

1 pdf2submission ブロック [1]

1.1 機能概要

pdf2submission ブロックは、紙に書かれたレポートや試験解答を Moodle[2, 3] 上で扱うために開発した拡張機能である。この拡張機能により、手書きの図やイラストが描かれたレポート、または手書きの試験解答などをスキャナで PDF ファイルに変換するだけで、自動的に Moodle の「課題 (単一ファイルのアップロード)」の提出画面でアップロードしたかのように受け付けることができ、紙媒体を利用して行っている通常の授業でも Moodle の機能を利用して提出されたレポート等の管理を行うことができる。

pdf2submission ブロックを Moodle コース上に設置した例を図 1 に示す。

使い方は次の通りである。

1. 図 1 のように課題名のリストが表示されるので、レポート提出者は、提出先として用いる課題を選び、[実行] ボタンをクリックすると、図 2 のように、ヘッダ部に自分のユーザ ID と課題番号が書かれた PDF ファイルが表示される。これをプリントアウトし、手書き提出レポートの表紙として用いる。(なお表紙に直接、文章や絵を書き込んで、1 ページだけで提出することも可能。)
教師権限でログインしている場合は、学生によって異なる全員分の表紙が生成されるので、これをプリントアウトして授業中に各学生に配布し、レポートの表紙として用いるように、あるいは、試験問題の解答用紙として用いるように等の指示を行う、という利用方法も可能である。
2. 提出用紙を使って学生が書いた紙媒体のレポートを、教員が回収する。一通のレポートを一つの (複数ページの) PDF ファイルへの変換する機能のあるスキャナでスキャンして PDF ファイルに変換する。

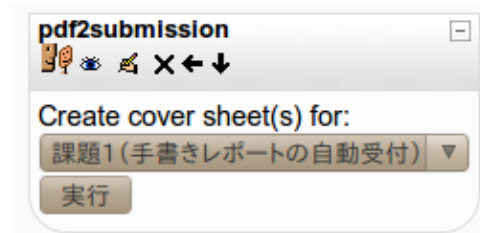


図 1 pdf2submission ブロック

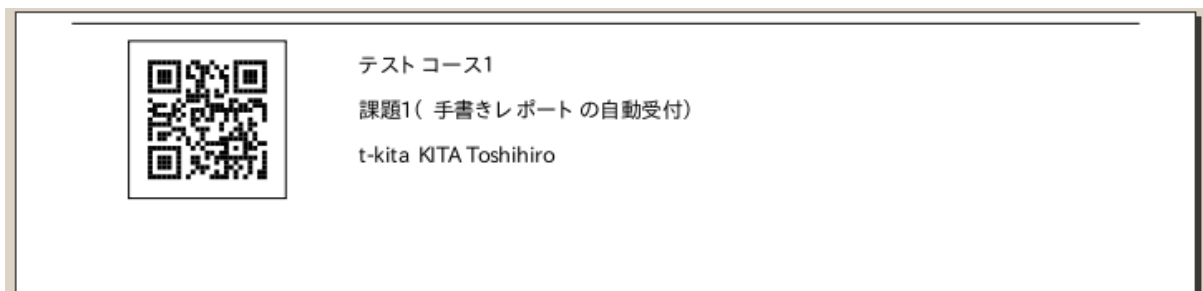


図 2 生成されるレポート表紙 (PDF ファイル) のヘッダ部分の例

生成された PDF ファイルはすべて、自動または手動で Moodle ディレクトリの一部にアップロードする。

3. Moodle の標準機能として (cron ジョブにより) 定期的に行われるスクリプトの中で、PDF ファイルが処理され、ヘッダ部の QR コードに印刷されたユーザ ID と課題番号の情報が抽出される。
4. 抽出されたユーザ ID と課題番号の情報に従って、システムによって Moodle 上に提出される。(Moodle のデータディレクトリの該当場所にコピーされると、各ユーザが提出した提出物として Moodle 上では見えるようになる。)

1.2 実装およびコーディングについて

システム全体として、Moodle が Linux 上で動作していることを前提としている。(Windows サーバに Moodle をインストールした場合の動作確認はしていない。)

表紙の生成には、PHP ライブラリとして、TCPDF[4] と FPDF+FPDF_TPL[5] を組み合わせて用いている [6]。QR コードの作成は、`$obj->write2DBarcode('コード化するデータ','QRCODE,M','','',38,38,$style,'N')` のように、TCPDF のメソッドである `write2DBarcode()` を用いて行っている [7]。QR コードの読み取りには、`zbarimg` コマンド^{*1}[8] を利用した。

スキャナで作成された PDF ファイルを直接 `zbarimg` コマンドに読み込ませると QR コード認識不能のケースが見られたため、一旦 Ghostscript で JPEG 化し `convert -blur 2` (ImageMagick コマンド) で局所的な穴を埋める等の前処理を経てから `zbarimg` コマンドで QR コード画像をデコードさせた。

PDF ファイルを分割したり、統合したりするには、`pdftk` コマンド^{*2}[9] を用いている。

1.3 開発した理由

対面講義でレポート提出を要求する場合、手書きにしる、ワープロで作成にしる、紙媒体で提出させることが多い。

レポートは、e ラーニングのシステム上で提出させて回収するのが、教員側の回収・整理の手間を軽減することや、学生間での相互公開を行って学習効果を高めるためなどにも有効であると考えられる。

また文章だけなら、電子文書ファイルや e ラーニングシステムへの書き込みとして提出させることも可能だが、図形的デザインの学習や、都市景観についてのアイデアをイラストで表現するなどの学習に関するレポート提出をさせる場合は、「手書きの味」を大事にしたいというニーズもある [10]。

また、質保証等の観点から、学習成果物を保存したり、試験解答を保存する必要性も増加してきており、それらが別途そのためだけに構築されたシステムに保存されてしまうよりも、LMS 上で保存し管理できることの利点は大きいと思われる。

*1 コマンド自体だけ (GUI 不要) であれば、`./configure --with-x=no --enable-video=no --with-python=no --with-gtk=no --with-qt=no` で簡単にビルドできる。

*2 多くの Linux ディストリビューションでは、`apt-get install`, `yum install` 等でインストールできる。

1.4 前バージョンからの改善点

過去に開発した pdf2submission の前バージョン [10] では、QR コードではなく、文字を印刷してその情報を OCR を用いて認識していたが、識別不能率が数 % から 10 今回 QR コードの採用したことで、識別不能率がほぼ 0 とし、大規模な授業でも問題なく利用できるようになった。

一つの PDF ファイル内に、複数のユーザの提出物が含まれている場合（全ユーザのレポートをまるごとスキャンして一つの PDF ファイルにした場合など）も、自動的にユーザ毎に切り出して登録することができるようになった。

また、テンプレート挿入機能の追加も行った。これにより、例えば試験の回答用紙として使うための回答欄なども簡単に挿入できるようになった。

参考文献

- [1] 喜多 敏博, 根本 淳子, 上田 公代, 宇佐川 毅 (2011.9) : 通常教室での授業で Moodle を使うための追加機能の開発 –pdf2submission ブロックと barlogin 認証プラグイン–, 日本教育工学会第 27 回全国大会発表論文集 2a-201-03
- [2] moodle.org <http://moodle.org/>
- [3] William H. Rice IV 著, 福原明浩 訳, 喜多敏博 訳/監訳 (2009) : Moodle による e ラーニングシステムの構築と運用, 技術評論社
- [4] TCPDF <http://sourceforge.net/projects/tcpdf/>
- [5] FPDI + FPDF_TPL
<http://www.setasign.de/products/pdf-php-solutions/fpdi/>
- [6] TCPDF 4.5.xxx + FPDI で既存 PDF の読み込みと編集
<http://se-suganuma.blogspot.com/2009/02/tcpdf-45xxxfpdi-121.html>
- [7] TCPDF Documentation
<http://www.tcpdf.org/doc/classTCPDF-members.html>
- [8] ZBar bar code reader <http://zbar.sourceforge.net/>
- [9] Pdftk – The PDF Toolkit
<http://www.pdf-labs.com/tools/pdftk-the-pdf-toolkit/>
- [10] 喜多 敏博, 森川 健太郎, 田口 健治, 根本 淳子, 本間 里見 (2008) : 手書きレポート提出を Moodle で自動受付するシステムの開発, 日本教育工学会第 24 回全国大会講演論文集 3a-A207-01