

pdf2submission: a Moodle Plugin for Partially Integrating Moodle into Offline Courses Where Handwritten Document Submissions are Required

KITA Toshihiro, USAGAWA Tsuyoshi, NAKANO Hiroshi

Kumamoto University, kita@iied.kumamoto-u.ac.jp

Abstract

We have developed a Moodle plugin called pdf2submission for utilizing Moodle also for teaching at ordinary classrooms. The plugin is based on opensource libraries and tools. With the plugin, handwritten paper documents can be easily accepted in Moodle as if they were submitted online. Though the original version of the plugin is available only for Moodle 1.9, a new version for Moodle 2.2 has been developed.

Keywords

plug-in for Moodle, PDF documents, QR Code, paper-based assignment submission, scanned documents

Introduction

In the environment such as online universities, an LMS like Moodle is regarded as the core part of the learning environment. Also in the teaching that is done at conventional style classrooms without any computers, features available in an LMS can be supposed to be helpful and effective for more efficient and engaging teaching.

pdf2submission block is a Moodle plugin that can handle reports or exam answers which were written on paper sheets. With this plugin, only by converting reports (on which handwritten letters, figures and illustrations were drawn) into a PDF file with a scanner, the reports are automatically accepted in Moodle as if they were uploaded by each student as a Moodle Assignments activity (of the 'Upload a single file' assignment type). Therefore, in conventional style courses using paper sheets for assignment or examination submission, teachers can also manage the handwriting materials on Moodle.

The screenshot example of a pdf2submission block on a Moodle course is shown in Figure. 1.



Figure 1: pdf2submission Block

How to use this plugin

Installation of pdf2submission is straightforward. Just as ordinary block installation, copy the files, and when you login the Moodle site as admin, pdf2submission block will be installed. At each course, you can add a block named pdf2submission as shown in Figure 1.

You can use pdf2submission as follows:

1. As a drop-down list of the assignment names is displayed like Figure 1, each student (who submits a report or something) chooses an assignment to use as the submission destination and clicks on the 'Create' button. Then, Like Figure 2, a PDF file that includes the information of the user ID of the student and the

- assignment in the header part is thereby displayed. The student prints this out and uses it as a cover sheet of the handwritten report. (Or writing or drawing only on the cover sheet and submitting it is also possible.) If you have the Teacher privilege of the course, the cover sheets for all the students is generated and displayed. Teachers can print them out and distribute them to each student in their class to let students use the sheet as the cover sheet for report submission or use it as exam answer sheets.
2. The teacher collects the handwritten reports that students submitted using this cover sheets and scans them with a scanner to convert them into a PDF file. The PDF file is automatically (by a function of the scanner) or manually uploaded to some directory (that is specified at the pdf2submission configuration) of the server that Moodle is working on.
 3. The uploaded PDF files are checked by the Moodle cron job to extract the information of the user ID and the assignment number printed as a QR code at the header part of each page.
 4. According to the extracted information of the user ID and the assignment number, it is submitted to the corresponding assignment activity on the Moodle site. (pdf2submission script copies the PDF file to the directory that the assignment activity uses and also writes submission information into the database table.)



Figure 2: Example of a generated cover sheet header (in PDF format)

How it is implemented and coded

It is assumed that your Moodle is working on a Linux server. The plugin is solely based on opensource tools and libraries indicated in this section.

For generation of the cover sheet, PHP libraries of TCPDF and FPDF+FPDF TPL (Setasign, n.d., Suganuma, 2009) are used in combination. QR code generation is done by a method of TCPDF called `write2DBarcode()` (Asuni, n.d.), like the following:

```
$obj->write2DBarcode('Datato be coded','QRCODE,M','','',38,38,$style,'N')
```

For recognition of QR codes, `zbarimg` command (Brown, n.d.) is used. The CUI version of `zbarimg` command can be easily be built by the following command:

```
./configure --with-x=no --enable-video=no--with-python=no --with-gtk=no --with-qt=no
```

Because, in the case of the PDF files that the authors' scanning machine creates as the scanned result, `zbarimg` command often failed to recognize QR codes, we added some preprocessing such as converting the scanned image into JPEG format by Ghostscript, and burying local small holes in the image by `convert -blur 2` (ImageMagick command) before the decoding process of QR codes by `zbarimg`.

`pdftk` command (Steward, n.d.) is used to divide and merge PDF files. For most of the Linux distributions, you can easily install `pdftk` command using a command like `apt-get install` or `yum install`

In ordinary face-to-face lectures, teachers often require students to submit paper sheets of reports or essays, which are handwritten or printed based on a word processor file. Of course it is possible to let students submit assignments as electronic document files or as postings in an LMS if the assignments can be completed only by writing letters, but there are also needs to make use of "handwriting taste" when letting students submit reports on their ideas about a good cityscape or graphic designing using illustrations. If it is possible to collect also such paper-based reports in an e-learning system, teachers can manage the reports without time-consuming work and can easily set up some kind of peer learning by sharing reports among students.

In addition, if such reports or examination answers are to be preserved on the viewpoint of quality assurance, it is thought that there is much more advantage of storing and managing such reports or exam answers on LMS than storing them into a single-purpose dedicated system.

That is the background in which the authors have developed pdf2submission block for Moodle.

Improvement from the prior version

Not a QR code but alphabet letters and numbers were used to print the cover sheet header in the very early version of pdf2submission and the information included in the header was recognized using an opensource OCR tool, but there was a problem that the around 10% of scanned sheets were not recognizable (i.e., the OCR tool misrecognize the letters), therefore manual correction was necessary to store the scanned documents on Moodle correctly. By adopting QR codes, the rate of unrecognizable headers has become almost zero and even for a large-scale class pdf2submission can be used without problems.

One of the improvements from the previous version is handling of a PDF file that includes plural users' reports. If you scan the reports from many students and make one big PDF file having many pages, pdf2submission automatically divides the PDF file into each user's report as one PDF file to be stored in Moodle separately. Also PDF template insertion feature was added, which is useful for, for example, generating examination answer sheets for each student by inserting an answering form PDF file.

In addition, the former version supported only Moodle 1.9, but the current version of pdf2submission works on Moodle 2.2, the latest version of Moodle.

Usages

Using this plugin, for example, the following usages are possible:

Learning by submitting handwriting reports

You can manage an assignment that requires handwritten or hand-drawn report submissions (e.g., a report on the ideal scenery of your town with illustration) on a pdf2submission-installed Moodle course. The reports collected and stored on Moodle can be nice instructive materials if you share the reports among all the learners and let the learners make peer reviewing with them.

Preserving and rearranging of handwritten examination answers

If you have a number of handwritten documents or examination sheets to be preserved and rearranged, this plugin will be helpful. It was actually tested for 80-student course by converting manually graded examination sheets into PDF files and storing them into Moodle with this plugin, and everything went fine.

Conclusion

The pdf2submission plugin enables online management of the paper documents submitted by learners on Moodle without requiring students to digitize documents and upload them online. It in turn makes the introduction of effective activities such as the peer reviewing of the submitted materials by students possible and easier. It should be also helpful for collecting and storing various kinds of sheets that students submitted through their learning activities. This plugin is potentially applicable to various purposes. The authors would like to hear about nice and practical applications from users.

The previous version of pdf2submission is downloadable at a Japanese developer forum thread (Kita, 2011) of moodle.org and has been already downloaded about 40 times. In some organizations it is in trial or practical use.

The author would like to improve this plugin according to the users' comments and feedback. The current version that works on Moodle 2.2 is going to be released in Moodle Plugins Directory.

References

- Asuni, N. (n.d.). TCPDF Documentation. <http://www.tcpdf.org/doc/classTCPDF-members.html> [viewed 14 May 2012].
- Brown, J. (n.d.). ZBar bar code reader. <http://zbar.sourceforge.net/> [viewed 14 May 2012].

- Kita, T., Nemoto, J., Ueda K., & Usagawa, T. (2011). Development of 2 plugins for using an LMS at ordinary classroom teaching. IPSJ SIG Technical Report, Vol. 2011-CE-111, No. 2, 2011-10-07, pp. 1-4
- Kita, T. (2011). pdf2submission (Tentative release of pdf2submission). (in Japanese) <http://moodle.org/mod/forum/discuss.php?d=186387> [viewed 14 May 2012].
- Moodle.org. (n.d.). Moodle.org: open-source community-based tools for learning. <http://moodle.org/> [viewed 14 May 2012].
- Setasign. (n.d.). FPDF - Import existing PDF documents into FPDF. <http://www.setasign.de/products/pdf-php-solutions/fpdf/> [viewed 14 May 2012].
- Steward, S. (n.d.). Pdftk – The PDF Toolkit. <http://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/> [viewed 14 May 2012].
- Suganuma, D. (2009). TCPDF 4.5.xxx + FPDF PDF . (in Japanese). <http://se-suganuma.blogspot.com/2009/02/tcpdf-45xxxfpdi-121.html> [viewed 14 May 2012].
- TCPDF. (n.d.). TCPDF - PHP class for PDF. <http://sourceforge.net/projects/tcpdf/> [viewed 14 May 2012].

Acknowledgements

The authors would like to thank KAWAMUA Ryo for his assistance in a part of development of pdf2submission.