

PDF ファイルをベースとした電子教材作成支援システム

荒本 道隆[†] 小泉 大城^{††} 須子 統太[‡] 平澤 茂一^{**}
 アドソル日進株式会社[†] サイバー大学^{††} 早稲田大学[‡] 早稲田大学^{**}
 先端 IT 技術センタ IT 総合学部 メディアネットワークセンター 理工学術院 総合研究所

1. はじめに

近年の ICT の進展は、教育、特に e-learning の学習スタイルに大きな変化を与えている。電子教材（デジタル教科書）の分野では、小中高の学校だけではなく大学教育向けにも、その優れた学習効果が期待されている。しかし現在、制作されたすべてのコンテンツがすべての端末で閲覧できるとは限らないという問題がある。そのため、教員側のコンテンツ開発・制作環境と学生側の閲覧環境（PC・タブレット・スマートフォンなど）の組み合わせを問わないマルチプラットフォームが望まれている。

そこで本稿では、電子教材の作成において、PDF (Portable Document Format) の「どんなデバイスでも同一の表示ができる」という特徴を持ち、「OS に依存しない HTML/CSS/JavaScript」を利用してマルチプラットフォーム化を実現する方法について報告する。開発したアプリケーションプログラム（以下単にアプリ）は、電子教材に必要な表現力の追加が可能であると同時に、閲覧ログ機能を向上させたものとなっている。開発したアプリを既存の電子教材「統計入門（仮）」[1]に適用し、その有効性を示す。

2. アプリの基本機能の開発

2. 1. アプリ開発のバックグラウンド

例えば Adobe の PDF というフォーマットを利用すれば、既存のオーサリングアプリが必ず持っている印刷機能を使って PDF を作成することができ、既存の膨大な PDF ファイルを流用することも可能となるが、独自機能を追加するには独自アプリを開発しなければならない。

一方、クライアントで広く使われている 5 種類（Windows/Mac/Linux/Android/iOS）の OS は、それぞれアプリ開発環境が異なり、アプリ配布方法も異なるため、それぞれを別途開発すると多くの開発工数がかかってしまう。

そこで、OS に依存しない開発環境として、HTML/CSS/JavaScript を使用する。今後、もし新し

い OS が出現、もしくは OS がバージョンアップしても、ブラウザが動作して HTML/CSS/JavaScript の標準に準拠していれば本アプリが動作する可能性は高い。

2. 2. 実現方法

ベースとする PDF をすべて画像に変換し、画像をページ単位で表示するアプリを HTML/CSS/JavaScript で作成した。PDF を画像に変換するには、GNU General Public License ソフトウェアである Ghostscript [2] コマンドを利用する。

本アプリの基本機能は、変換した画像ファイルを JavaScript によりページ切り替えを行う非常にシンプルなものである。画像化したものを表示するため、表示端末ごとの解像度の違いによりレイアウトが崩れる心配は無い。

3. 電子教材のための機能の開発

電子教材に必要な機能のうち、以下の機能を実装した。

- ページ内の一部スクロール/切り替え
- 動画の再生と資料の連携
- 別 HTML のはめ込み、指定 URL を開く
- 閲覧ログ収集機能

これらは、大学で授業を行う教員数名からヒアリングをし、既存の試作電子教材 [1] で使用している機能の実現と、別チームでの検証に必要な機能とを優先し選定している。

3. 1. ページ内の一部スクロール/切り替え

ページ内の一部だけをスクロール、もしくは切り替え可能にし、同一ページ内に設問の解答を隠したり、長い説明を埋め込んだりするための機能である。この機能は <div> タグと タグで容易に実現できる。埋め込む部分も画像化することで、本文と同様にレイアウト崩れを防いでいる（図 1 参照）。

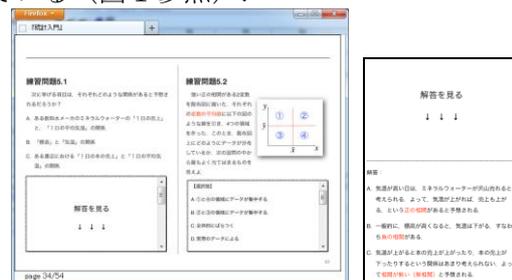


図 1 スクリーンショットと埋め込み画像

The e-learning materials production supporting system based on the existing PDF file.

[†]ARAMOTO Michitaka, Ad-Sol Nissin Corp.

^{††}KOIZUMI Daiki, Cyber University

[‡]SUKO Tota, Waseda University

^{**}HIRASAWA Shigeichi, Waseda University

3. 2. 閲覧ログ収集機能

Web のアクセスログにはユーザーの Web サイト内の移動が記録されるが、ブラウザのキャッシュが使われた部分はサーバにアクセスしないために記録されず、「どこを何度見たか？」を正確に把握できない。また、PDF を閲覧した場合には「PDF をダウンロードした」としか記録されず、どのページを見たかは分からない。

本アプリでは、ページを開いた時と、ページを閉じた（次ページを開いた、もしくは一覧画面に戻った）時に「コンテンツ ID、ページ番号、開いた日時、閉じた日時、開いていた秒数、緯度経度」を Ajax で送信する（図 2 参照）。

```
grep -e viewPage -e viewVod access_log
192.168.150.54 - [29/Nov/2013:14:54:52 +0900] "GET /ppt2/viewPage?contentId=C001&page=1&start=2013-10-29T15:03:08&now=2013-10-29T15:03:29&time=10.8&pos= HTTP/1.1" 404 291 "http://192.168.130.168/ppt2/" Mozilla/5.0 (Windows NT 6.1; WOW64; rv:25.0) Gecko/20100101 Firefox/25.0"
192.168.150.54 - [29/Nov/2013:14:54:59 +0900] "GET /ppt2/viewPage?contentId=C001&page=3&start=2013-10-29T15:03:18&now=2013-10-29T15:03:29&time=7.6&pos= HTTP/1.1" 404 291 "http://192.168.130.168/ppt2/" Mozilla/5.0 (Windows NT 6.1; WOW64; rv:25.0) Gecko/20100101 Firefox/25.0"
192.168.150.54 - [29/Nov/2013:14:55:10 +0900] "GET /ppt2/viewPage?contentId=C001&page=3&start=2013-10-29T15:03:18&now=2013-10-29T15:03:29&time=7.6&pos= HTTP/1.1" 404 291 "http://192.168.130.168/ppt2/" Mozilla/5.0 (Windows NT 6.1; WOW64; rv:25.0) Gecko/20100101 Firefox/25.0"
192.168.150.54 - [29/Nov/2013:14:55:11 +0900] "GET /ppt2/viewPage?contentId=C001&page=4&start=2013-10-29T15:03:29&now=2013-10-29T15:03:29&time=0.5&pos= HTTP/1.1" 404 291 "http://192.168.130.168/ppt2/" Mozilla/5.0 (Windows NT 6.1; WOW64; rv:25.0) Gecko/20100101 Firefox/25.0"
```

図 2 アクセスログの例

ユーザー名の収集は本アプリの対象外としている。ユーザー名の管理は利用する環境ごとに異なり、例えば Web サーバの認証機能を使っている場合、アクセスログの第 3 パラメータにユーザー名が記録される。また、Moodle を使っている場合、Moodle の設定でユーザー情報を付与するようにすれば、URL の最後にユーザー名が自動的に追加される。そのため、アプリ内に認証機能を持つのではなく、アプリを設置する環境の認証機能の利用を前提としている。

スマートフォンでの利用時に、確実に高い精度でログを収集するために、オフライン時の再送処理や、スクリーン OFF 時の秒数カウント停止処理も実装している。

4. 動作検証

PDF 版の「統計入門 (仮)」[1]からアプリを作成し、Windows7/MacOSX/Android4.2/iPad3 の標準ブラウザで開き、動作検証を行った。

HTML の標準的な機能で開発したため、すべての機能が正常に動作している。

しかし、スマートフォンで使用する上で、1 点問題が発生している。今回の PDF を画像ファイルに変換すると、各画像ファイルのサイズで最大 100 倍近くもの差が現れた。スマートフォンでは最も大きなファイルをダウンロードするのに 5 秒以上かかったため、それがそのままページめくりにかかる時間になってしまう。解像度に合ったサイズの画像をダウンロードする機

能を追加し、ページめくりにかかる時間を短縮する必要がある。

5. 閲覧ログ機能の評価

少人数ではあるが、セミナーにて講師と受講者で閲覧ログの記録を行った。セミナーでは講師の画面をプロジェクターで投影し、それを見ながら受講者は自分の PC で好きなページを参照する形式で行った。

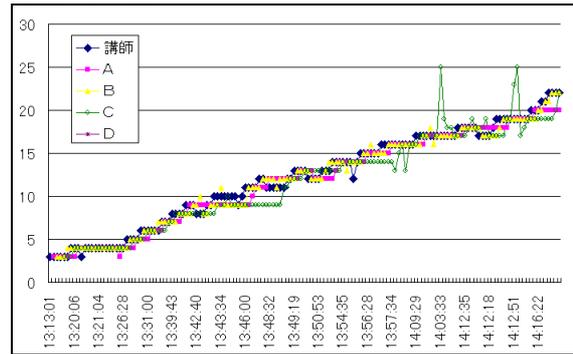


図 3 講師と受講者のページ遷移

図 3 は、縦軸に電子教材のページ番号、横軸に参照時刻をとり、収集した閲覧ログをグラフ化したものである。講師と各受講者がいつどのページを見ていたかを確認できる。また、受講者 A が遅れがちであることや、受講者 C が先行して章の最後まで読んでいることが読み取れる。

6. まとめと今後の課題

本研究では、PDF をベースとした電子教材作成支援システムの開発にあたり、まずはアプリの基本的な部分と閲覧ログを収集する機能を作成し、電子教材の試作、そして様々な端末での評価を行った。その結果、一部の問題は残るものの、マルチプラットフォームを実現できた。

今後の課題としては、教員からの改善案を反映して機能拡張していくとともに、教員が閲覧ログをその場で確認してリアルタイムに受講者の行動を反映する仕組みや、協働学習のための機能を追加していく予定である。

謝辞

本研究の一部は独立行政法人日本学術振興会 学術研究助成基金助成金基盤研究(C) 23501178 の助成による。

参考文献

[1]小泉大城, 須子統太, 平澤茂一, “大学教育のための電子教材の試作 ～タブレット端末向け統計基礎教材～”, 情報処理学会 第 75 回全国大会 講演論文集, pp.4-467-4-468, 2013 年 3 月.
[2]Ghostscript, <http://www.ghostscript.com/>