МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное образовательное
учреждение высшего образования
ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

Институт математики, механики и компьютерных наук
имени И. И. Воровича

Направление подготовки
Прикладная математика и информатика

Кафедра информатики и вычислительного эксперимента

РАСШИРЕНИЕ ПЛАГИНА ПУБЛИКАЦИИ ЗАДАНИЙ В MOODLE

Выпускная квалификационная работа
на степень бакалавра

Студента 4 курса
Точи Бенджамен Рубен

Научный руководитель:
преподаватель / А.М.Пеленицын

Ростов-на-Дону
2016

Аннотация

Электронная система Moodle используется во многих учебных заведениях (в том числе, на мехмате ЮФУ) для поддержки учебных курсов. На странице учебного курса в Moodle создаются элементы разных типов. Наиболее популярные типы элементов с большой текстовой составляющей это элементы модуля Assignment (Задание) и модуля Page (Страница). Для редактирования текста в обоих модулях по умолчанию используется WYSIWYG-редактор в браузере, который имеет ряд ограничений. Альтернативный способ изменения текста в элементе курса предоставляет плагин, разработанный научным руководителем данной работы. Существенным ограничением плагина было то, что он позволял работать только с модулем Assignment. *Целью* этой работы является расширение функциональности плагина для поддержки модуля Page.

Для достижения поставленной цели в работе были поставлены и успешно решены следующие *задачи*: во-первых, требовалось провести анализ и отбор функций API Moodle для работы с модулем Page, во-вторых, выделенные функции следовало использовать при модификации кода плагина, в-третьих, тестирование изменений следовало провести на локальной машине с развёртыванием необходимого серверного окружения на этой машине.

Работа состоит из пяти частей, включая введение, и списка литературы из четырёх пунктов. Текст содержит шесть рисунков и одну таблицу. Во введении кратко описывается предназначение электронной системы поддержки обучения Moodle. Во втором разделе анализируются проблемы Moodle: формат HTML и необходимость использования WYSIWYG-средств для его редактирования в системе. Здесь же кратко описан формат Markdown – легковесный язык разметки, альтернатива HTML, и обозначен подход к рабочему процессу через упомянутый выше плагин. В третьей части описывается WampServer: программное средство для получения серверного окружения на Windows-машине. Данное средство используется для локального развёртывания Moodle, установки в него и последующей модификации плагина, а также тестирования этого плагина. В пятом разделе приводится перечень наиболее важных функций API Moodle и выделяются те, которые позволяют решить поставленную задачу, а именно изменять содержимое элемента Page. В конце этого раздела приводится аннотированный исходный код для решения этой задачи. В пятом разделе описана схемы взаимодействия с полученной модификацией плагина и приводятся результаты его работы.

Table des matières

1. INTRODUCTION

Moodle est un environnement facilitant la mise en ligne de sites de cours. Moodle permet de diffuser du contenu et de soutenir les interactions entre enseignants et étudiants, à l'aide d'une variété d'outils facilitant la mise en ligne de ressources et d'activités de communication, d'évaluation et de gestion de sites de cours.

1.1. Origines Moodle

C'est en Australie, dans les annees 1990, que Martin Dougiamas, alors a l'emploi de la Curtin University of Technology, a amorce le developpement de Moodle [1]. Le terme « Moodle » etait à l'origine un acronyme pour « Modular ObjectOriented Dynamic Learning Environment ». Cette terminologie est surtout utile aux programmeurs et aux theoriciens de l'education. Mais « moodle » est aussi un verbe qui decrit la façon de flaner paresseusement a travers quelque chose, faire des choses quand cela vous sied le mieux, une maniere agreable d'agir qui mene souvent à la reflexion et a la creativite. Ce terme s'applique donc a la façon dont Moodle a ete developpe, tout comme a la maniere des etudiants et enseignants d'approcher l'apprentissage et l'enseignement dans un cours en ligne.

2. INCONVENIENTS MOODLE

Bien que cet environnement facilite la diffusion des cours des enseignants envers les etudiants,**Moodle** manque encore quelques caractéristiques qui peuvent être améliorés. Par exemple la forme et le moyen de stockage texte. Notons certains de ces inconvenients.

- Pour les tâches d'enregistrement et de stockage du texte a l'aide de l'editeur HTML.

- Le texte est stocké uniquement sur le serveur et Il n'y a aucun moyen de changements de gestion de versions (versioning).
- Le manque des composants de page

Examinons ces inconvenients et les moyens de les surmonter.

2.1. Pour les tâches d'enregistrement et de stockage du texte a l'aide de l'editeur HTML.

L'editeur HTML est un outil nécessaire pour placer le texte sur Internet. Cependant,travailler avec celui-ci est tout à fait inconfortable car il va falloir toujours aller sur l'editeur HTML pour afficher de l'information. Imaginons que nous ayons une multitude d'informations a afficher sur notre site Moodle, Il va falloir trouver un moyen pourqu'on ne puisse pas toujours ,lorsque, nous avons besoin d'afficher une information, aller sur l'editeur HTML . C'est vraiment trop ennuyant. Donc, ce que nous voulons faire,c'est de pouvoir afficher des informations sur la page Moodle ,sans pouvoir le faire sur editeur HTML. Nous devons trouver un autre chemin. Nous allons utiliser un des editeurs de texte, Ceci nous permettra de ne plus recourir a l'editeur HTML pour l'affichage de texte. Il faut aussi bien le rappeller qu'il n'est pas sans risque. Iconvenients :

— La plupart de ces editeurs sont capables d'ajouter(et parfois meme de changer) le texte de l'information prete a etre publiee.

— Il a une faible vitesse lors de frappe de donnees

Parmi les editeurs qui existent, nous allons utiliser Markdown [2] Il est facile a utiliser.

2.2. Markdown

Un éditeur WYSIWYG, WYSIWYG signifie "What You See Is What You Get", c'est-à-dire "ce que vous voyez est ce que vous obtiendrez". En quelques mots, un éditeur WYSIWYG permet d'éditer visuellement du texte, c'est-à-dire sans en passer par l'utilisation du HTML que connaissent bien les développeurs de sites web.

Comme exemples d'éditeurs de texte WYSIWYG, on peut citer des logiciels comme Microsoft Word, Libre Office Writer ou encore des éditeurs spécifiques au Web comme TinyMCE, CKEditor ou Froala. un exemple de Markdown (voir Figure 1, p. 7).

2.3. Le texte est stocké uniquement sur le serveur et Il n'y a aucun moyen de changements de gestion de versions (versioning).

Après réception d'une version complète de la tâche, il est souhaitable de la fixer. A l'avenir, la tâche peut être modifiée de façon répétitive, et il va falloir être sûr qu'il est toujours possible de revenir à la precedente version, ou tout au moins un aspect différent d'elle.Il existe aujourd'hui des logiciels permettant d'appliquer cette fonctionnalité et de fournir un système moderne de gestion de versions . Nous n'allons pas les etudier. Parmi la catégorie de logiciels,nous allons utilise Git SCM pour gerer ces problemes, bien que représenté par le système, il peut bien evidemment fonctionner avec d'autres logiciels de gestion de versions. Le texte que nous allons publier doit etre stocké localement dans .md- (fichiers de format Markdown), pour compenser a un certain nombre des risques évidents liés au stockage du texte exclusivement sur le serveur. C'est de cette facon que nous allons publie le code HTML obtenu à partir de Markdown dans Moodle. Un tel système de logiciel devrait nous permettre de trouver un moyen de stocker md-fichier sur d'autres serveurs. Cela donnera accès à partir d'un endroit où il ya l'accès au réseau, ainsi que l'assurance contre

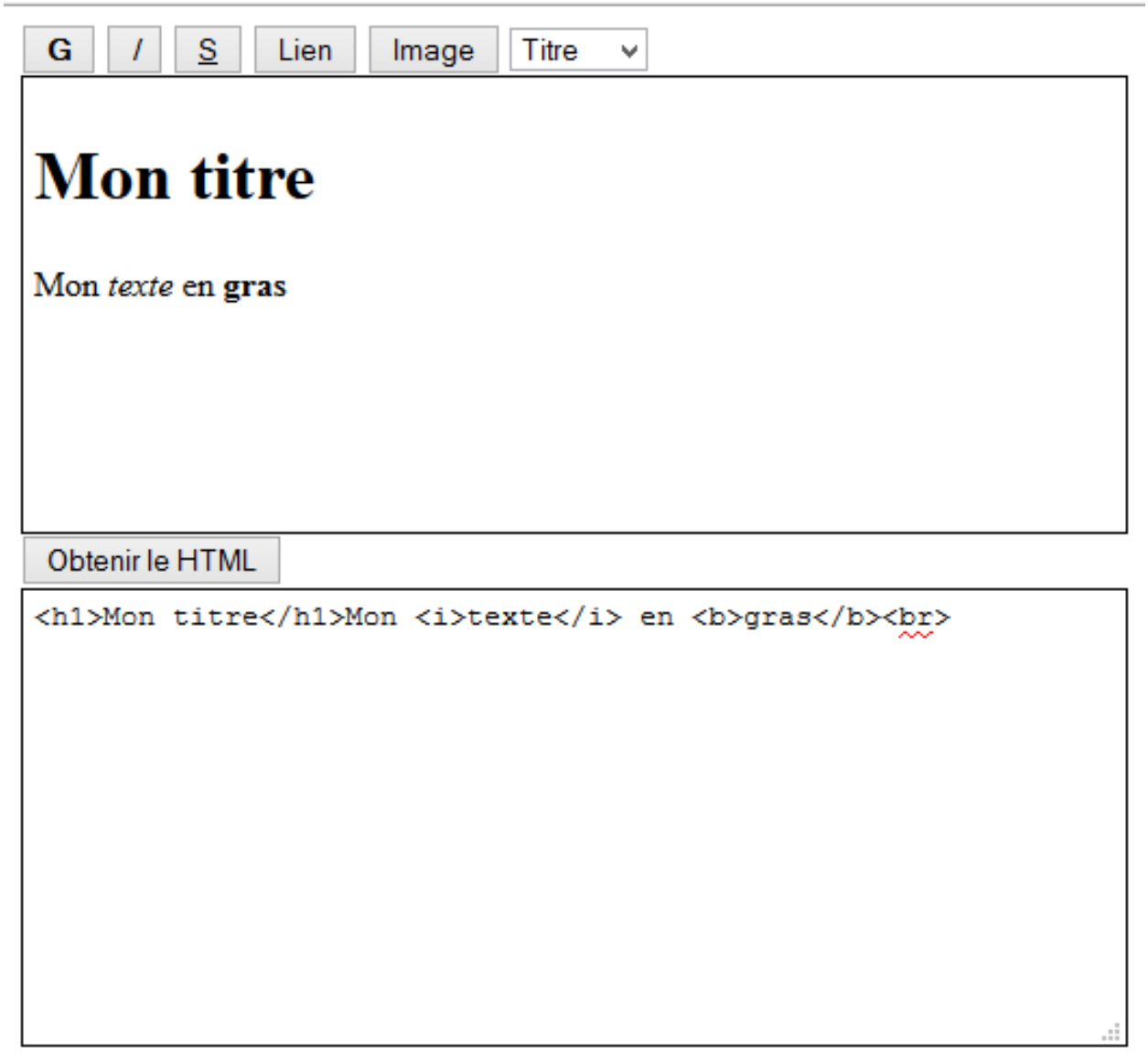

Figure 1 – Editeur Markdown

les cas ci-dessous : -lorsque le disque dans lequel se trouve le texte est en mauvais état, -lorsque le virus informatique bloque l'information, etc. A cet effet, les services de publication de texte(projet) fonctionnent a merveille,stockés a l'aide de Git ou autres systemes de gestion de versions. Les services qui sont particulièrement populaires sont : GitHub (si l'enseignant est prêt à stocker le code source au public) et BitBucket (si nécessaire un accès privé).

2.4. Le Manque des compasants de page

C'est exactement ici que commence notre travail. Jusque la, dans notre Moodle, nous pouvons faire beaucoup de choses cependant nous ne sommes pas en mesure d'afficher l'information sur la page a travers le logiciel Git Bash . Et le but de notre travail,c'est faire en sorte d'afficher cette derniere sur la page. Un travail que devons ajouter sur notre Moodle . Il y a bien evidemment un moyen pour afficher de l'information sur la page Moodle sans pour autant utiliser Git Bash , mais cette maniere ne nous interresse pas. Il suffit d'aller sur l'editeur HTML , et mettre l'information que nous voulons. un exemple de l'editeur HTML (voir Figure 2, p. 9).

Quant a nous,nous allons travailler autrement ,c'est a dire avec Git Bash. Il va falloir rajouter dans le programme **'externallib.php'** de plugin quelques composants de page manquant pour la realisation de notre travail [3].
Le but de notre travail : Elargir le plugin Moodle pour le module d'emploi de publication des composants de page. **Les Taches de faire** :

— L'utilisation des fonctions Moodle API avec des informations sur le module de page.

— L'Application des fonctions Moodle API pour la mise à jour des informations sur l'instance de la page demandée par plugin.

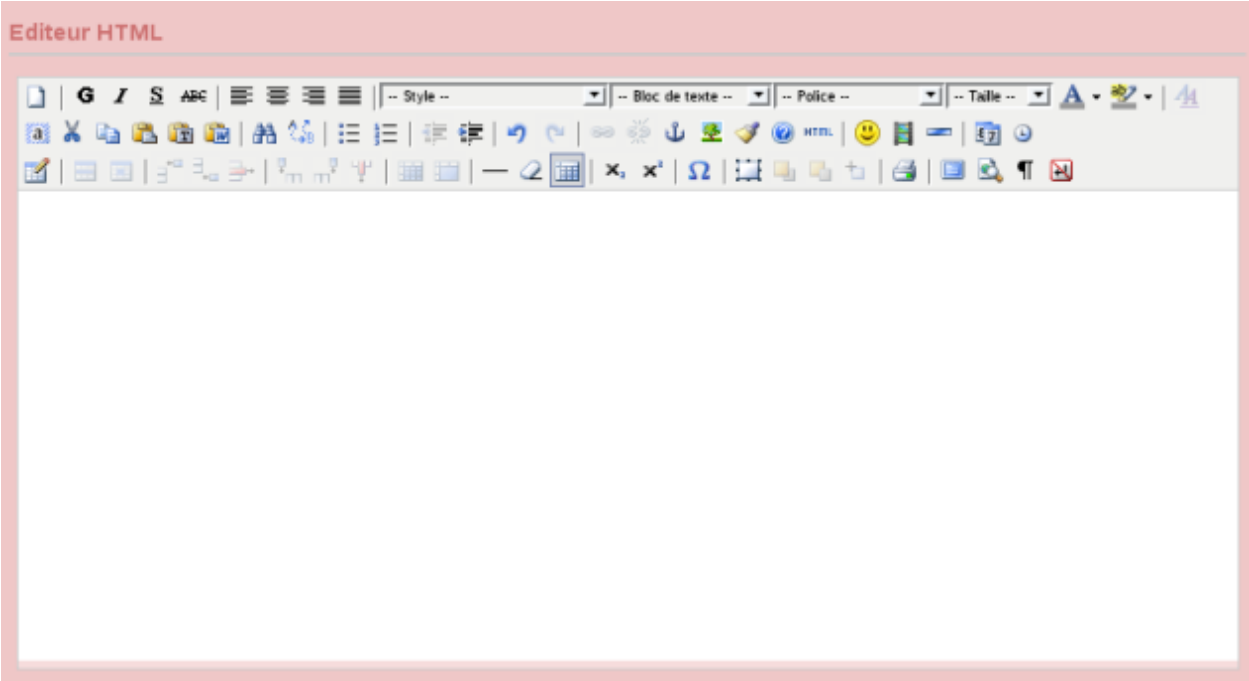

Figure 2 – Editeur HTML

— Système local de déploiement Moodle API et l'installation de L'elargisseent de plugin.

— Le test local de l'elargissement de plugin

3. PLUGIN ET WAMPSERVER

En informatique, un plugin ou plug-in, aussi nommé module d'extension, module externe, greffon, plugiciel, ainsi que add-in ou add-on en France, est un paquet qui complète un logiciel hôte pour lui apporter de nouvelles fonctionnalités. Le terme plugin provient de la métaphore de la prise électrique standardisée et désigne une extension prévue des fonctionnalités, par opposition aux ajouts non prévus initialement apportés à l'aide de correctifs (patchs). La plupart du temps, ces programmes sont caractérisés de la façon suivante :

- Ils ne peuvent fonctionner seuls car ils sont uniquement destinés à apporter une fonctionnalité à un ou plusieurs logiciels ;
- Ils sont mis au point par des personnes n'ayant pas nécessairement de relation avec les auteurs du logiciel principal.

Tous les logiciels ne sont pas capables de fonctionner à l'aide de plugin, le logiciel en question doit être conçu pour pouvoir communiquer avec des programmes extérieurs selon certaines règles que ces derniers doivent respecter pour qu'ils puissent échanger des informations. Les objectifs des auteurs choisissant de concevoir ce genre de logiciel est de pouvoir ajouter des fonctionnalités sans avoir à tout reprogrammer et également de permettre aux utilisateurs d'ajouter leurs propres fonctionnalités de manière indépendante. Idéalement, cette indépendance inclut la possibilité pour le logiciel principal d'évoluer tout en restant compatible avec les plugins existants ; cette condition est cependant loin d'être toujours remplie.

Wampserver WampServer (anciennement WAMP5) [4] est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi que phpMyAdmin pour l'administration Web des bases MySQL.

Il dispose d'une interface d'administration permettant de gérer et d'administrer ses serveurs au travers d'un tray icon (icône près de l'horloge de Windows). Nous aurons bien evidemment besoin de Wampserver pour l'affichage de nos codes en php car c'est sur Wampserver que les paquetages de plugin devront etre installes. precisement sur :

`C:\wamp\www\moodle`

Figure 3 – Wampserver Logo

Il ne faut surtout pas oublier que le programme que nous avons ecrit pour l'affichage de l'information sur Moodle,est ecrit en php, se trouvant precisement sur l'un des paquetages de plugin appele **'externallib.php'**.
Pour l'ouvir ,il va falloir proceder comme ca :

```
C:\wamp\www\moodle\local\ws_assign\externallib
```

Lors du lancement de notre travail,il faut toujours verifier que wampserver soit demarre, autrement il nous affichera un message d'erreur. Pour lancer wampserver,c'est trop simple : allons sur :

```
Demarrer\tous les programmes\wampserver\start wampserver64
```

puis faisons double-cliquer
Lorsque nous lancons WAMP, une icône doit apparaître en bas à droite de la barre des tâches, à côté de l'horloge, comme sur la figure suivante.

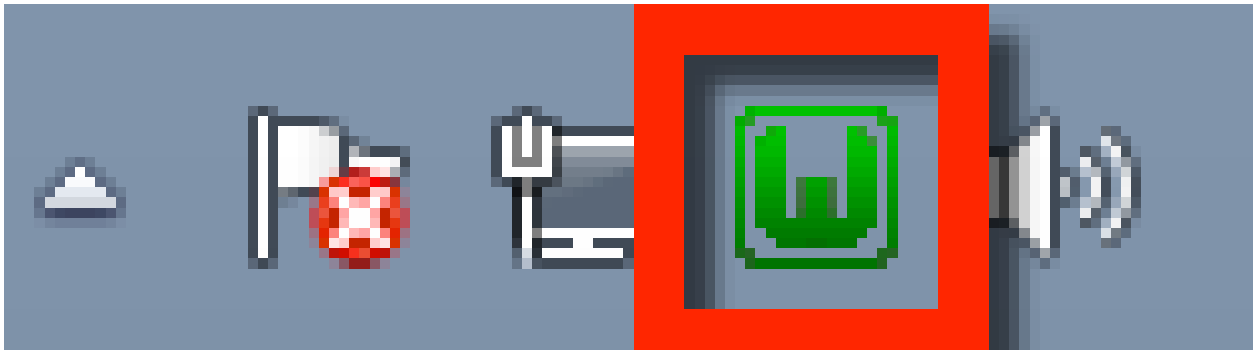

Figure 4 – Icone Wampserver

4. Les parties principales

4.1. Structure d'une base de donnees Moodle(API : Fonctions)

Lorsque nous mettons en place un site web,il va falloir gerer les informations que nous donnent les utilisateurs de ce dernier. Par exemple ; Comment peut-on faire pour recuperer les informations introduites la semaine passee ? ou bien n'importe quelle information introduite par l'utilisateur. C'est Pour cela,Il est necessaire pour nous de comprendre quelques fonctions Moodle pour pouvoir modifier,inserer,recuperer,etc les informations des utilisateurs dans la base de donnees. Il nous sera d'une importance capitale pour l'affichage des informations sur la page Moodle. Voila pourquoi nous allons aborder ce sujet et bien evidemment expliquer l'importance qu'il aura dans notre travail. Quelques fonctions dont nous allons aborder seront divisees en 4 parties, selon leurs importances.

4.2. Fonctions relatives à la localisation

- **get_string()** :va chercher la chaîne de caractères dans le paquetage de langue sélectionné par l'utilisateur
- **print_string()** :imprime la chaîne de caractères dans le paquetage de langue sélectionné par l'utilisateur
- **print_error()** :imprime un message d'erreur, en étant allé le chercher dans le paquetage de langue

4.3. Fonctions interrogeant le statut de l'utilisateur

- **isadmin()** :vérifie si l'utilisateur est un administrateur du site
- **isloggedin()** :vérifie si l'utilisateur est loggué

- **isguest()** :vérifie si l'utilisateur n'est pas en statut "d'invité"

4.4. Fonctions d'impression et de génération de HTML

- **print_header()** :ajoute l'en-tête de la page
- **print_footer()** :ajoute le bas de page
- **print_table()** :imprime une table formattée sur la base d'une structure la décrivant.
- **print_tabs()** :imprime une ou plusieurs rangées d'onglets avec un système pour les utiliser comme barre de navigation.

4.5. Fonctions d'accès au modèle de données

- **get_records()** :récupère une hash d'enregistrements choisis
- **get_record()** :récupère un enregistrement particulier
- **get_records_select()** :
- **get_record_select()** : mêmes fonctions que précédemment, mais permet d'écrire la clause SELECT à la main.
- **get_records_sql()** :
- **get_record_sql()** : mêmes fonctions que précédemment, mais permet d'écrire la totalité de la requête SQL.
- **get_records_menu()** :permet d'obtenir directement le tableau d'options d'une liste déroulante à partir d'un lot d'enregistrements choisis.
- **get_records_list()** :permet d'obtenir les enregistrements à partir d'une liste d'id (typiquement id IN (liste)).
- **insert_record()** :permet d'ajouter un nouvel enregistrement. Retourne le dernier "id" créé en base de données.

- **update_record()** :permet de mettre à jour un enregistrement. L'objet passé DOIT avoir un membre "id".
- **delete_records()** :permet de détruire des enregistrements

4.6. Algorithmes d'ajout(modification)des informations dans la base de donnees Moodle

Dans le point precedent, nous avions vu quelques fonctions Moodle qui nous permettront d'effectuer enormement de taches. Mais dans ce point present,allons voir la maniere dont il faut les utiliser.
Ce point décrit les fonctions disponibles pour accéder aux données dans la base de données Moodle. Nous devons uniquement utiliser ces fonctions afin de récupérer ou de modifier,.. le contenu de base de données parce que ces fonctions offrent un haut niveau d'abstraction et de garantie que notre manipulation de base de données fonctionnera contre différents SGBD.
Tous les appels des fonctions sur ce point sont ceux des méthodes publiques de l'objet global $DB, de sorte que nous aurons besoin d'importer" dans un délai de nos fonctions.
Avant d'utiliser nos fontions , il est important de declarer d'abord l'objet global $DB . Ce qui est juste parce qu'on ne peut pas utiliser une variable si on ne la declare pas,c'est une evidence. Il est declare de la maniere suivante :

global **$DB** ;

L'objet global **$DB** est une instance de la classe moodle_database. Voyons voir quelques exemples de **$DB** avec les fonctions illustrees ci-haut :

```
$user = $DB->get_record('user', array('id'=>'1'));
```

Ici,On recupere un enregistrement dans le tableau ou identifiant vaut 1
Lorsque nous utilisons les fonctions xxx_sql (),les noms de tables doivent etre places entre accolades.

```
$user = $DB->get_record_sql('SELECT * FROM {user} WHERE id = ?',
   array(1));
```

positionnement inconnu :

```
$DB->get_record_sql('SELECT * FROM {user} WHERE firstname = ? AND
    lastname = ?', array('Martin', 'Dougiamas'));
```

placeholders nommes :

```
$DB->get_record_sql('SELECT * FROM {user} WHERE firstname = :
   firstname AND lastname = :lastname', array('firstname'=>'
   Martin', 'lastname'=>'Dougiamas'));
```

Obtenir un seul enregistrement

```
$DB->get_record($table, array $conditions, $fields='*',
   $strictness=IGNORE_MISSING)
```

Obtenir un enregistrement de base de donnees unique comme objet dans lequel toutes les conditions donnees reunies. IGNORE_MISSING signifie rigueur mode compatible, faux retourne si l enregistrement ne se trouve pas, un message de debogage s il est trouve ;
Obtenez un enregistrement de base de donnees unique comme un objet qui correspond a un particulier clause WHERE.

```
$DB->get_record_sql($sql, array $params=null, $strictness=
   IGNORE_MISSING)
```

Obtenir un enregistrement de base de donnees unique comme un objet a l aide d une instruction SQL.
insertion d'enregistrements
La methode pour inserer des enregistrements est appele assez justement, insert_record (). La methode accepte 4 parametres, mais le quatrieme, "en vrac", dans la plupart des implementations est utilise.

```
$DB->insert_record($table, $dataobject, $returnid=true, $bulk=
   false)
```

Inserez un enregistrement dans une table et retourner le champ "id", si necessaire.

```
$DB->update_record($table, $dataobject, $bulk=false)
```

Inscription 4.1. `update_description` fonction

```
public static function update_description($assignment_id,
   $assignment_text)
{

  global $USER,$CFG, $DB;
  require_once($CFG->dirroot . '/mod/assign/locallib.php');

  $params = self::validate_parameters(
     self::update_description_parameters(),
     array(
    'assignment_id'   => $assignment_id,
    'assignment_text' => $assignment_text));

  $context = get_context_instance(CONTEXT_USER,$USER->id);
  $cm = get_coursemodule_from_id('page', $assignment_id);
  $page = $DB->get_record('page', array('id' => $cm->instance), '
     *', MUST_EXIST);
  $page->content = $assignment_text;
  return $DB->update_record('page', $page);
}
```

Modifie un enregistrement sur une page.

Concernant notre travail,il ne s'agit pas d'apprendre par coeur ces fonctions ou bien de les utiliser un peu partout dans nos codes. Elles sont tres importantes ces fonctions mais les fonctions qui vont nous interresser c'est get_record,update_record puisque nous allons les utiliser dans notre travail. Voici un exemple de notre travail (Inscription 4.1).

Voyons ce que reelement font ces codes.

```
public static function update_description($assignment_id,
   $assignment_text)
```

Retourne vrai si la requete a bien ete trouvee et faux si c'est le contraire. la reponse est retournee en forme booleenne.
$assignment_id,$assignment_text ne sont simplement que des variables, Il n'est pas obligatoire d'utiliser seulement ces memes noms.

```
global $USER,$CFG, $DB;
```

Nous avons declare ces variables de maniere globale,c'est a dire qu'il est possible de les utiliser de maniere generale dans nos fonctions.

```
require_once($CFG->dirroot . '/mod/assign/locallib.php');
```

Nous avons recouru au fichier 'locallib.php'. nous avons fait appel a ce fichier dans notre travail. C'est le chemin dans lequel se trouve ce fichier.

```
  $params = self::validate_parameters(
    self::update_description_parameters(),
    array(
      'assignment_id'   => $assignment_id,
      'assignment_text' => $assignment_text));
```

Nous verifions si les parametres enregistres sont valides. nous verifions d'abord la fonction, si c'est ok ,alors nous passons au tableau qui,lui contient 2 variables :(un identifiant et le texte)

```
$context = get_context_instance(CONTEXT_USER,$USER->id);
```

Nous verifions la validation du Contexte. Il est facultatif mais dans la plupart des services web, il devrait etre présent.

```
$cm = get_coursemodule_from_id('page', $assignment_id);
$page = $DB->get_record('page',
              array('id' => $cm->instance), '*', MUST_EXIST);
$page->content = $assignment_text;
return $DB->update_record('page', $page);
```

La page service web.
C'est ici que nous allons pouvoir afficher du texte sur la page Moodle .

5. Schema du travail

Dans ce point,il sera justement question d'enumerer les chemins empruntes pour la realisation de ce travail. Il est necessaire de rappeller que l'affichage de notre travail se fera en Markdown. Nous allons ecrire notre texte sur un fichier en format markdon pour l'afficher dans Moodle grace

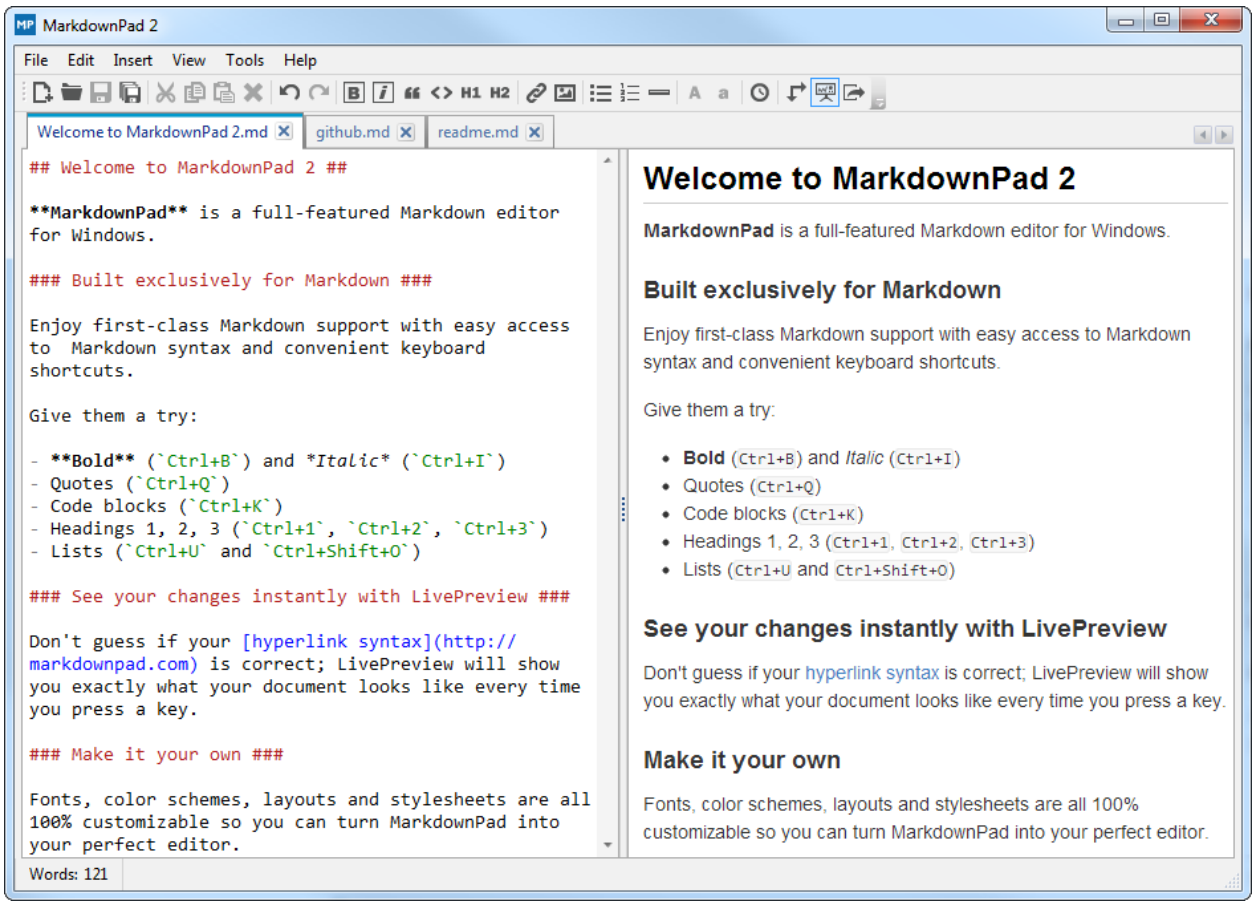

Figure 5 – Un exemple de Markdown

au programme que nous avons cree. Voici un exemple de ce que nous voulons faire en utilisant Markdown (voir Figure 5).

Nous aimerions que le texte ecrit sur le fichier markdown soit traduit pour pouvoir enfin l'afficher sur la page Moodle. Voici le genre de resultat de travail en Markdown que nous aimerions afficher sur la page Moodle (voir Figure 6, p. 19).

5.1. Schéma du stockage et de l'affichage de texte

Le processus de conversion Markdown en HTML et de l'affichage de ce dernier en Moodle sera effectue avec un ensemble de bash-scripts (Il est prévu d'assurer un multi-plateforme en ajoutant des scripts similaires a ceux de PowerShell) En même temps,pour les travaux de l'affichage de

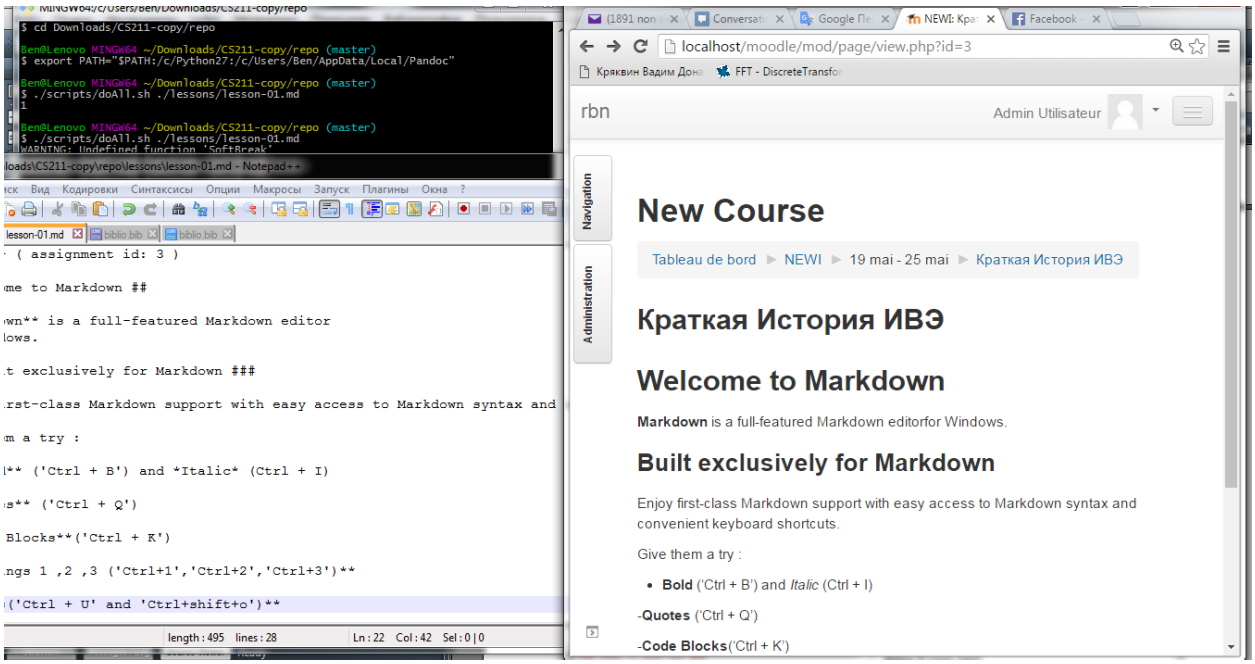

Figure 6 – Texte ecrit en Markdown

Table 1 – Schéma du stockage et de l'affichage de texte

Begin→	fichier-.md →	Script Pandoc →	HTML →	Script serveur →	End
	↓	↓		↓	
	Git SCM	Pandoc		Authentification	

texte dans Moodle, ils sont realises spécialement par le protocole de service Web XML-RPC dans l'infrastructure Moodle (ce système a une API spécialisée pour ajouter de tels services) et de son Python-client. L'ensemble du processus de l'affichage du point de vue de l'utilisateur sur la machine locale peut être illustré par le schéma suivant (voir. Fig.table 1, p. 19), ce qui est tout à fait réalisable avec seulement le script doAll.sh et un seul argument : le nom .md-fichier contenant la tâche a publiér.

Remerciements

Je tenais d'abord a remercier chaleureusement les membres professionnels de **MEXMAT** dans son ensemble. Il me paraît important de vous remercier pour le travail parfait accompli pendant 4 ans.

Merci à vous aussi , Monsieur le Doyen du Departement de l'Informatique et Calcul de l' Experience **(V. S. Pilidi)**, Monsieur le Doyen du Departement d'Algebre et Mathematiques Discretes **(V.D. Kryakvin)**, Monsieur les professeurs **(V.A Nesterenko, V.A. Saveliev, V.N.Bragilevsky)**,

Et plus particulierement a vous, Mon instructeur de travail**(Artem Pelenitsyn)**, d'avoir mis tout en œuvre pour que mon travail se déroule dans les meilleures conditions possibles. Durant ces (4 annees passees), j'ai eu l'occasion d'être associé à votre travail et d'acquérir de nouvelles connaissances et compétences. Celles-ci me seront fort précieuses pour la réalisation de mes projets à venir. Ainsi, le temps, l'attention, l'intérêt que vous avez bien voulu me témoigner n'ont pas été perdus. Ils m'ont donné envie de persévérer dans ce métier pour lequel vous m'avez donné le plus grand respect. Je possède désormais une expérience qui me donne des pistes pour m'améliorer.

Avec toute ma reconnaissance, je vous prie d'agréer, (Madame, Monsieur), l'expression de mes salutations distinguées.

Références

1. Moodle LMS. — URL : `http://moodle.org` (visité le 01/05/2016).
2. Wikipedia : Markdown. — URL : `https://en.wikipedia.org/wiki/Markdown` (visité le 01/05/2016).
3. *Белякова Ю. В.*, *Пеленицын А. М.* Система хранения и публикации текстов заданий в учебной среде Moodle // Современные информационные технологии : тенденции и перспективы ра-

звития : материалы конференции / sous la dir. de Л. А. Крукиер, Г. В. Муратова, В. Ю. Тополов. — Ростов-на-Дону, 2015. — P. 61–63.

4. WampServer : A Windows Web development environment for Apache, MySQL, PHP databases. — URL : `https://sourceforge.net/projects/wampserver/` (visité le 01/05/2016).