

B-Learning システムの機能拡充

An improved B-learning system with new features

杉浦友嗣[†], 沢田克敏^{††}, 中村 栄治^{†††}
Tomotsugu Sugiura, Katsutoshi Sawada, Eiji Nakamura

Abstract Blended learning is a way of study environments which is a combination of computer resource aided E-learning environments and traditional blackboard-and-note type in-classroom learning environments. We have already developed a B-Learning system which has been successfully employed in some of our classrooms. This paper describes improvements of our present B-learning system by providing 4 new features, such as (1) fill in the blank type questioning, (2) mobile terminals compatibilities, (3) collecting feedbacks from students on class materilas, and (4) a mailing module. An implementation of these features is discussed in great deal.

1. はじめに

B-Learning (Blended-Learning)¹⁾とは、(1) 従来からの教室授業による学習 (Class-Learning) と(2) コンピュータおよびネットワークを利用した学習 (e-Learning) の2つを組み合わせた学習を指す。この B-Learning のための学習管理システム (以降、B-Learning システムと呼ぶ) としては Moodle がよく知られている。Moodle は多種多様で豊富な機能を持つが、それらを使いこなすことはそれほど簡単ではない。本学においてもその利用が可能であるが、実際に使用している教員はまだ限られているのが実情である。

筆者らはシンプルで使い易いことを狙って、2009～2010 年度にかけ独自の B-Learning システム²⁾を検討し、そのプロトタイプを構築した。そして、これまでに情報通信工学専攻および電子情報工学専攻の実際の授業において有効に利用してきた。このシステムはその機能を限定して、「授業連絡掲示」、「授業教材配布」、「授業映像配信」、「多肢選択式演習問題」、「成績通知」、「定型アンケート」を基本機能として実装している。

本研究では、このシステムの利便性を一層向上させることを目的として、上記の基本機能をベースとして新規機能の検討を行った。新たに検討した機能は、(1) 空欄補充式演習問題、(2) 携帯端末への対応、(3) アンケート

の新規作成、(4) メール配信 の4つである。以下、本論文ではこれらについて、その仕様と構成、動作を述べる。

2. システム環境

2.1 サーバ環境

表1に本システムのサーバ環境を示す。2009～2010年度に構築された既存の B-Learning システムをベースとしているので、そのサーバ環境もそれと同じである。

表1. サーバ環境

OS	Fedora 9
WEB サーバソフトウェア	Apache software foundation
RDBMS	PostgreSQL

また、使用言語は以下の通りである。

- (1) HTML
- (2) PHP
- (3) JavaScript
- (4) CSS

HTML は WEB サイトを記述するためのマークアップ言語であり、PHP 及び JavaScript は動的に HTML 要素を生成するために用いられる。また、CSS は HTML 文書の見栄えを記述するための言語である。

2.2 データベース

本 B-Learning システムではいくつかのデータベースを使用している。表2～表4にデータベース毎に定義されているテーブルを示す。

[†] 愛知工業大学大学院 工学研究科 (豊田市)

^{††} 愛知工業大学 工学部 (豊田市)

^{†††} 愛知工業大学 情報科学部 (豊田市)

表 2. login_system データベース

テーブル名	用途
user_table	各種ユーザ情報
q_login_user	携帯用簡単ログイン機能

表 3. grade_book データベース

テーブル名	用途
four_choice	多肢選択問題の成績データ
ana	空欄補充問題の成績データ

表 4. enq データベース

テーブル名	用途
enq_sbj\$i	アンケートへの回答が可能か否か

このうち表 4 のテーブルは授業科目毎に動的に生成される。このとき生成されるテーブル名は表 4 におけるテーブル名「enq_sbj*\$i*」の「*\$i*」の部分授業科目固有の番号で置き換えた形(例:「enq_sbj1」)となる。

3. 空欄補充式演習問題

学生が自ら学習するためのツールとして演習問題機能が有効であり、既存システムにおいては「多肢選択形式」演習問題機能が実装されている。ここではその機能拡充として「空欄補充形式」演習問題機能を検討した。

レイアウト等は既存の多肢選択形式に近いものを採用し、違和感なく利用できるようにした。1 科目あたりの授業回数を 15 回と想定し、1 授業回あたりの問題数を 6 問に設定している。そのため、1 科目あたりの作成可能な最大問題数は 6 問×15 回=90 問となっている。メニューから「演習問題」を選択すると授業回 (1~15) の選択を行える。授業回を選択すると更に問題番号 (1~6) を選択することができる。

本機能は大きく分けて「問題作成」、「問題演習(解答)」、「問題編集・削除」、「成績参照」の 4 つからなる。以下、これらの機能について述べる。

3.1 問題作成

教員もしくは管理者権限を持ったユーザでログインしている場合、問題がまだ作成されていない問題番号に対して図 1 のように「問題作成」というリンクが表示される。これを選択することで図 2 のように該当問題番号の問題を作成することが可能である。一方、学生権限を持つユーザの場合には「問題が作成されていません」というメッセージのみが表示される。

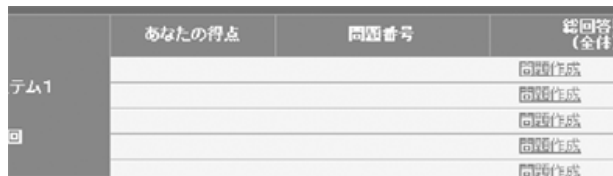


図 1. 問題選択(問題が未作成状態)

問題作成ページでは問題本文や設問数、正答等の登録を行う。問題本文についてはテキスト形式だけでなく、doc ファイルや mht ファイルなどを添付することも可能である。これら添付ファイルは 1 つの問題につき 6 つまで設定可能で、解答時に表示される問題文では添付ファイルそれぞれに対して拡張子に応じた処理が行われる。表 5 に添付ファイルの扱いをまとめて示す。

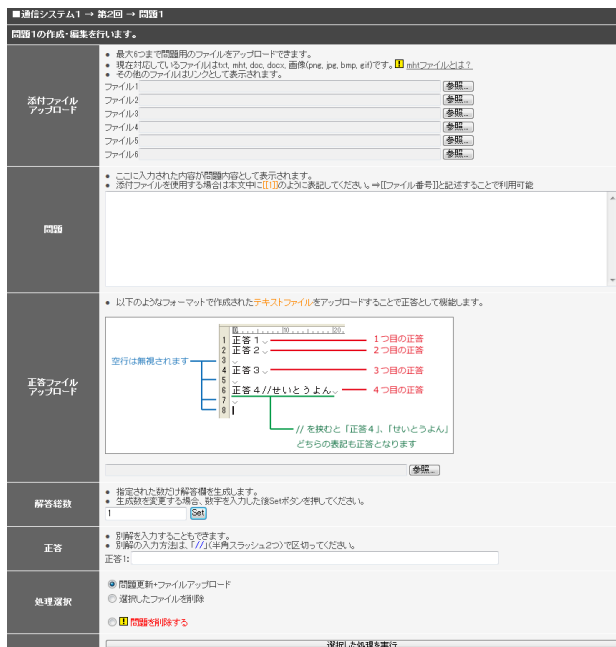


図 2. 問題作成

表 5. 添付ファイル毎の処理

拡張子	処理内容
ワードドキュメント(.doc)	リンクを表示
MHTML ドキュメント(.mht)	インラインフレームとして内容を表示
画像(.jpg, .png, .bmp)	画像を表示
その他	リンクを表示

MHTML ドキュメントとは、HTML 文書と参照される画像や動画まとめた形式のファイルである。これを用いることでワードドキュメントファイルの内容をブラウザ上で直接閲覧することが可能となる。ワードドキュメントを保存する際に MHTML ファイルとして保存することで作成できる。

なお、MHTML ドキュメントファイルの扱いはブラウザに依存しており、Internet Explorer コンポーネントを持つブラウザ以外(firefox や google Chrome 等)では正常に表示できない場合がある。そのため、MHTML ドキュメントファイルを添付する際には対応するワードドキュメントファイルを一緒に添付することが望ましい。

これらの添付ファイルを問題文として表示するには、問題本文を記述する際にファイル番号を二重角括弧で括弧でタグ(例: [[1]] など)として記述する。添付しても上記のタグを記述しなければ表示はされない。問題本文としてはこのタグ以外テキストとして出力される。例えば JavaScript のコードを記述してもその内容のテキストが表示されるだけであり、コードの実行はされない。これは XSS 対策³⁾のためである。

正答を登録する方法には二種類あり、ひとつは決められた形式で記述されたテキストファイルを添付する方法、もうひとつは設問の数を入力してから WEB 上で正答を 1 つずつ入力していく方法である。

正答用のファイルの記述形式としてはテキストファイルの行数が設問番号に対応している。内部的には改行コードを設問の区切りとして認識している。また、半角スラッシュ 2 つ(「//」)を区切り文字として使用することで別解として設定することが可能である。例えば、1 行目に「周波数//しゅうはすう」と記述しておくことで設問 1 の解答は漢字の「周波数」とひらがなの「しゅうはすう」のどちらでも正解として扱われる。そのため、設問に対して適切な正答を入力しておくことができれば解答表現が柔軟になる。しかし、問題作成者の側で正答となり得る単語を全て把握しておく必要があり、漏れていると本来であれば正解であるべきにも関わらず不正解扱いになってしまう。

正答を WEB 上で入力していく際には、まず設問数を入力して「Set」ボタンを選択すると入力された数だけテキストボックスが生成される。この生成されたテキストボックス内に正答を入力していく。この時、先に述べたように「//」を区切りとすることで別解の登録も可能となっている。

問題本文及び正答を設定後、「問題更新+ファイルアップロード」を選択した上で、「選択した処理を実行」ボタンを選択することで添付ファイルのアップロードや問題本文、正答ファイルの生成を行う。これらの生成ファイルは授業科目、授業回、問題番号で階層化されて保存される。

3.2 問題演習

問題が作成されていればユーザ権限に関係なく図 3 のように作成された問題へのリンクが表示され、選択すると図 4 の解答画面へと移動する。

	あなたの得点	問題番号
システム1	2	問題1
	1	問題2
		問題3
		問題4

図 3. 問題選択(作成済み)

図 4. 解答画面の例

解答画面は問題文に解答欄を直接埋め込むのではなく、図 4 のように上部に問題文、下部に解答欄という形で表示される。解答を入力し、「この内容で解答する」ボタンを選択することで解答を行う。サーバ側では入力された解答内容とサーバに記録されている正答を比較して正誤判定を行う。

サーバに記録されている正答ファイルは設問毎に改行コードで、また 1 つの単語毎に「//」で区切られている。そのため、まず正答ファイル全体を読み込んで変数へ格納してから explode 関数により改行コード区切りで配列化する。次に配列化した要素毎に explode 関数を実行し、「//」で区切って配列化を行う。こうすることで正答を二次元配列化することができる。このとき、1 つ目の要素番号が設問番号を、2 つ目の要素番号が該当設問の解答ラベルとして扱われる。学生が解答した内容を同様に配列化した場合、正答配列 correct[\$i][\$j] の第一要素番号と解答配列 ans[\$i] の要素番号は関連付けて扱うことが可能である。例えば、3 つ目の設問に 4 つの解答候補が存在する場合、ans[2] と correct[2][0] ~ correct[2][3] (注: 配列の要素番号は 0 から始まる) のいずれかが一致すれば正解として処理する。

従来システムでの多肢選択形式演習問題では 1 問あたりに 1 つの設問しか存在していないため、各問題の最大得点は等しかった。それに対して空欄補充形式では 1 問あたりの設問数は自由に定めることが可能なため、正解数=得点とすると、問題間での対等な点数比較が難しくなってしまう。そのため、ここでは得点範囲を 0~10 に規格化している。

規格化のための処理は以下のように行われる。

- (1) 正解数を算出
- (2) 正解数をもとに正解率(%)を算出
- (3) 正解率を 10 で割り 0.0~10.0 の実数化
- (4) 実数化した結果が 10 であれば得点を 10 点とする
- (5) 9 以上 10 未満であれば得点を 9 点とする
- (6) (4)、(5) のどちらにも該当していなければ四捨五入を行い、結果が 0 であれば得点を 1 とする(正解率 0%でも参加点として 1 点を与える)
- (7) それ以外は四捨五入した結果をそのまま得点とする

これにより、得点が 0 でなければ問題への解答が一度は行われていることになる。また、単純に四捨五入すると 9.5 以上で得点は 10 点となってしまう、全問正解との差別化が行えないため、10 点を獲得できるのは全問正解した場合のみとしている。

獲得した得点はデータベースに記録される。一度解答した演習問題に再度挑戦することが可能となっている。その場合、データベースに記録されている成績を上回った場合のみ成績が更新される。再挑戦結果の得点がデータベース上に記録されている得点よりも低い場合は更新を行わず、以前の成績が記録されたままとなる。つまり過去にその問題で獲得した最高得点が記録されるという仕組みである。

3.3 問題編集・削除

ログインしているユーザが管理者もしくは教員権限を持っていれば、作成済みの問題に対して「編集」リンクが表示される。これを選択すると「問題作成」と同様の画面が表示されるが、本文や正答等が既に入力済みの状態となっている。そのため、修正したい箇所だけ書き換えればそのまま内容を更新することができる。

入力済みの要素やアップロード済みの添付ファイル等にはチェックボックスが付加されている。これにチェックを入れて「選択したファイルを削除」を選択することでチェックを入れた箇所のデータのみを削除することができる。また、「一括削除」を選択するとチェックの有無に関係なくその問題の関連ファイルが全て削除される。誤って一括削除を行ってしまうと修復するのに非常に手間が掛かるため、少しでも誤削除を減らすために他の選択肢とは離して表示している。

3.4 成績管理

ログインしているユーザが管理者もしくは教員権限を持っている場合、演習問題の関連ページ上部に「成績確認」リンクが表示される。これにより演習問題の解答結果の集計を行うことができる。集計方法として以下の 3

つが設定されている。

- (1) 全成績
- (2) 特定授業回での全問題の成績
- (3) 個人成績

全成績では該当する授業科目に登録されている全ての学生に対して授業回毎の得点及び合計点、平均点を表示する。科目番号のみをキーとして全ての成績データをデータベースから取得している。

特定授業回での全問題の成績では、表示する授業回を選択するとその授業回における全ての学生の成績を表示する。科目番号と授業回の 2 つをキーとして成績データを学籍番号でソートしてデータベースから取得している。

個人成績では、成績を表示する学生の学籍番号を選択すると、その学生の全授業回の成績を表示する。科目番号と学籍番号の 2 つをキーとして成績データを授業回数でソートして取得している。

いずれの表示方法でも、取得したデータをもとに合計点や平均点等の算出を行い、表として出力する。

4. 携帯端末への対応

本 B-Learning システムは基本的に PC からのアクセスを想定して構成されている。しかし、試験結果や授業科目の可否判定等を速やかに確認したい場合には携帯端末からの閲覧が便利である。近年ではスマートフォンも含めて携帯端末が非常に普及しているが、これらは PC と比べると基本的に解像度が低く、画面中に表示可能な文字数も非常に少ない。また、携帯端末でのスクロールは決して使い勝手の良いものとは言えないので、少ないスクロール量で必要な情報を表示することが必要である。そこで、携帯端末からの閲覧を容易とすることを目的として、以下の点について検討を行った。

- (1) 利用可能な機能を限定する
- (2) トップページをメニューリストとする
- (3) 表示ページの下部にメニューリストを表示する
- (4) メニュー選択が円滑に行えるようキー操作による選択を可能にする
- (5) 簡単ログインを可能とする

PC を対象とした基本機能では JavaScript 等のように携帯端末では正常に動作しないことが多い機能を多用している。そこで、ここでは利用可能な機能を制限して、「授業連絡掲示」「試験成績の確認」「ユーザ情報の表示」等のみ利用可能としている。

PC 用ではログイン後に表示されるトップページとして授業連絡が表示されていた。しかし、携帯端末では前述の通り、表示可能な文字数が少ない。そのため、トップページでは授業連絡の表示ではなく、利用可能な機能をリストとして表示することとした。また、メニューリストから各ページへ移動した際、ページ下部にメニュー

リストを設置しておくことで、わざわざトップメニューまで戻って機能を選択するという手間を省いている。

携帯電話では数字キーをナビゲーションキーとして用いることができる。これは数字キーを押下した際に関連付けられたリンクをクリックしたという扱いになるものである。対応している番号を覚えていればメニューを探するためにスクロールするという手間を省くことが可能である。利用可能な機能とナビゲーションキーの対応は表 6 の通りである。

表 6. メニュー内容とナビゲーションキー

機能	対応キー
メニューリスト表示	1
授業連絡	2
成績確認	3
表示科目の変更	4
ユーザ情報の表示	5
設定	6
ログアウト	0

授業連絡及び成績確認は PC 用のものと同程度の機能を持つが、表示可能文字数が少ないという制約上、表示内容を携帯向けに変換している。これについては後述とする。表示科目の変更からは複数の科目に登録されている内容を表示する科目を切り替えることが可能である。ユーザ情報の表示ではログインしている学籍番号や科目名等の設定を確認することが出来る。また、パスワード変更もここから行うことが出来る。設定では各種設定項目の変更を行うことが出来る。現時点では簡単ログインの登録・解除のみとなっている。

4.1 表示の変換ルール

携帯端末では PC と比べると表示可能な文字数が著しく制限される。近年ではスマートフォンの普及により、携帯端末の性能も向上したが、現在でも通常の携帯電話のみを使用している者も多い。そこで、それに合わせて表示内容を PC 用携帯用に変換して、画面内に表示する情報量を可能な限り多くすることを試みた。表示の変換ルールは以下の通りである。

- (1) 全角英数字及びカタカナ、特殊記号類を半角に
 - (2) 2 行以上の改行を削除
 - (3) 4 つ以上連続したハイフンで囲まれた領域を水平線として認識し、<hr> タグに置換
 - (4) 添付ファイルへのリンクを削除
- (1) 全角英数字及びカタカナと一部の記号については php 標準の mb_convert_kana 関数で変換を行える。しかし、用いられる頻度の高い記号の中にはこの関数で対応できない文字も多く存在する。そのため自作関数として変換処理を行う regulation 関数を定義した。この関数では引

数として与えられた文字列に対して mb_convert_kana 関数を最初に呼び出して変換を行う。その後、変換されずに残った記号類に対して個別に置換処理を行う。このようにある関数内で別関数を呼び出すことでコード内では 1 つの関数を呼び出すだけで一連の変換を行うことが可能となっている。

(2) 2 行以上の改行とは、見栄えをよくするために複数の空行を挿入して間を空けることであるが、表示可能文字数の少ない携帯端末ではデメリットが大きい。改行コードとしては「CR」「LF」「CR+LF」の三種類が存在し、プラットフォームによって異なる。C 言語や Java 等のプログラミング言語では、「CR」は「\r」、「LF」は「\n」でそれぞれ表現される。PHP においても同様の表現方法が用いられる。読み込んだ文字列に改行コードが連続している箇所を置換処理によって 1 つの改行コードに置き換える。これを繰り返すことで連続した空行を削除している。また、改行コードそのままでは HTML 上で認識されないため、最後に nl2br 関数により改行コードを HTML において改行を意味する
 タグへ変換し、改行であることを認識させる必要がある。

(3) <hr> タグとは HTML で定義されているタグで horizon、つまり水平線を示す。一方、水平線のような領域の区切りを表現する際にハイフンを繰り返して記述することで実現する場合もある。この場合、携帯端末からでは画面中にハイフンばかりが表示されて非常に見栄えが悪くなってしまう。そこで連続したハイフンで囲まれた領域(----任意の文字列----)を水平線として認識し、その箇所だけ <hr> タグで置き換える。

(4) 資料としてよく用いられるワードドキュメントや PDF ファイルが開けない携帯端末も多い。そのためそれらへのリンクを削除して添付ファイルが存在する旨のみを表示している。

4.2 簡単ログイン

携帯電話機には個体識別情報(以下 UID)と呼ばれるものである。これは機種やキャリアの違いだけでなく、端末 1 つ 1 つに割り振られた固有の値(ユニーク値)である。簡単ログインとはこの UID とログインユーザ名及びパスワードを関連付けることでパスワード等の入力を省くことができる機能である。本システムでは、簡単ログイン機能は DoCoMo、AU、SoftBank の 3 キャリアの携帯電話機のみに対応させている。表 7 にキャリア毎の UID 名称とその値を示す。なお、スマートフォンには UID として利用可能なデータが存在していないため、対象外とした。また、WILLCOM では一般サイト向けに UID の送出手を行っていないことから WILLCOM 製携帯電話も対象外とした。

表 7. UID 名称と値

キャリア名	UID 名称	UID 値
DoCoMo	個体識別情報	大小英数字 7 桁
	I モード ID	(I モード ID)
AU	サブスクライバ ID (EZ 番号)	数字 14 桁_英字 2 桁
SoftBank	端末シリアル番号	英数字 11 桁もしくは 15 桁

UID はキャリアによって名称やその値が異なる。また、先に述べた通り、その値はキャリア毎に固有の値となる。そのため、接続している端末のキャリア情報を最初に取り得し、キャリア毎に UID の取得処理を行うことで簡単ログイン機能を実現している。

セキュリティの観点から言えば、UID は固有の値であるので外部に漏洩した場合、そこから個人情報の特定をすることが可能となってしまう。そこで、データベース上にはこの UID 値をそのまま記録するのではなく、暗号化を施した値を記録している。本システムでは PHP の標準関数で対応可能な MD5 と SHA1 を用いて暗号化を行っている。しかし、どちらの方式においても強衝突耐性の突破が確認されており、万全とは言えない。そのため、暗号化方式の検討も今後必要と考えられる。

5. 新規アンケートの作成

5.1 選択式アンケートと記述式アンケート

既存システムの基本機能では項目内容の決まった定型アンケート、(1) ネット配信授業について、(2) B-Learning システムについて、のみ利用可能であった。しかし、授業科目や教員によって集計したいデータは異なることが多い。そこで、教員が新規にアンケートを作成することが可能な機能を検討した。

本機能では 2 種類の項目要素を用いて柔軟なアンケート作成を行うことが可能である。2 種類の項目要素とは「選択式アンケート項目」と「記述式アンケート項目」の 2 つである。それぞれ例を交えて説明する。

(1) 選択式アンケート項目

選択式アンケート項目とは、その名の通り多肢選択型の項目のことで、アンケートとして一般的なものである。図 5 にその例を示す。項目名にラジオボタンが付加されており、表示されたいずれかの項目のみ選択することが可能となっている。

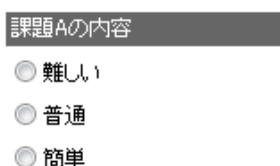


図 5. 選択式アンケート項目の例

(2) 記述式アンケート項目

記述式アンケート項目では、回答者が内容を自由に記入できるスペースが設定されている。図 6 にその例を示す。項目名に対してテキストエリアが 1 つ設けられており、そこに入力された内容を項目毎に記録している。



図 6. 記述式アンケート項目の例

これら 2 種類の項目を自由に組み合わせることで新規アンケートの作成を行うことが出来る。

5.2 アンケート作成

ログインしているユーザが管理者もしくは教員権限を持っている場合、アンケートの選択画面に「新規アンケートの作成」リンクが表示される。これを選択することで図 7 に示す作成画面へと移動する。初期状態ではアンケートタイトルを入力するテキストボックスと項目追加ボタン 2 つが表示されている。ここで項目追加ボタンを選択する毎に各要素が追加されていく。

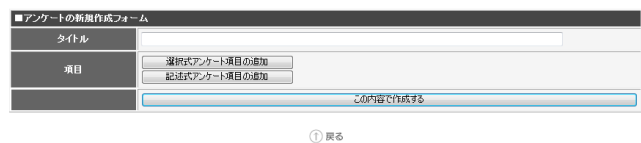


図 7. アンケート作成画面(初期状態)

「選択式アンケート項目の追加」ボタンを選択すると、新たに項目名を入力するテキストボックスと削除ボタン、「選択肢の追加」ボタンの 3 つが生成される。「選択肢の追加」ボタンをクリックすると、今度は新たに選択肢名を入力するテキストボックスと削除ボタンが追加される。これらの生成要素は包括関係にあり、HTML の <div> タグでまとめられている。選択式項目を 1 つ生成した場合における生成要素の包括関係を図 8 に示す。

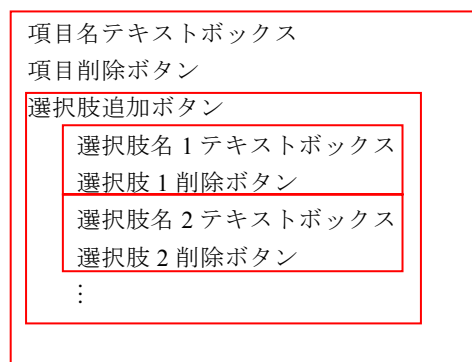


図 8. 選択式項目における生成要素の包括関係

このように追加された選択肢毎に1つのまとまりとなり、全ての選択肢は該当する項目の下位要素としてまとめられている。削除ボタンが選択された場合は削除ボタン自身が所属している <div> 要素とその下位要素を全て削除することで削除機能を実現している。記述式項目の場合は選択肢追加ボタン下がテキストエリアに置き換わっただけであり、基本的な構造は同様である。

図 9. アンケート作成画面

以上の説明を踏まえた上で作成例を示す。図 9 は項目名等を入力した状態(作成前)、図 10 は図 9 の内容で作成した場合の回答画面である。肝心な部分のみを抜粋、拡大して示している。図 9 で追加された選択肢が図 10 ではラジオボタンとともに表示されていることがわかる。また、記述式項目では回答画面でテキストエリアが生成されていることもわかる。

図 10. アンケート回答画面

5.3 作成アンケートへの回答

作成されたアンケートに対して回答を行うと回答内容が図 11 のように表として表示される。一度回答したアンケートに再度回答しようとするとう回答画面ではなく図 12 のように「回答不可能」の旨がメッセージとして表示される。これは、ひとりのユーザに対してアンケートへの複数回の回答を許すと正確な集計が行うことができなくなるためである。

図 11. 回答内容の確認

■実験6について
このアンケートには既に回答しています。
ご協力ありがとうございます。

図 12. 回答内容の確認(回答済みの場合)

アンケートへの回答が可能か否かの情報は授業科目毎にデータベース上に記録されている。ある授業科目で初めてアンケートを作成した際に、enq データベース内に「enq_sbj\$Si」という名称のテーブルデータを自動的に生成する(\$i は科目番号)。表 8 にこのとき生成されるテーブルの定義を示す。アンケートの作成時に授業科目へ登録されているユーザ全てに対して作成するアンケートの識別名とセットで登録を行う。このとき、flag フィールド値は 0 で初期化される。回答時に学籍番号 g_number とアンケート識別名 enq_name をもとにして flag フィールド値を調べる。この flag フィールド値が 0(初期状態)であれば回答可能として回答画面を表示する。flag フィールド値は回答内容を記録する際に 0 から 1 へと書き換えられる。これを用いて、1 の場合は回答済みとして図 12 のメッセージを表示する。

表 8. 生成テーブルの定義

フィールド名	g_number	enq_name	flag
フィールド値	文字列	文字列	整数(0or1)
意味	学籍番号	アンケート 識別名	回答の可否

5.4 要素の動的な生成

最初に述べたように本 B-Learning システムでは PHP 及び JavaScript を利用して動的なページを構成している。PHP だけでは動的な要素の生成を行いたい場合にページのリロードが必要となる。そのため、今回のアンケートのように選択肢や項目をいくつも追加したい場合には、ボタンを選択する度にリロードが発生して煩わしいものになってしまう。そこで、JavaScript を用いて DOM ノードへの操作を行うことでリロードを発生させることなく要素を動的に生成することを実現している。

DOM とは、Document Object Model の略称であり、HTML 文書及び XML 文書をアプリケーションから利用するための API のことである。DOM により文書をツリー構造(階層構造)で表現することができる。このツリー構造化した際の枝葉となる要素をノードと呼ぶ。HTML タグの属性値も同様にノードと呼ばれる。JavaScript には

このノードを追加・削除するメソッドが用意されているため、これを利用して処理を行う。表 9 にノードの種類をまとめて示す。

エレメントノードの追加は `createElement` メソッドによってエレメントオブジェクトを生成し、生成したオブジェクトを引数として `appendChild` メソッドを実行することで行う。テキストノードの追加は `createTextNode` メソッドに挿入するテキストを引数として実行することで行う。属性ノードについてはエレメントノードを追加する際に、`createElement` メソッド実行後、`appendChild` メソッドの実行前に「オブジェクト名.属性名 = 属性値」の形式で設定することが可能である。

表 9. ノードの種類

ノードの種類	ノードの名前	ノードの値
エレメントノード	タグ名	Null(なし)
テキストノード	#text	テキスト内容
属性ノード	属性名	属性値

ノードの削除は子ノードを削除する `removeChild` メソッドによって行う。このメソッドでは「親オブジェクト.`removeChild`(削除ノード)」という形式で記述することで親オブジェクトから削除ノードに指定したノードを削除する。親オブジェクトの指定に要素の ID や Name を取得する `getElementById` メソッド及び `getElementByName` 等を利用することで特定の要素のみを削除することが可能である。

6. メール配信

本 B-Learning システムにおける掲示等の更新通知が必要な場合(学生に更新内容を緊急に連絡したい場合には)は本学の学内連絡掲示板機能を持つ CO-NET を通じて行っている。しかし、教員側から見れば B-Learning システム内で掲示を更新してから CO-NET でその旨を連絡するという二度手間状態となっている。学生側からみても同様で、CO-NET で確認してから B-Learning システムへログインし、掲示を確認するという状態である。また、レポート未提出者の呼び出し等、学生へ直接連絡を行いたい場合もある。これらに対処するため、本システム内から学生宛に直接メール配信を行うことが可能となるようにした。

メール配信を行うにあたり、メールアドレスの登録が不可欠である。そのため、ユーザ情報の変更によりメールアドレスの登録フォームを追加した。ここで入力されたメールアドレスは `login_system` データベース内のテーブル `user_table` に記録されるようになっている。

6-1 送信メールの形式

PHP においてメール送信を行う方法にはいくつか種類があるが、ここでは PEAR の Mail パッケージを用いた方法を採用している。PEAR とはパッケージ単位で分類された PHP の追加ライブラリ集であり、必要なパッケージ単位でインストールが可能となっている。本サーバ環境では図 13 に示すコマンドによってメールパッケージをインストールすることが可能である。1 行目は PEAR 自体をインストールするコマンドであり、過去に別パッケージ等を利用する際に導入済みであれば不要である。2 行目の `-a` オプションはそのパッケージが依存している前提となるパッケージを全てインストールするためのものである。これによりパッケージをインストールした筈なのに動作しない、という事態を避けることができる。

```
yum install php-pear
pear install -a mail
```

図 13. Mail パッケージの導入用コマンド

Mail パッケージを用いたメール送信では `factory` メソッドによって Mail クラスオブジェクトを生成する。生成した Mail オブジェクトで `send` メソッドを用いることで送信処理を行う。Mail オブジェクトではどのような送信方法を用いるか等のパラメータを設定する必要がある。今回送信するメールの設定を表 10 に示す。バックエンドは送信方法の種類、host 以下は SMTP 接続による送信を行う場合に用いる SMTP サーバの設定である。

表 10. メール送信のパラメータ設定

パラメータ	設定値
バックエンド	SMTP
host	mailsrv.aitech.ac.jp
port	25
auth	false

表 11. 送信メールのヘッダ設定

ヘッダフィールド名	設定値
From	任意の文字列：送信者のアドレス (default:EV B-Learning システム) 例：AIT B-Learning システム <sender_b-learning@aitech.ac.jp>
Subject	任意(フォームで入力したメールの件名)

送信するメールのヘッダ設定を表 11 に示す。From フィールドは送信者のメールアドレスを意味し、送信専用

アドレスを設定している。メール送信時には送信専用のため返信ができない旨を本文に付加して送信を行う。

6.2 配信フォーム

メニューから「メール配信」を選択することで図 14 に示す配信フォームが表示される。ここでは件名や本文の入力、送信対象の選択等を行う。科目名の項目はログイン中の科目が自動的に選択されるため変更は不可能となっている。また、送信者名は初期値として B-Learning システムからのものだとわかるように設定しているが、変更は可能である。送信対象の項目ではその授業科目に登録されている全学生宛に送信するか、選択した一部の学生のみを送信するかを選択することができる。これは更新通知のような全体宛のものと学生呼び出しのような一部宛のものを考慮したためである。内容を入力し、「確認」ボタンを選択することで、送信内容の確認画面を表示する。ここでメール内容の誤りに気付いた場合には「修正」ボタンにより図 14 の配信フォームに戻って修正を行うことができる。このとき入力していた内容はリセットされない。そのため、修正箇所のみ変更すれば良い。

図 14. メール配信フォーム

6.3 定型文の送信

掲示内容の更新等を通知する場合、その配信メールの内容は毎回似たようなものになると考えられる。先に挙げた配信フォームでは自由に内容を記述できる半面、同じような内容を毎回入力して送信する場合には非常に手間が掛かってしまう。そこで、授業掲示板の上部に定型文を送信するためのボタンを設置した。図 15 の「更新を通知」ボタンを選択することで図 14 と同様の配信フォームへと移動するが、図 16 のように予め本文等が入力された状態となっている。また、「定型文の送信」としているが微調整を行うことも可能である。挿入される定型文は更新された科目名と併せて更新された旨とサイトアドレスが記述されている。



図 15. 更新通知ボタン

図 16. 定型文の配信フォーム

6.4 送信処理

送信処理は関数化されており、自作定義関数である `mail_sending` 関数を呼び出すだけで送信先の取得から実際の送信処理までを行う。`mail_sending` 関数は送信処理が正常終了した場合 `true` を、何らかの要因で失敗した場合は `false` を返す関数である。`mail_sending` 関数は内部で送信先を取得する関数 `get_recipient` 関数を呼び出している。これにより 1 つの関数を呼び出すだけで一連の処理を実行している。`get_recipient` 関数は科目番号を引数として送信先のリストを配列として取得する関数である。この関数では正常に取得できれば送信先のリスト配列を、失敗したら `false` を返す。科目に登録されている全学生を取得する場合は、科目番号をキーにして `login_system` データベースの `user_table` テーブルから学籍番号とメールアドレスを組にして取得する。指定した一部の学生のみを取得する場合は、科目番号と指定した学生の学籍番号リスト配列をキーにして、学籍番号と一致したデータ列のメールアドレスのみを取得する。

`get_recipient` 関数によって正常にアドレスを取得できた場合、処理を続行して送信データのエンコード、メールオブジェクトの生成、パラメータの設定、送信処理を順に行う。オブジェクト生成とパラメータ設定は 1 回行えば良いため、`send` メソッドによる送信処理のみ `for` 文によって繰り返す。また、処理の最初に一時的にタイムアウト時間を無制限に設定し、処理終了時にタイムアウト時間を再設定する。これはメール送信の試行実験において送信先が多い場合にタイムアウトという事態が発生したためである。本来なら、200 名程度宛の送信でタイムアウトが発生することはあまりないはずなので、この送信処理は見直す必要もある。

7. むすび

本論文では既存の B-Learning システムの基本機能をベースに、ユーザにとってより利便性の高いシステムとすることを目的として新規機能の検討・実装を行った。新規実装した主な機能は (1) 空欄補充形式演習問題、(2) 携帯端末への対応、(3) 新規アンケートの作成、(4) メー

ル配信の 4 つである。

既存の多肢選択形式の演習問題に加えて空欄補充形式のものを追加したことで演習問題の幅が広がった。これにより学習効果の向上が見込まれる。なお、現在は多肢選択形式と空欄補充形式で別システムとなっているが、両者を統合し、問題の作成時に形式を選択できるようにすれば利便性がより向上すると考えられる。

携帯端末への対応を行ったことで外出先等、手元に PC やインターネット環境がなくても容易に掲示の確認を行えるようになった。現在ではクライアントサイドスクリプトである JavaScript を多用している演習問題機能やアンケート機能は利用できないが、JavaServerPages などのサーバサイドスクリプトに置き換えることで利用可能とすることを検討するとより利便性が向上するものと考えられる。

新規アンケートを作成する機能を実装したことで定型アンケートのみだった従来と比べて、必要なデータを集計できるようになった。

メール配信機能を実装したことで、教員が学生に直接連絡を行えるようになった。また、定型文により容易に更新通知を行うことも可能となった。この機能と携帯端末への対応とを組み合わせることで成績の確認などを即

時に行うことができる。体を見直す必要も考えられる。

今後の課題として、今回新規に実装した機能を実際の授業において利用していくことでその有効性の確認及び調整等を行っていく必要がある。また、既存の機能も含めた各機能の内部処理のモジュール化と統合を行い、処理の再利用性の向上も必要と考えられる。

文献

- 1) Kaye Thorne, Blended Learning, Kogan Page, 2003
- 2) 久保田優, B-Learning システムの構築, 愛知工業大学大学院修士論文, 平成 23 年 2 月
- 3) Seth Fogie, et al., XSS Attacks: Cross Site Scripting Exploits and Defense, Syngress, 2007
- 4) 大垣靖男, PHP ポケットリファレンス, 技術評論社, 平成 17 年 10 月
- 5) 古籬一浩, JavaScript ポケットリファレンス, 技術評論社, 平成 19 年 8 月
- 6) 藤岡功, HTML&CSS ビジュアル・リファレンス (改訂版), エムディエヌコーポレーション, 平成 19 年 12 月

(受理 平成 24 年 3 月 19 日)