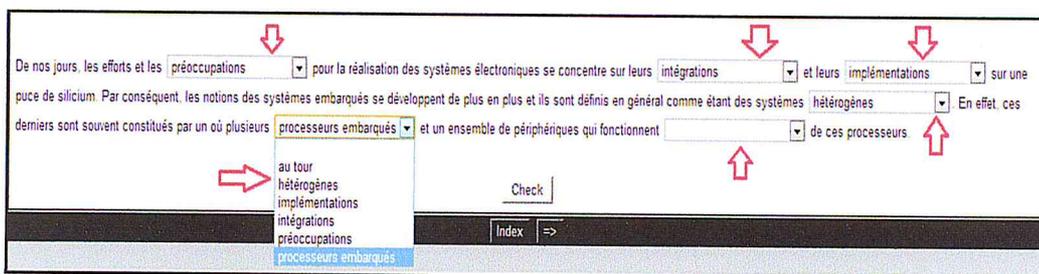


Figure 2.21 : Exemple de JCloze avec menu déroulant.

- Inclure une liste des mots avec le texte : pour aider l'étudiant de ne pas faire des fautes d'orthographe.

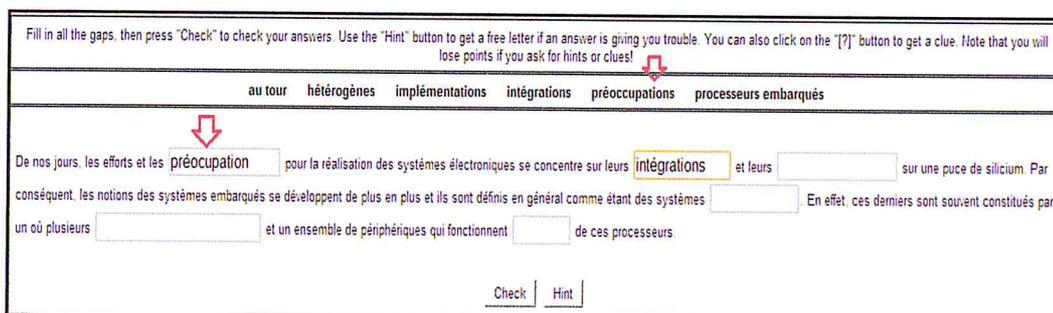


Figure 2.22 : Exemple de JCloze inclusion des mots.

- Respect des majuscules et minuscules.
- On peut limiter la durée de chaque exercice : à la fin du temps, automatiquement le system donne la note de la réponse actuelle.

## 2.2. La question « Quiz »:

- Si l'étudiant saisi la bonne réponse il obtient la note complète, sinon le système affiche qu'elle incorrecte jusqu'à l'obtention de la bonne réponse. Le score est basé sur le nombre de tentatives de l'étudiant avant d'obtenir la bonne réponse.

- Il existe le bouton « **Hint** » qui donnera à l'étudiant une lettre de la réponse pour l'aider, mais après utilisation de ce bouton l'étudiant sera pénalisé.

- **Le mode avancé : JQuiz** peut fonctionner de deux façons, en mode débutant ou avancé. Le mode avancé permet d'introduire la pondération des questions et des réponses, en d'autres termes, une question peut être de moindre importance au sein de l'exercice qu'une

autre, une réponse peut être partiellement juste. Le mode avancé permettra de définir les poids attribués à chaque question et possibilité de réponse.[8]

### 2.3. La question Hybride:

- A chaque fois que l'étudiant donne une réponse incorrecte contient une lettre existe dans la réponse de l'enseignant, le system lui donne certain pourcentage pour cette lettre.
- Quand l'apprenant entre une réponse fausse « hybrid » au lieu « hybride » : le système lui indique la partie correcte (1) et la partir incorrecte dans sa réponse (2).

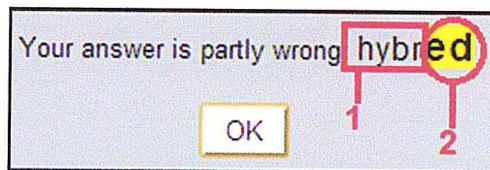


Figure 2.23 : Exemple1de **Hybride**.

- Même s'il manque une lettre dans la réponse, il lui précise l'emplacement.

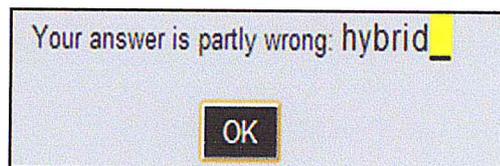


Figure 2.24 : Exemple2de **Hybride**.

- Modifier le nombre d'essais autorisés avant de basculer du type « **Hybride** » vers « Qcm ».
- Respect des majuscules et minuscules.
- **Le mode avancé : JQuiz** peut fonctionner de deux façons, en mode débutant ou avancé. Le mode avancé permet d'introduire la pondération des questions et des réponses, en d'autres termes, une question peut être de moindre importance au sein de l'exercice qu'une autre, une réponse peut être partiellement juste. Le mode avancé permettra de définir les poids attribués à chaque question et possibilité de réponse.[8]

### 3. la réponse courte dans « Claroline » :

La réponse courte dans **Claroline** se présente sous un seul : **Texte à trous** .

#### 3.1. Texte à trous :

- L'étudiant peut répondre par les majuscules au lieu minuscules (ou minuscules au lieu majuscules).
- Les espaces ne sont plus acceptés ni avant ni au milieu ni après la réponse.
- On ne doit pas remplacer le « é » par « e » ou par « E » ni par « è ».
- Pour chaque trou on ne peut attribuer qu'une seule bonne réponse.

On distingue deux sous types :

**1/Remplir les zones de texte:** L'étudiant remplit les blancs.

Figure 2.25 : Exemple de remplissage de zones de texte.

#### 2/Choisir dans une liste déroulante :

On attribue à chaque trou une liste qui contient les réponses attendues des autres trous, dans ce type l'étudiant ne saisit pas la réponse, il doit juste choisir un élément de la liste.

Figure 2.26 : Exemple de listes déroulante.

- On peut aussi ajouter des mauvaises réponses dans la liste déroulante, en mettant une réponse par ligne.

**Remarque:** L'ajout des mauvaises réponses donne plus de choix.

#### 4. La réponse courte dans « eFront » :

La réponse courte dans eFront se présente sous les types suivants :

1. Espaces vides.
2. Texte libre.

##### 4.1. Espaces vides:

- Il accepte plus qu'un espace entre les mots.
- L'étudiant peut répondre par les majuscules au lieu minuscules (ou minuscules au lieu majuscules) alors l'insensibilité à la casse est activé par défaut.
- On ne doit pas remplacer le « é » par « e » ou par « E » ni par « è ».
- Si l'enseignant coche la case « **les possibilités d'affichage dans la boîte de sélection** », et il saisie quelques choix séparés avec un « | », on trouve une liste déroulante qui s'affiche dans l'emplacement vide, sinon les cases s'afficheront vides.

##### 4.2. Texte libre:

- L'étudiant sépare entre les mots du texte par un ou plusieurs espaces.
- si on rajoute un espace à un mot (au milieu) la réponse est fautive.
- insensible à la casse par défaut.

## 5. Evaluation des diverses approches pour les réponses courtes :

## 5.1. Question à trous :

Tableau 2.1- Comparaison entre les questions à trous.

Points de comparaison.	Moodle	Claroline	HotPotatoes	eFront
Les espaces (avant et après).	*acceptés.	*non acceptés.	*acceptés.	*acceptés.
Si le trou contient plus d'un mot (les espaces au milieu).	*il ne faut pas séparer entre les mots par plus d'un espace.	*il ne faut pas séparer entre les mots par plus d'un espace.	* on peut séparer les mots du trou par plusieurs espaces.	* on peut séparer les mots du trou par plusieurs espaces.
Notation.	*Une note pour chaque trou.	*une note pour chaque trou.	*Il n'y a pas de note précisée mais un pourcentage par défaut.	*Il n'y a pas de note précisée mais un pourcentage par défaut.
Pénalité.	*oui	*oui	*non	*oui
Liste déroulante.	*non	*il est possible de le mettre sous forme de liste. *on peut même ajouter des réponses fausses à la liste.	*il est possible de le mettre sous forme de liste.	*il est possible de le mettre sous forme de liste. *on peut même ajouter des réponses fausses à la liste.
Sensibilité à la casse.	* Les réponses peuvent ou non être sensibles à la casse.	*Insensible à la casse.	*Les réponses peuvent ou non être sensibles à la casse.	*Insensible à la casse.
Plusieurs réponses possibles.	*oui	*non	*oui	*oui

5.2. Question à réponse courte :

Tableau 2.1- Comparaison entre les questions à réponses courte.

Points de comparaison.	Moodle	Claroline	HotPotatoes	eFront
Les espaces (avant et après).	*acceptés.		*acceptés.	*acceptés.
Séparation des mots par des espaces.	*il ne faut pas séparer entre les mots par plus d'un espace.		* on peut séparer les mots du trou par plusieurs espaces.	* on peut séparer les mots du trou par plusieurs espaces.
Association de pondération aux réponses : juste, acceptable, faux, pourcentage...	*oui		*oui	*oui
Pénalité.	*oui		*non	*oui
Sensibilité à la casse.	* Les réponses peuvent ou non être sensibles à la casse.		*Les réponses peuvent ou non être sensibles à la casse.	*Insensible à la casse.
Utilisation des pronoms (le, la ...el en Arabe etc...).	voir la remarque (1)		voir la remarque (1)	*voir la remarque (2)
Plusieurs réponses possibles.	*oui		*oui	*oui

Remarque (1) :

On peut utiliser les pronoms (le, la ..... ) d'une manière à définir plusieurs réponses par exemple : Cheval, Un cheval et Le cheval.....

**Remarque (2) :**

Puisqu'on a la possibilité d'insérer plusieurs composant de la réponse, on peut mettre le pronom comme un composant si c'est nécessaire on lui donne une note sinon on le laisse à 0.

**5.3. Les caractéristiques non considérés**

Les critères suivants **ne sont pas pris en charge** par aucun des logiciels qu'on a traité :

1. La considération ou non des accents.
2. Les doublons (deux mêmes caractères au lieu d'un seul) du a une erreur de frappe.
3. La considération des expression sonore très proche (on, an, em, om, ph, f.)...
4. L'association d'outils externe tels qu'un dictionnaire ou un correcteur d'orthographe.
5. Construction de dictionnaire particulier contenant les mots à utiliser.

## CHAPITRE 3



---

---

# Evaluation Flexible des réponses courte

---

---

## 2. Détermination des éléments à considérer par l'enseignant :

Dans ce qui suit, nous essayerons de déterminer les situations à partir desquelles nous déterminerons les aspects les plus importants que les enseignants doivent considérer et classer comme important ou non lors de la préparation des questions. Ces situations concernent surtout l'aspect erreur d'orthographe et d'inattention.

- Les espaces : Dans une situation ordinaire, un seul espace est inséré entre les mots lors de l'écriture d'une réponse. C'est ce que fait l'enseignant en définissant la réponse type. Lors des réponses, il est fort probable que l'étudiant ne suit pas ce style d'écriture et ajoute des espaces supplémentaires, au moins par inattention.
- Les accents : souvent celui qui répond ne peut déterminer s'il faut un accent et si cet accent est grave, aigu ou circonflexe dans le cas de la langue française
- Les expressions ayant un fond sonore similaire ou homophones : A titre d'exemple le mot aire pourrait être écrit ère. Le mot diffuser pourrait être écrit diffuzer.
- La conjugaison : un verbe qui devrait être conjugué au présent est alors conjugué au passé ou au futur ou très mal conjugué
- Le doublement de mot ou de lettre. Ce phénomène apparaît souvent et est du souvent à une erreur d'inattention
- Utilisation de mauvais déterminant

En plus de ces éléments il est fort probable que selon la nature de la matière enseignée, l'enseignant pourrait introduire de nouveaux éléments permettant de rendre l'évaluation automatique plus efficace. A titre d'exemple, si nous considérons le module de programmation dans un langage évolué, l'enseignant pourra accepter des instructions qui ne respectent pas très rigoureusement le langage de programmation. A titre d'exemple, il pourra indiquer que le point-virgule à la fin de l'écriture d'une instruction est optionnel et qu'un bloc, commençant par une accolade ouvrante et se terminant par une accolade fermante et contenant une instruction est considéré comme correct. Ainsi il serait intéressant de permettre à l'évaluation automatique des réponses courtes de s'adapter aux spécificités des matières. Ainsi chaque matière donnera lieu à un style particulier de préparation des questions et à l'introduction d'une fonctionnalité de gestion de style pour ajouter, mettre à jour ou supprimer des styles

Dans ce qui suit nous présentons comment divers éléments seront considérés par l'évaluation automatique

### 3. Architecture générale de l'outil à concevoir et réaliser :

L'outil est composé les parties fondamentales suivantes :

- La partie IHM qui permet de préparer une question.
- La partie gestionnaire de style de préparation de la question.
- La partie sélecteur de style.
- La partie préparation de la question et des réponses types selon le style sélectionné
- La partie saisie des réponses
- La partie évaluation
- La partie test ou prévisualisation

La partie IHM contiendra les éléments suivants :

- La zone identifiant la question, notamment le titre et la banque de question dans laquelle cette question sera sauvegardée.
- La zone contenant la question
- Une zone contenant la réponse type et sa pondération. En général la pondération est de 100%.
- Une zone permettant de sélectionner le type de matière concernée. Selon le type de matière sélectionnée, un certain nombre de champs seront alors affichés pour spécifier les éléments à considérer lors de l'évaluation automatique. Les champs présentés dans ce qui suit apparaissent dans le cas où aucune matière particulière n'est sélectionnée et que nous avons appelé générique.

Les champs concernant le type de matière générique :

- Une ou plusieurs zone, contenant chacune les cas à considérer lors de l'évaluation automatique. Une pondération est aussi associée à chaque zone. Elle pourrait être de 100% ou moins. C'est l'enseignant qui décidera de cette pondération en fonction des erreurs à supportés dans une réponse. Une zone de ce type contient les éléments suivants :
  - Une zone permettant de définir une réponse approchée de la réponse types. La réponse approchée est définie pour permettre de faire abstraction lors de l'évaluation automatique des erreurs d'orthographe, des erreurs d'inattention et

des oublis. Dans cette zone nous trouverons en général des mots clés spécifiés complètement ou partiellement.

- Une zone pour sélectionner et définir si nécessaire les homophones à considérer
- Des cases à cocher indiquant l'importance à donner aux éléments définis précédemment. A titre d'exemple on aura une case appelée « ignorer les accents », une case appelée ignorer les doublons, une case appliquer les homophones.
- Une zone de test de la prise en charge des réponses et choix indiqués

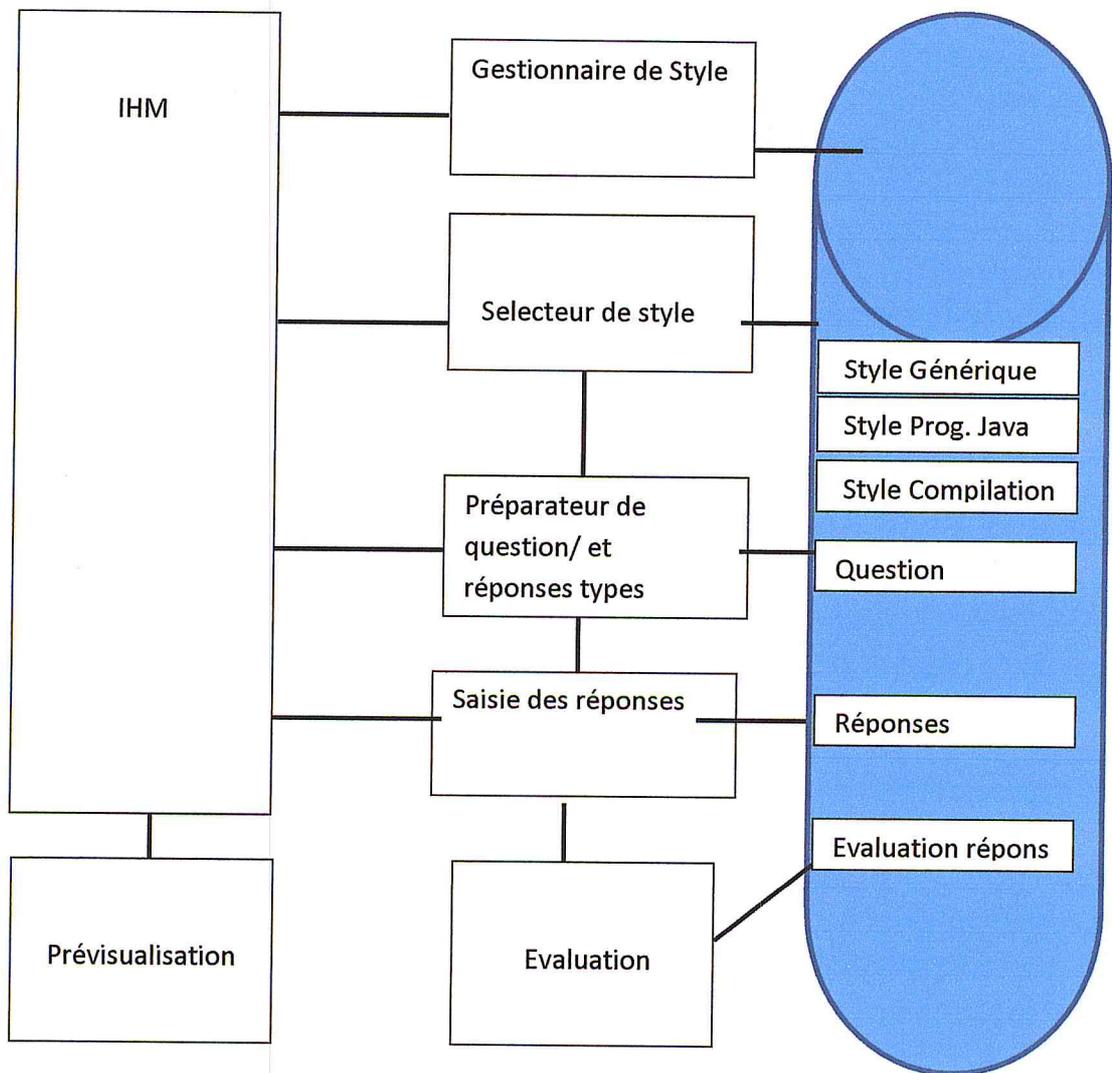


Figure 3.0 : Architecture du système.

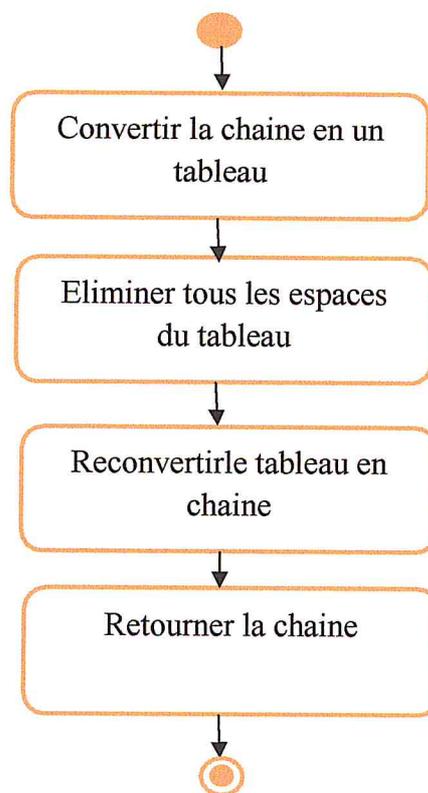
#### 4. Conception et description des différentes fonctionnalités ajoutées :

##### 4.1 L'approche utilisée pour résoudre le problème des blancs :

###### Description :

On veut réaliser une fonction qui élimine tous les espaces qui existe dans une chaîne.

###### Diagramme d'activité pour la fonction:



```
function elim_esp($chaine)
{
    $n=strlen($chaine); //longeur de la chaine
    $tab=str_split($chaine); //chaine->tab
    $tab2=array('');
    $i=0;
    $j=0;
    while($i<$n)
    {
        if($tab[$i]!=" ")
        {
            $tab2[$j]=$tab[$i];
            $i++;
            $j++;
        }
        else $i++;
    }
    $txt=implode("", $tab2); //tab->chaine
    return $txt;
}
```

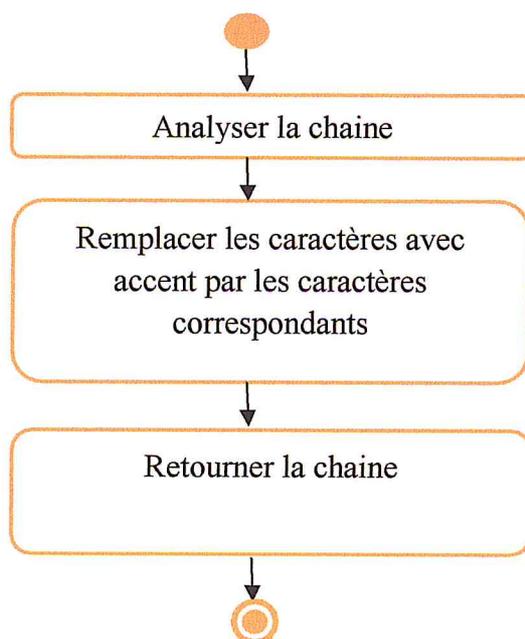
Figure 3.1 : La fonction qui élimine les espaces.

#### 4.2 L'approche utilisée pour résoudre le problème des accents :

##### Description :

A ce niveau on a besoin d'une fonction qui retourne une chaîne sans aucun caractère ayant accent.

##### Diagramme d'activité pour la fonction:



```
function elim_accent($chaine)
{
    $e= array ('é', 'è', 'ê', 'Ê', 'Ë', 'É');
    $a= array ('à', 'á', 'â', 'Â', 'Ã');
    $i= array ('ì', 'í', 'ï', 'Î');
    $o= array ('ô', 'ó', 'ò', 'Ô');
    $u= array ('û', 'ú', 'Û');
    $chaine=str_replace($e, 'e', $chaine);
    $chaine=str_replace($a, 'a', $chaine);
    $chaine=str_replace('ç', 'c', $chaine);
    $chaine=str_replace($i, 'i', $chaine);
    $chaine=str_replace($o, 'o', $chaine);
    $chaine=str_replace($u, 'u', $chaine);
    return $chaine;
}
```

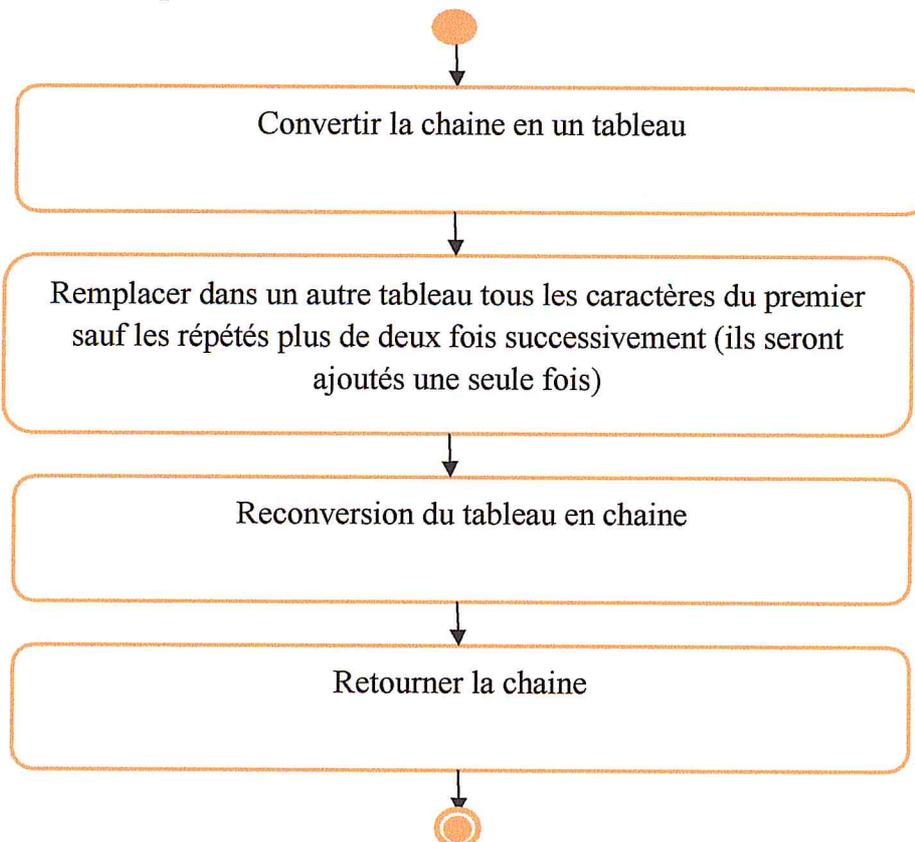
Figure 3.2 : La fonction qui élimine les accents.

### 4.3 L'approche utilisée pour résoudre le problème des doubles :

**Description :**

Cette fonction élimine tous les caractères doublés dans une chaîne.

**Diagramme d'activité pour la fonction:**



```
function elim_double($chaine)
{
    $n=strlen($chaine);
    $tab=str_split($chaine); // chaine->tab
    $i=0;
    $j=0;
    while($i<$n-1)
    {
        if($tab[$i]!= $tab[$i+1])
        {
            $tab2[$j]=$tab[$i];
            $i++;
            $j++;
        }
        else $i++;
    }
    $tab2[$j]=$tab[$n-1];
    $txt=implode("",$tab2); //tab->chaine
    return $txt;
}
```

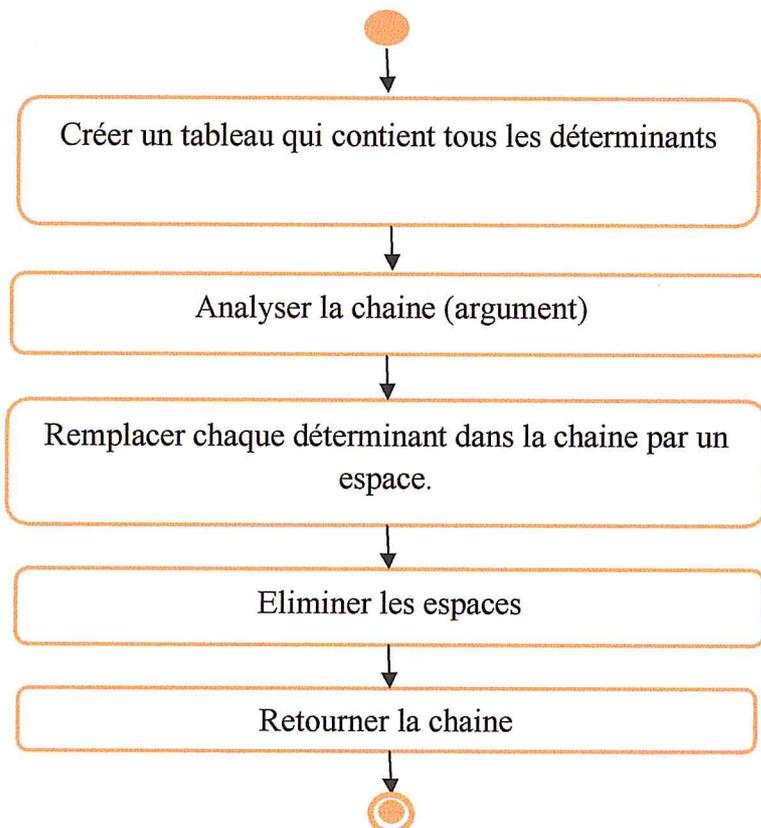
Figure 3.3 : La fonction qui élimine les doubles.

4.4 L’approche utilisée pour résoudre le problème des déterminants :

**Description :**

Dans cette option on a besoin d’une fonction qui ne prend pas en considération les déterminants (une, un, la, le,les, des ...) dans une chaine de caractères.

**Diagramme d’activité pour la fonction :**



#### 4.6 L'approche utilisée pour résoudre les problèmes des questions orientées vers un langage de programmation:

##### Description :

On a besoin d'une fonction pour éliminer les espaces avant et après les caractères spéciaux.

##### Description de la fonction:

On met dans un tableau tous les caractères spéciaux, en suite on converti la chaine principale en un tableau de caractères ; dans ce tableau on élimine tous les espaces qui sont avant et après un caractère spécial, on reconverti et on retourne la chaine.

```
function espace_caractere($chaine)
{
    $n=strlen($chaine);
    $i=0;$j=0;
    $src=array('!', '=', '{', '}', '[', ']', '|', '$', '+', '-', '*', '/', '<', '>', '@', '?', '>', '\\', '\\');
    $tab1=str_split($chaine);
    while ($i<$n)
    {
        if((($tab1[$i]==" " && in_array($tab1[$i-1],$src)) || ($tab1[$i]==" " && in_array($tab1[$i+1],$src)) )
        {
            $tab2[$j]=$tab1[$i+1];
            $i++;
        }
        else
        {
            $tab2[$j]=$tab1[$i];
            $j++;
            $i++;
        }
    }
    $chaine=implode("", $tab2);
    return $chaine;
}
```

Figure 3.6 : la fonction qui traite les caractères spéciaux

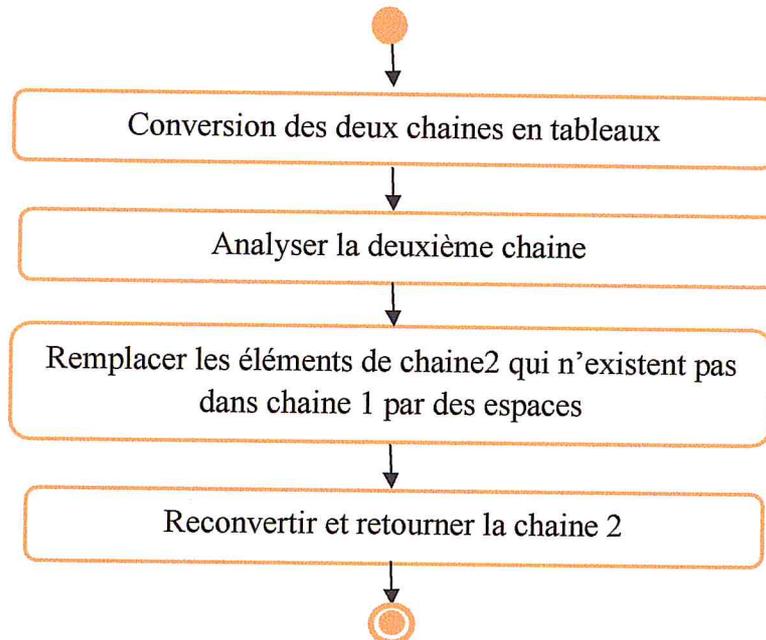
#### 4.7 L'approche utilisée pour réaliser des réponses acceptées à base de mots clés :

##### Description :

On utilise une fonction pour vérifier si un certain nombre de mots existe par ordre dans une chaine de caractère.

**Diagramme d'activité pour la fonction:**

**Remarque :** La fonction a deux chaînes de caractères et un séparateur comme arguments, chaîne 1 contient les mots clés et chaîne 2 contient la réponse.



```
function mots_clé($chaîne1,$chaîne2,$séparateur)
// $chaîne1-->lé mots clés $chaîne2-->rep de l'étudiant
{
    $chaîne1=explode($séparateur,$chaîne1); //chaîne->tab
    $chaîne2=explode(" ",$chaîne2); //chaîne->tab
    $n=count($chaîne2); //taille du tab
    $i=0;
    while($i<$n)
    {
        if(in_array($chaîne2[$i],$chaîne1))
        {
            $i++;
        }
        else
        {
            $chaîne2[$i]=' ';
            $i++;
        }
    }
    $chaîne=implode("",$chaîne2); //tab->chaîne
    return $chaîne;
}
```

Figure 3.7 : La fonction des mots clés.

#### 4.8 L'approche utilisée pour réaliser des réponses acceptées à base de parties de mots clés :

##### Description :

On a besoin d'une fonction pour vérifier si des parties de mots existent dans une chaîne de caractères.

Pour cette option on a prévu 4 fonctions :

##### 1/ Une fonction qui élimine les symboles du début et de la fin d'un mot :

##### Description :

A ce niveau, on récupère la chaîne de caractère qui contient les mots clés séparés par un symbole, et les met dans un tableau de mots, en suite on analyse le tableau ; s'il contient le symbole au début, on récupère la partie de la chaîne qui se trouve après (le symbole) et si le symbole est à la fin on récupère la partie qui est avant, et si le symbole est au début et à la fin on récupère la chaîne qui est au milieu.

```
function elim_etoile($chaine)
{
    $csrc=$_POST['csrc'];
    $sep=$_POST['separateur'];
    $rep2=explode($sep,$chaine); //tab des mots////////////////////////////////////
    $i=0;$n=count($rep2);
    $stab=array('');
    while ($i<$n)
    {
        $rep=$rep2[$i];
        $x=strlen($rep);
        if ($rep[$x-1]==$csrc && $rep[0]==$csrc)
        {
            $rep=substr($rep,1,-1);
            $stab[$i]=$rep; $i++;
        }
        else if ($rep[$x-1]==$csrc)
        {
            $rep=substr($rep,0,-1);
            $stab[$i]=$rep;$i++;
        }
        else if ($rep[0]==$csrc)
        {
            $rep=substr($rep,1);
            $stab[$i]=$rep;$i++;
        }
        else
        {
            $stab[$i]=$rep; $i++;
        }
    }
    return $stab; // retourne une chaine
}
```

Figure 3.8 : la fonction qui élimine les symboles dans une chaîne.

## 2/Les fonctions qui vérifie si un mot commence ou se termine avec une partie indiquée :

La fonction « debut\_avec » : retourne true si chaine 1 commence par chaine 2.

La fonction « fin\_avec » : retourne true si chaine 1 se termine par chaine 2.

```
function debut_avec( $str, $sub ) {
    return ( substr( $str, 0, strlen( $sub ) ) == $sub );
}

function fin_avec( $str, $sub ) {
    return ( substr( $str, strlen( $str ) - strlen( $sub ) ) == $sub );
}
```

Figure 3.9 : les deux fonctions de vérification.

## 3/La fonction qui vérifie si une partie de mot existe dans une chaine :

Cette fonction contient trois tableaux, le premier contient la réponse, le second contient les parties des mots à rechercher, on analyse ce dernier pour mettre les mots qui se corresponde avec le premier dans le troisième tableau. En suite on retourne la chaine.

```
function compare_partie_mot($tab1=array(''),$tab2=array(''))
//tab1== tab des mot clés  tab2== la chaine de l'étudiant
{
    $u=0;
    $i=0;
    $j=0;
    $n=count ($tab1); $m= count($tab2);
    $tab3=array('');
    for($i=0;$i<$n;$i++)
    {
        for($j=0;$j<$m;$j++)
        {
            if(debut_avec($tab2[$j],$tab1[$i]) || fin_avec($tab2[$j],$tab1[$i]))
            {
                $tab3[$u]=$tab2[$j];
                $u++; break;
            }
        }
    }
    return $tab3;
}
```

Figure 3.10: la fonction de comparaison.

## 5. Les fonctions PHP utilisées :

Tableau 3.1-les fonctions PHP utilisées	
FONCTIONS	DESCRIPTION
<b>explode</b> ( string \$delimiteur, string \$chaine)	Coupe une chaîne en segments
<b>implode</b> (string \$séparateur, array \$tableau)	Rassemble les éléments d'un tableau en une chaîne
<b>strlen</b> (string \$chaine)	Calcule la taille d'une chaîne
<b>str_replace</b> (\$search,\$remplacé, \$chaine)	Remplace toutes les occurrences du \$search par \$remplacé dans la chaîne \$chaine
<b>substr</b> ( string \$string , int \$debut ,int \$fin)	Retourne un segment de chaîne qui est entre \$debut et \$fin
<b>str_split</b> (string \$chaine)	Convertit une chaîne de caractères en tableau
<b>in_array</b> ( mixed \$val , array \$tableau)	Indique si une valeur appartient à un tableau
<b>count</b> (array \$tab)	Compte tous les éléments d'un tableau.

# CHAPITRE 4

---

## Réalisation et tests

---

## 1. Introduction :

Dans ce chapitre nous allons montrer le contexte dans lequel nous avons réalisé nos outils. En suite nous présenterons comment nous avons réalisé notre outils, les techniques utilisé pour réaliser cet outils et le divers test

## 2. Environnement de réalisation

L'outil a été réalisé dans le contexte de la plateforme Moodle. Il représente ainsi une extension de la plateforme Moodle. C'est ce qui est souvent appelé par plugin. Le langage utilisé est celui utilisé par Moodle, c'est-à-dire le langage PHP.

La conception et réalisation d'un plugin Moodle suit une démarche bien précise qu'il est nécessaire de maîtriser. La plus grande complexité que nous avons rencontré c'est le fait que Moodle soit réalisé complètement en orienté objet et que ce dernier soit composé d'un nombre très important de fichiers interdépendant. Ainsi malgré la présence d'une méthode claire de mise en place de nouveau module pour Moodle, la mise en place réelle d'un nouveau module est très délicate et nécessite un temps important pour la compréhension et la mise en œuvre. Dans ce qui suit nous allons illustrer comment un module peut être créé sous Moodle. Par la suite nous exposerons, par les exemples les diverses fonctionnalité de notre outil

## 3. Modules et plugins de « Moodle » :

### 3.1. Définition d'un module ou plugin:

Le **M** dans Moodle est synonyme de **modularité**. La façon la plus facile à entretenir et à ajouter de nouvelles fonctionnalités à Moodle est écrite par l'un de ces types de plugin. [9]

Il existe plusieurs catégories de plugins telle que:

- **Thème:** Pour changer le look des pages du cours.
- **Format de cours:** Pour changer la structure (layout) des pages du cours.
- **Activités:** La principale activité des étudiants dans un cours.....

### 3.2. Création d'un nouveau module Moodle :

Pour ajouter un nouveau module dans la plateforme Moodle on a suivi les étapes d'installation d'un plugin de la communauté Moodle.

- 1/ téléchargement du fichier du plugin vierge compressé (**NEWMODULE.zip**).

2/ Décompression de l'archive et lecture du fichier **README.txt** pour suivre les étapes d'installation.

3/ Renommer le dossier **NEWMODULE** (dans ce cas on a donné à notre plugin le nom 'nvtype'). Le nom du dossier du module doit être en minuscules et ne contient que des caractères alphabétiques.

4/ Modification de tous les fichiers dans le répertoire et ses sous-répertoires et changement de toutes les instances de la chaîne "newmodule" par 'nvtype'.

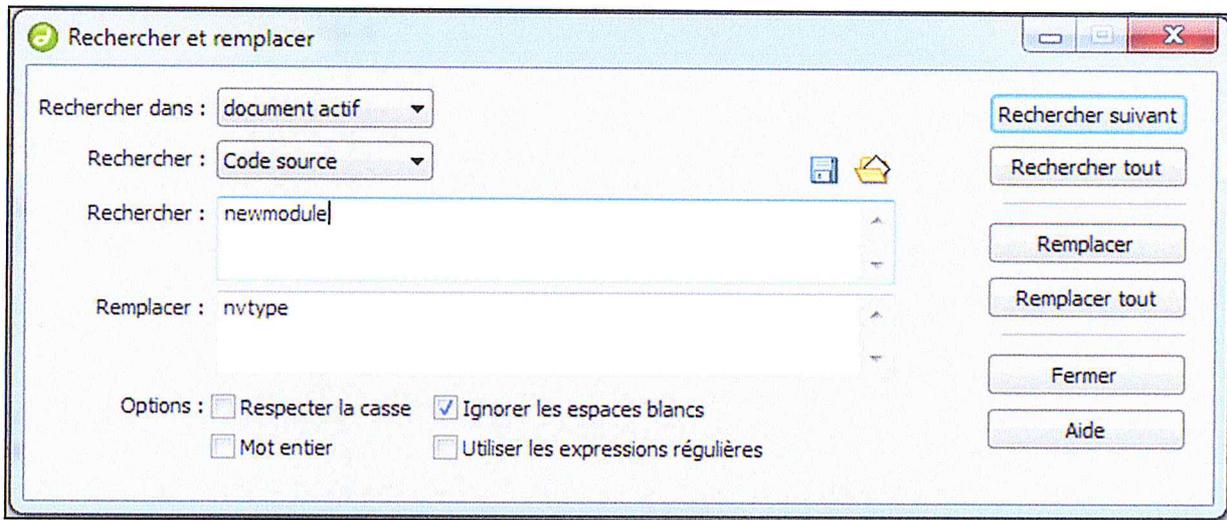


Figure 4.1 : renommer les instances du module.

5/ Mettre le dossier de 'nvtype' dans le dossier /mod du répertoire de Moodle (dans notre cas on a placé le dossier dans moodle/question/type).

6/ Modification des tables du module dans Paramètres>Administration du site> Développement>éditeurXMLDB.

question/type/numerical/db	[Créer]	[Charger]	[Modifier]	[Enregistrer]	[Doc]	[XML]	[Annuler]	[Unload]	[Supprimer]
question/type/nvtype/db	[Créer]	[Charger]	[Modifier]	[Enregistrer]	[Doc]	[XML]	[Annuler]	[Unload]	[Supprimer]
question/type/random/db	[Créer]	[Charger]	[Modifier]	[Enregistrer]	[Doc]	[XML]	[Annuler]	[Unload]	[Supprimer]
question/type/randomsamatch/db	[Créer]	[Charger]	[Modifier]	[Enregistrer]	[Doc]	[XML]	[Annuler]	[Unload]	[Supprimer]

Figure 4.2 : modification des tables.

7/ On trouve les tables du module créé avec succès dans Paramètres> Administration du site>Notifications.



Figure 4.3 : succès d'installation.

8/ lors de la visite de **Administration>Plugins>Modules d'activités> Gérer les activités**, on trouve que ce nouveau module a été ajouté à la liste des modules installés.

Numerical qtype_numerical	Standard	2012061700	Uninstall
Mon type qtype_nvtype	Add-on	2012061700	Uninstall
Random qtype_random	Standard	2012061700	Uninstall
Random short-answer matching qtype_randomsamatch	Standard	2012061700	Uninstall

Figure 4.4 : l'ajout du module à la liste des modules.

Voilà, notre nouveau module a été ajouté avec succès à la liste des types de question.

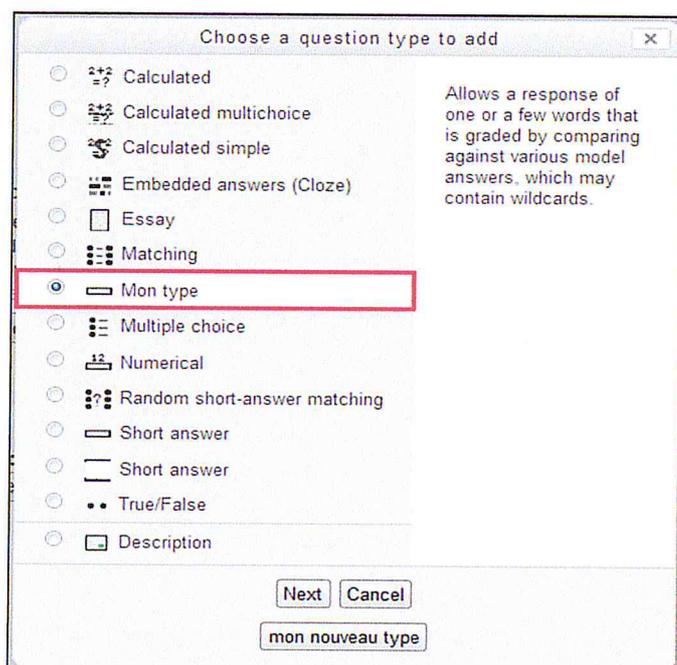


Figure 4.5 : liste des types de question.

Après le succès de ces étapes, il faut ajouter notre propre code pour que le plugin arrive à réaliser nos buts.

### 4. Réalisation des fonctionnalités du module :

La figure suivante illustre globalement le fonctionnement de notre système:

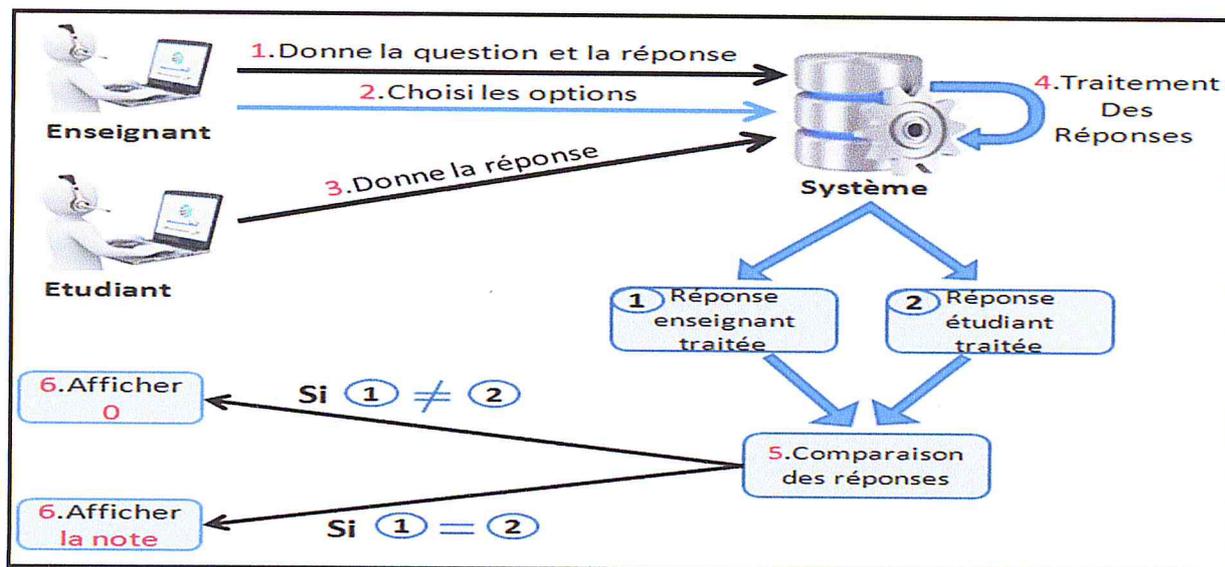


Figure 4.6 : schéma générale de la correction automatique dans le nouveau système.

#### 4.1 :L’option «Négliger les blancs» :

L’enseignant donne la question et son nom et la note correspondante.

**Généraux**

Nom de question

Texte de la question

Note par défaut\*

Figure 4.7 : création de question.

Puis, il donne la réponse de la question et il choisit l’option pour que les blancs ne posent pas de problèmes lors de la correction.

**Réponse de l'enseignant**

Réponse de l'enseignant \*

Négliger les espaces

Négliger les accents

Négliger les caractères en doubles

Figure 4.8 : choix de l'option «négliger les blancs ».

L'étudiant répond à la question.

**Réponse de l'étudiant**

Réponse de l'étudiant \*

Figure 4.9 : réponse de l'étudiant.

Le système considère la réponse correcte et donne la note complète même s'il y a plus d'un blanc par rapport à la réponse de l'enseignant.

**Question 1**

Correct

**Note : 10 sur 10**

**Texte de la question**

quel est le nom complet du président de l'Algérie?

**Réponse de l'étudiant**

Reponse :

**La reponse correcte est :**

abdelaziz bouteflika

Figure 4.10 : correction avec plusieurs blancs exemple1.

Un autre exemple de réponse à la même question et avec des espace au début et à la fin de la réponse.

**Question 1**  
Correct

Note : 10 sur 10

**Texte de la question**  
quel est le nom complet du président de l'Algérie?

**Réponse de l'étudiant**  
Reponse : abd elaziz bou teflika

**La reponse correcte est :**  
abdelaziz bouteflika

Figure 4.11 : correction avec plusieurs blancs exemple2.

### 4.2. L'option « Négliger les accents » :

L'enseignant écrit une question et donne sa réponse, puis il choisit cette option pour que les accents ne soient pas pris en compte.

La figure si dessous montre que la réponse est correcte même s'il y a une différence des accents.

<p><b>Réponse de l'étudiant</b> Reponse : eleve</p> <p><b>La reponse correcte est :</b> élève</p>	<p><b>Réponse de l'étudiant</b> Reponse : élève</p> <p><b>La reponse correcte est :</b> élève</p>	<p><b>Réponse de l'étudiant</b> Reponse : batiment</p> <p><b>La reponse correcte est :</b> bâtiment</p>
---	---	---

Figure 4.12 : exemple de l'option « négliger les accents ».

### 4.3. L'option « Négliger les caractères en doubles » :

Après avoir écrit la question, il donne la réponse correspondante à la question et choisi l'option pour que le system ne prend pas en considération les erreurs de frappe et les doubles caractères.

L'étudiant connaît la réponse mais il pense qu'elle s'écrit avec un double 'p', alors qu'en réalité elle s'écrit avec un seul 'p', dans ce cas il aura la note complète.

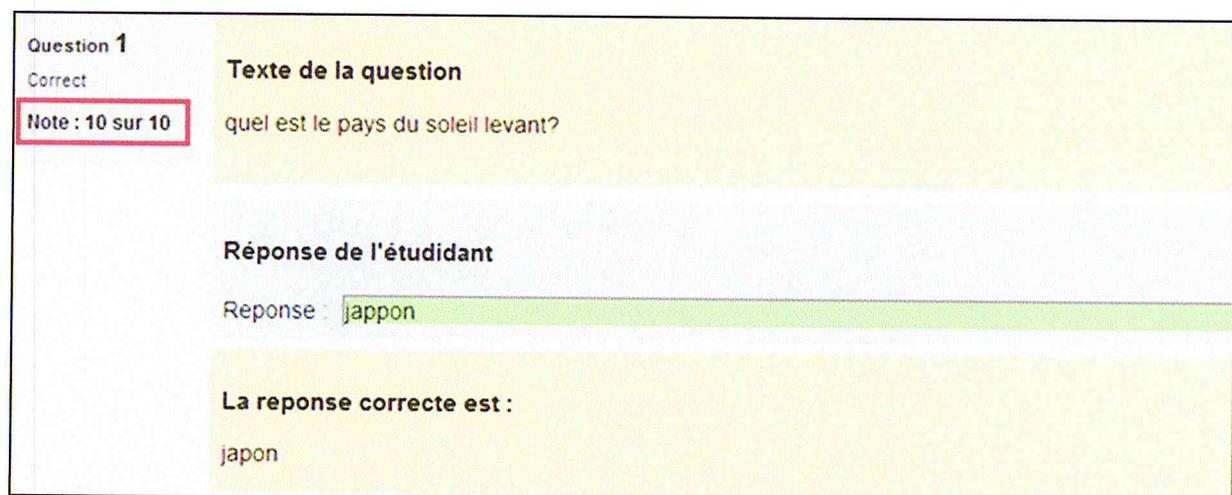


Figure 4.13 :exemple1 de l’option «Négliger les doubles »

Dans cet exemple, la bonne réponse est ‘le japon’ mais l’étudiant en plus de ses fautes d’orthographe il a fait une erreur de frappe en doublant un caractère.Dans ce cas l’étudiant aura la note complète, ce qui montre l’utilité de cette option.

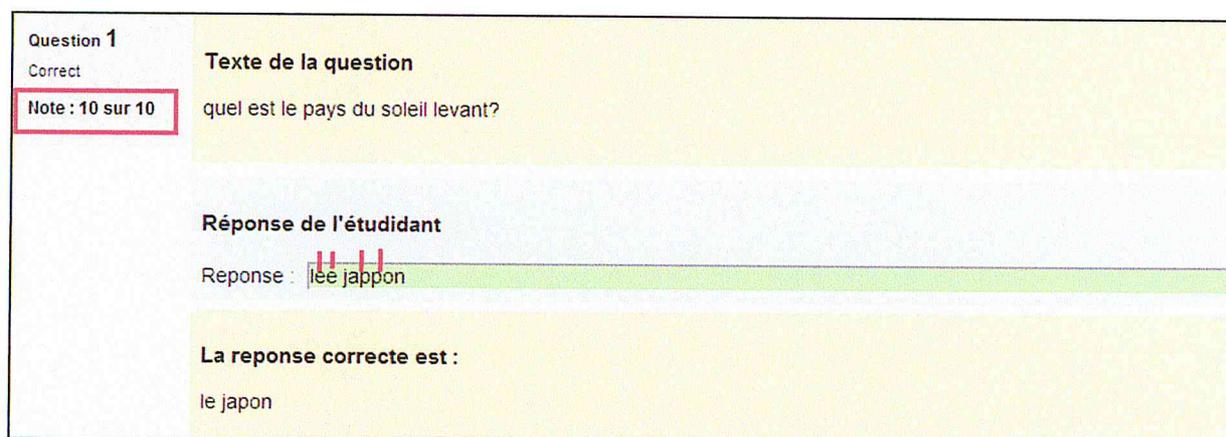


Figure 4.14 : exemple 2 de l’option «Négliger les doubles »

#### 4.4. L’option« Négliger les déterminants » :

D’après la figure si dessous qui illustre le fonctionnement de cette option, on peut facilement comparer la réponse de l’étudiant avec la réponse donné par l’enseignant, et voir qu’il y a une différence dans l’écriture de la phrase à cause des déterminants mais pas vraiment dans le sens, alors le système donne la note complète. Mais si l’étudiant ajoute un caractère en plus de la réponse donné par l’enseignant le system lui donne un zéro.

<p><b>Réponse de l'étudiant</b></p> <p>Reponse : l'endroit ou habite une famille</p> <p><b>La reponse correcte est :</b></p> <p>l'endroit ou habite la famille</p>	<p><b>Réponse de l'étudiant</b></p> <p>Reponse : l'endroit ou habite des famille</p> <p><b>La reponse correcte est :</b></p> <p>l'endroit ou habite la famille</p>
--	--

Figure 4.15 : exemple de l'option « négliger les déterminants ».

#### 4.5. L'option « Accepter les même homophones » :

Négliger les déterminants (la,le,les...)

Accepter les même homophones

Orienter vers un langage du programmation

Figure 4.16 : choix de l'option « Accepter les même homophones ».

Cette option permet à l'étudiant d'avoir la note complète même s'il ne connaît pas bien l'orthographe, on peut dire qu'il suffit d'entré le mot comme celase prononce et sa réponses sera accepter.

La figure si dessous montre un exemple de réponse avec 'f' au lieu 'ph', le système considère la réponse correcte car 'ph' et 'f' se prononcent de la même façon.

**Question 1**

Correct

**Note : 10 sur 10**

**Texte de la question**

d'ou on achète les médicaments?

**Réponse de l'étudiant**

Reponse : de la farnacie **ph <=> F**

**La reponse correcte est :**

de la pharmacie

Figure 4.17 : exemple de l'option « Accepter les même homophones ».

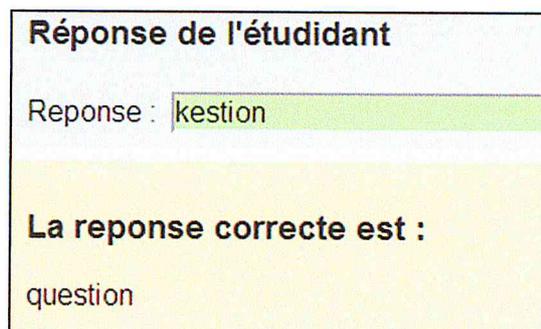


Figure 4.18 : exemple de 'ke' au lieu 'que'.

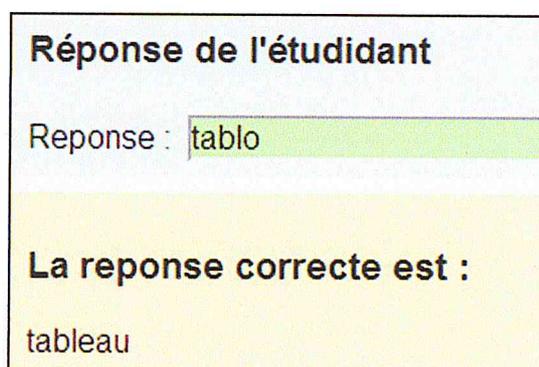


Figure 4.19 : exemple de 'o' au lieu 'eau'.

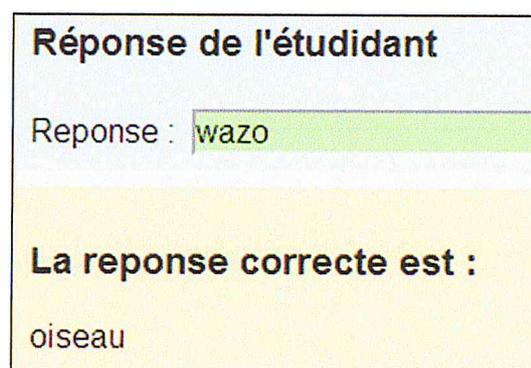


Figure 4.20 : exemple de 'wa' au lieu 'oi'.

Dans cet exemple l'emplacement de 'si' dans le mot retourne la prononciation du mot 'farmazien' alors que la réponse est 'farmassien'.

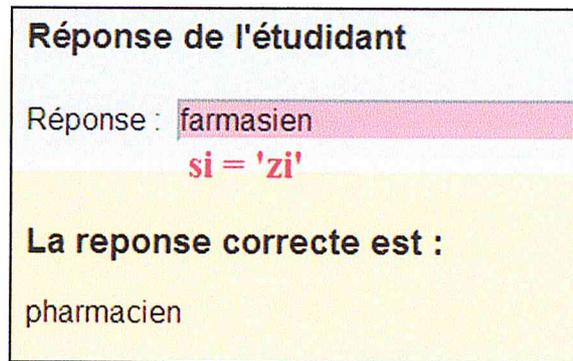


Figure 4.21 : exemple de 'si' au lieu 'zi'.

'ge' se prononce 'je' mais pas 'gue' :

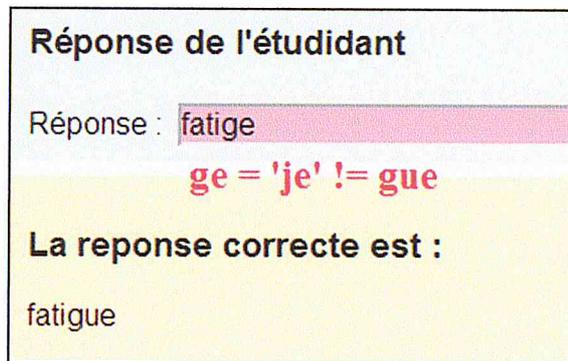


Figure 4.22 : exemple de 'je' au lieu 'gue'.

Il existe encor d'autres cas de similarité phonétique qui peuvent être traités par cette option comme le 'ç', 'c+e' ou 'c+i' avec 's', le 'y' avec 'i', le 'ch' avec 'sh'.....

#### 4.6. L'option « Orienter vers un langage de programmation » :

Après avoir donné la réponse, l'enseignant choisit cette option si la réponse est un programme (code) en un langage de programmation, par exemple JAVA, C++, PHP...

**Réponse de l'enseignant**

Réponse de l'enseignant \*

```
function note_exam (Snote) { if (Snote>=10) { Snote=100%; } else if (Snote<=90 && Snote>=70) { Snote=80%; } return Snote; }
```

- Négliger les espaces
- Négliger les accents
- Négliger les caractères en doubles
- Négliger les déterminants (la, le, les...)
- Accépter les même sons
- Orienter vers un langage du programmation

Figure 4.23 : Exemple d'une question pour et choix de l'option.

L'étudiant écrit son programme, il peut ajouter des espaces mais pas entre les mots réservés et les noms de variables, bien sur il faut qu'il suive la syntaxe.

**Réponse de l'étudiant**

Réponse de l'étudiant \*

```
function note_exam ( Snote ) { if ( Snote >= 10 ) { Snote = |100% ; } else if ( Snote <= 90 && Snote >= 70 ) { Snote = 80% ; } return Snote; }
```

Figure 4.24 : Exemple de réponse de l'étudiant.

Alors, après l'analyse de la réponse, si le système ne trouve pas de problèmes d'espaces ajoutés dans la réponse de l'étudiant et qu'elle est syntaxiquement correcte, il lui donne la note complète.

Question 1  
Correct

**Note : 10 sur 10**

**Texte de la question**  
Ecrire en PHP une fonction note\_exam qui retourne 100% si la note > ou égale 10, et 80% si la note est entre 70 et 90.

**Réponse de l'étudiant**  
Reponse : `function note_exam ( Snote ) { if ( Snote >= 10 ) { Snote = 100% ; } else if ( Snote <= 90 && Snote >= 70 ) { Snote = 80% ; } return Snote; }` ✓

**La reponse correcte est :**  
`function note_exam (Snote) { if (Snote>=10) { Snote=100%; } else if (Snote<=90 && Snote>=70) { Snote=80%; } return Snote; }`

Figure 4.25 : Exemple de réponse de l'étudiant avec réponse correcte.

Mais, s'il y a un problème dans la syntaxe le system lui donne un zéro. Cela est illustré dans l'exemple suivant, dans la syntaxe le nom de la fonction ne contient pas des espaces.

Question 1 Incorrect Note : 0 sur 10	<b>Texte de la question</b> Ecrire en PHP une fonction note_exam qui retourne 100% si la note > ou égale 10. et 80% si la note est entre 70 et 90.
	<b>Réponse de l'étudiant</b> Réponse : <code>function note_exam ( Snote ) { if ( Snote &gt;= 10 ) { Snote = 100%; } else if ( Snote &lt;= 90 &amp;&amp; Snote &gt;= 70 ) { Snote = 80%; } return Snote; }</code>
	<b>La reponse correcte est :</b> <code>function note_exam (Snote) { if (Snote&gt;=10) { Snote=100%; } else if (Snote&lt;=90 &amp;&amp; Snote&gt;=70) { Snote=80%; } return Snote; }</code>

Figure 4.26 : Exemple1 de réponse de l'étudiant avec réponse fausse.

Un espace entre les mots clés est obligatoire si non la réponse est fausse.

Question 1 Incorrect Note : 0 sur 10	<b>Texte de la question</b> Ecrire en PHP une fonction note_exam qui retourne 100% si la note > ou égale 10. et 80% si la note est entre 70 et 90.
	<b>Réponse de l'étudiant</b> Réponse : <code>function note_exam ( Snote ) { if ( Snote &gt;= 10 ) { Snote = 100%; } elseif ( Snote &lt;= 90 &amp;&amp; Snote &gt;= 70 ) { Snote = 80%; } return Snote; }</code>
	<b>La reponse correcte est :</b> <code>function note_exam (Snote) { if (Snote&gt;=10) { Snote=100%; } else if (Snote&lt;=90 &amp;&amp; Snote&gt;=70) { Snote=80%; } return Snote; }</code>

Figure 4.27 : Exemple2 de réponse de l'étudiant avec réponse fausse.

### 4.7. L'option « Réponse à base des mots clés » :

L'enseignant crée la question et donne la réponse correcte et choisit l'option des mots clés, dans ce cas la réponse contient plus d'un mot (sous forme d'un paragraphe).

**Réponse de l'enseignant**

Réponse de l'enseignant \* Les éclipses lunaire se produisent lors d'un alignement parfait Soleil-Terre-Lune. La Lune est alors située dans le cône d'ombre de la Terre. C'est un phénomène courant, qui dure quelques heures et qui concerne une grande partie de la Terre.

- éliminer les espaces
- Les accents
- Les doubles (et les espases)
- Les determinants
- Accépter les sons
- Langage du programmation
- Réponse à base des mots clés

Figure 4.28 : Exemple de réponse de l'enseignant.

Après avoir choisi l'option, des champs à remplir s'affichent à l'enseignant. Par défaut le caractère de séparation est '|', mais l'enseignant peut le changer.

En suite il faut qu'il remplis le champ des mots clés, il entre les mots et les séparés par le caractère choisi sans aucun espace.

\* Caractère de séparation des mots clés :  ←

\* Caractère qui représente la partie négligeable du mot :

Entrez les mots clés et les séparez avec le caractère choisit, SVP :

alignement|soleil-terre-lune|cône|d'ombre ←

1                      2                      3                      4

Figure 4.29 : La saisit des mots clés .

L'étudiant répond à la question, cette réponse n'est pas typiquement égale à la réponse de l'enseignant mais il suffit juste qu'elle contienne les mots clés pour qu'elle soit correcte.

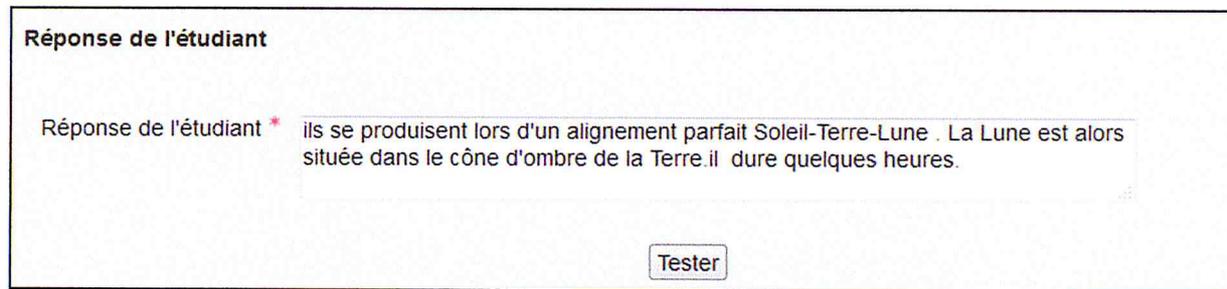


Figure 4.30 : Exemple de réponse de l'étudiant non corrigé.

Tous les mots clés existent dans la réponse de l'étudiant par ordre correcte donc il aura la note complète.

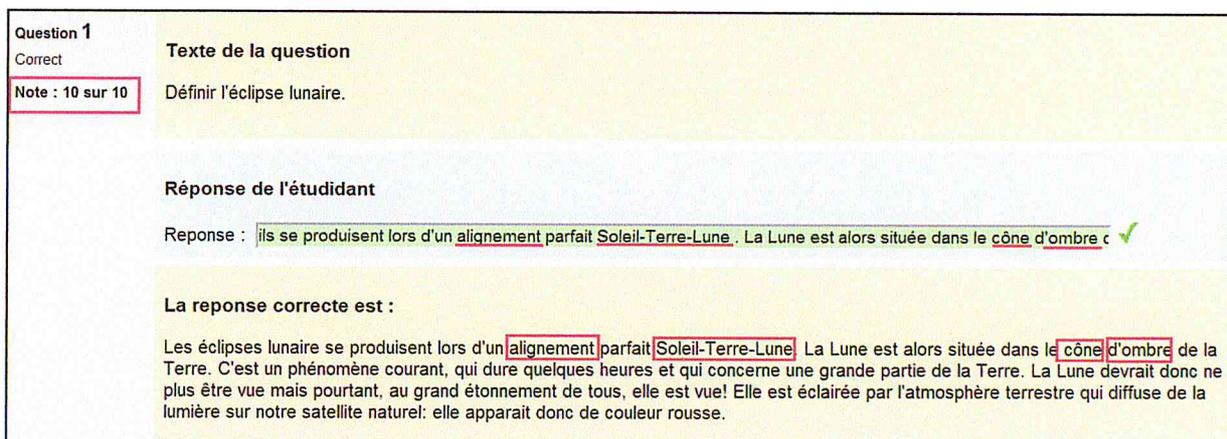


Figure 4.31 : Exemple1 de réponse de l'étudiant corrigé.

Le manque d'un mot clé, les erreurs, ou leur désordre ne sont plus acceptés.

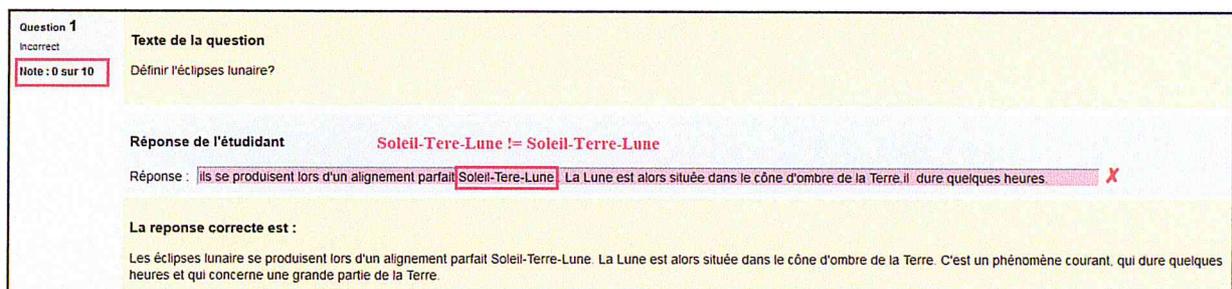


Figure 4.32 : Exemple2 de réponse de l'étudiant corrigé.

### 4.8. L'option «Réponse à base des partie de mots » :

❖ Cas d'un seul mot clé :

On peut dire que cette option est équivalente à la précédente, sauf que dans ce cas on ne cherche pas le mot clés entier mais on cherche s'il existe juste une partie de chaque mot clé dans la réponse de l'étudiant. Le créateur de l'exercice doit remplacer la partie partiel du mot clé par un caractère dont il propose (qui est "\*" par défaut) avant de remplir le champ spécial pour les mots clés.

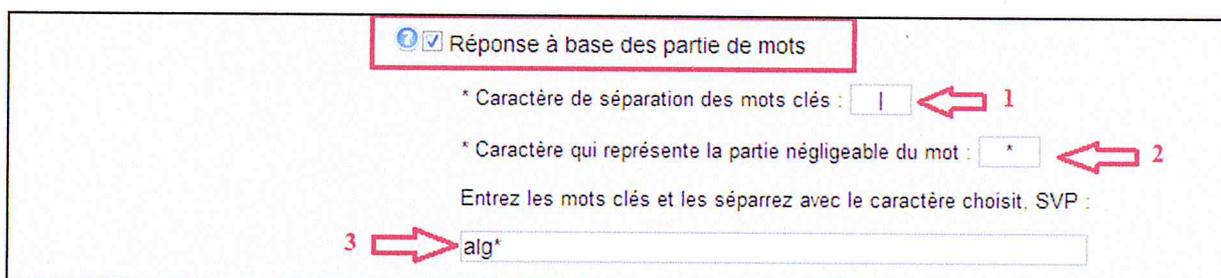


Figure 4.33 : choix de l'option et remplissage des champs.

Dans cet exemple, l'enseignant indique que la réponse de l'étudiant doit contenir obligatoirement un mot commençant avec 'alg' pour qu'il obtienne la note complète, même si le mot ne se termine pas comme le mot de la réponse proposé comme l'illustre la figure suivante.

<p>Réponse de l'étudiant</p> <p>Reponse : algérien</p> <p>La reponse correcte est : les algériens</p>	<p>Réponse de l'étudiant</p> <p>Reponse : algériennes</p> <p>La reponse correcte est : les algériens</p>	<p>Réponse de l'étudiant</p> <p>Reponse : les habitants de l'algerie sont des algerien</p> <p>La reponse correcte est : les algériens</p>
---	--	---

Figure 4.34: Exemple1 avec l'option « Réponse à base des partie de mots» corrigé.

❖ Cas de plusieurs mots clés :

Dans ce cas l'enseignant doit rajouter un autre caractère qui doit être différent du premier pour séparer les mots clés par défaut c'est le '|'.

Si la réponse de l'étudiant contient tous les mots dont lesquels sont incluent les partie proposé le système lui donnera la note complète.

<p>Question 1</p> <p>Correct</p> <p>Note : 10 sur 10</p>	<p><b>Texte de la question</b></p> <p>terminer la phrase : la table.....</p>
	<p><b>Réponse de l'étudiant</b></p> <p>Reponse : la table rouge dans ma maison est belle et très grande mais je l'aime pas</p>
	<p><b>La reponse correcte est :</b></p> <p>la table rouge est grande et belle</p>

Figure 4.35 : Exemple2 avec l'option « Réponse à base des partie de mots» corrigé.

S'il manque au moins un mot ou une partie dans l'ensemble des mots donné par l'étudiant, la réponse sera considéré comme incorrecte.

<p>Question 1</p> <p>Incorrect</p> <p>Note : 0 sur 10</p>	<p><b>Texte de la question</b></p> <p>terminer la phrase : la table.....</p>
	<p><b>Réponse de l'étudiant</b></p> <p>Reponse : la table rouge dans ma maison est belle mais je l'aime pas</p>
	<p><b>La reponse correcte est :</b></p> <p>la table rouge est grande et belle</p>

Figure 4.36 : Exemple3 avec l'option « Réponse à base des partie de mots» corrigé.

### 4.9. L'aide des options :

On a ajouté un bouton d'aide pour bien expliquer le fonctionnement de chaque option dans notre type. Si l'enseignant n'arrive pas à comprendre une option et ses arguments il clique sur le bouton, une boîte de dialogue contient l'explication s'affiche.

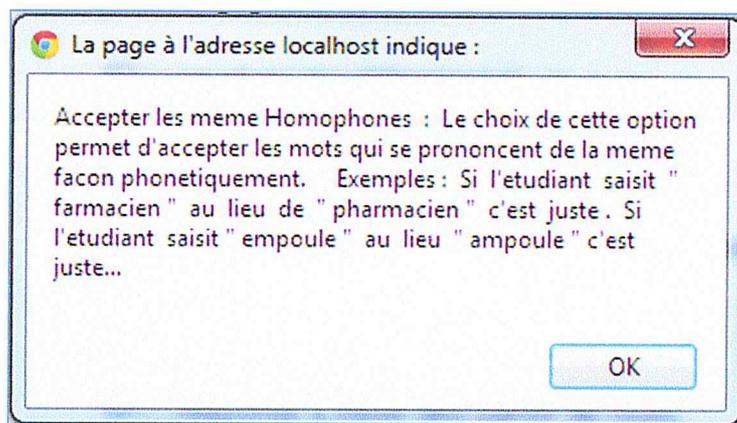


Figure 4.37 : exemple d'aide.

### 5. Comparaison entre l'ancien et le nouveau système :

On a fait un test pour 10 étudiants avec le nouveau et l'ancien système, le teste se compose de 10 questions, un point pour chaque question, les questions sont très simples. Les examinés ont eu entre 3/10 et 6/10 comme note dans l'ancien système alors que les mêmes étudiants avec les mêmes questions on eu entre 6/10 et 8/10 dans le nouveau système.

#### Remarque :

Quand l'étudiant obtient une note complète avec l'ancien système il l'aura certainement avec le nouveau système, mais les réponses correctes dans le nouveau système peuvent être fausses dans l'ancien, par exemple : la réponse est: 'élève', et l'étudiant répond par 'éleve' dans notre système c'est correct alors qu'avant une réponse pareille n'est pas acceptée.

Une importante partie de nos objectifs a été atteinte par ce test qui est l'aide de l'étudiant à avoir la note complète malgré les erreurs d'inattention.

---

---

www

# Conclusion général

---

---

www

---

---

o o o o o o o o  
o o o o o

# Bibliographie

o o o o o o o o  
o o o o o

## Bibliographies:

- [1] Un IHM intuitive et attractive pour le site de E-Learning de l'USDB réalisé par Boulahia Soumia et Harous Lyna Racha (2010-2011).
- [2] Mari Prat « E-Learning réussir un projet » 2<sup>eme</sup> édition (2010).
- [3] Evaluation de plate-formes pédagogiques : [www.refer.ga/IMG/pdf/Etude\\_comparative](http://www.refer.ga/IMG/pdf/Etude_comparative) ( juin 2003).
- [4] Manuel de Hot Potatoes version 6 : [www.framasoft.net/article1590.html](http://www.framasoft.net/article1590.html) 31 mars 200).
- [5] Plate-forme de travail : [http://lycee-levavasseur.ac-reunion.fr/site\\_spip/spip.php?rubrique5](http://lycee-levavasseur.ac-reunion.fr/site_spip/spip.php?rubrique5) (5 septembre 2010).
- [6] Efront : <http://www.efrontlearning-france.com/plate-forme-lms.htm>.
- [7] Moodle: <http://docs.moodle.org/19/fr/question/type/regexp>.
- [8] Le guide de Hot Potatoes 6 - 2004 Fabien Olivry.
- [9] Moodle plugin : <https://moodle.org/plugins/>
- [10] PHP Manual: <http://php.net/manual/fr/function.soundex.php>.
- [11] L'art des soundex : <http://sqlpro.developpez.com/cours/soundex/>.



