

PHP・MySQLの復習用教材サイトの構築について

原田 知佳

目 次

1	はじめに	1
1	1 テーマを選んだ理由	1
2	2 使用対象者	2
3	3 類似するものの有無	3
2	2 制作物についての構想	4
1	1 制作物について	4
2	2 ブラウザの指定	7
3	3 参照した資料	7
4	4 構成と実装する機能	8
3	3 本論	11
1	1 枠組み	11
2	2 インデックスページ	11
3	3 ログイン機能	11
4	4 問題一覧ページ	13
5	5 問題ページ	14
6	6 登録情報の確認・パスワード変更	15
7	7 管理者用ページ	16
4	4 結論	16
1	1 アンケートによる評価	16
2	2 修正した点としなかった点	21
3	3 自己評価と今後の課題	24
5	5 おわりに	25

1 はじめに

私は卒業制作として、PHP⁽¹⁾とMySQL⁽²⁾の復習用教材サイトを作成した。これは、福田ゼミの必修授業である『プログラミング演習 2』と『Unix 演習 2』の授業内容の復習ができる教材である。これはインターネット上にアップロードするので、学内だけでなくインターネットにさえ繋いでいれば自宅でも外出中でもアクセスでき、復習が可能である。この教材を課題や自主学習として学生が使用することで、PHP やMySQLをより理解してもらうことを目的としている。

(1) テーマを選んだ理由

福田ゼミの基本的なテーマは「人の役に立つものを作る」である。テーマを決める時点で私は情報教育アシスタントとして1年間授業のサポートに入ることが決定しており、実際に受講者の様子を見られるのでそれを活かし受講者に使ってもらえる教材を作りたいと考えていた。そこで福田先生と相談し、私は先生によるプログラミングの授業の後期分の復習用教材サイトの作成を提案された。

今年度まではプログラミングの授業は先に述べた2つの授業合わせて週に2回受けられたが、来年からカリキュラムが変わり、2回生の時に現在前期にやっている内容を週に1回、3回生で現在後期にやっている内容を週に1回の受講となってしまった。プログラミングは決して簡単ではなく覚えることも多い。週に2回の受講でも授業内容を自分のものにするのに必死であったのに、それが半分となれば今まで以上に大変になり、それだけ次の授業まで期間が開けば忘れてしまいやすくなるため、復習が必須になったのである。

そこでそれを学生の自主学習に任せると、やらない人が出る可能性

があり、そもそも授業内容が理解できていなければ自主学習さえ難しい。PHPを記述するだけならテキストエディタさえあれば可能だが、動かすには自宅のパソコンで環境の構築を行わない限り大学の3201教室かサポートルームのMacintoshでないと不可能と作業の場が限られるので、卒業制作として教材を作成することによりそれらの問題を解決する。

後期の授業で使用する電子書籍による教材⁽³⁾を一昨年に栗津さんが『PHP教材の電子書籍化について』にて作成しているが、私自身実際に昨年、ところどころ説明が不足しているため教材通りに書いてはいるけれどもどういう動きをするのか、なぜその関数がそこで必要なのかしばらく分からないまま使っていたことがあり、特に最終章のブログの作成についての説明は簡略化されすぎていて何がどうと理解するのに苦労した。それは私だけではなく受講している学生も同じだったようで、その関数を使い始めてしばらく経ってから「これはいったい何をするものなのか」と聞かれることが一度だけではなかった。そのような経験を踏まえ、本教材ではヒント等で補足情報を加え、電子書籍の教材では分かりにくい箇所を補うことも目的とする。

(2) 使用対象者

使用対象者は福田先生と、人文情報学科で先生の『Unix演習1・2』を受講する3回生の学生である。この教材では利用する学生の課題の進行状況を管理者が確認できる機能の実装のために、IDとパスワードを使ったログイン機能を付けているので先生と受講者以外は利用できない仕様となっている。管理者用ページ(課題の進行状況が見られる等)を作成し、管理者は先生なので先生のIDでログインされた時のみその

ページへのリンクを表示させる。

あくまで復習用の教材であるので授業内容をそれなりに理解していることを前提とし、ヒント等の補足情報を書いているがそれでも分からない場合は、学生に電子書籍の教材や自分が作成したファイルを見てもらうようにする。

(3) 類似するものの有無

過去の福田ゼミの卒業生達がすでに多くの卒業制作で PHP や MySQL についての教材を作成していたので、私は「PHP と MySQL の両方に触れているか」と「オンラインの教材であるか」の2点で絞って調べることにした。これらの条件で該当した卒業制作は1つで、2006年度卒業生である中村さんの『PHP,MySQLを組み合わせた入門用教材について』だった。

上記の卒業制作は、題目にある通り PHP と MySQL 両方の穴埋め問題を掲載した Web サイトである。私も基本は穴埋めの問題を想定していたのでその点だけ見ると類似しているように見えるが、中村さんの教材の問題内容はあくまで PHP と MySQL の基本的な使い方を学ぶものであり、授業で作成する制作物の内容自体には触れていないので、PHP と MySQL を応用している授業内容そのものの復習はできないことから私のテーマと類似しているものはないと言える。

なぜ今までに多くの教材が卒業制作として作成されながら、授業内容に即した復習用の教材がなかったのか。それは第1節(p.1)で述べたように授業が週に2回の受講であったため、今ほど切実にオンラインの復習用教材が必要でなかったからであろうと私は考える。

本教材を作成することにより福田先生にとっては復習用教材の作成の

手間が省け、受講する学生にとっては理解を深める助けになるので、福田ゼミのテーマに沿うことができることから私は本教材の作成の決定に至った。

2 制作物についての構想

(1) 制作物について

本教材は、PHP と MySQL を使用した Web サイトとする。なぜならフォームに入力された解答を受け取って行う正誤判定に PHP が必要であり、ログイン機能の実装のために ID とパスワードをデータベースで管理するからである。

問題内容は、授業内容の復習用教材サイトなので実際授業で作成したファイルと粟津さんが作成した教材を元に出題する。出題形式は基本は穴埋めとし、ところどころプルダウンメニューやチェックボックス、ラジオボタンも使用する。複数の形式を採用したのは穴埋めのみだと解答に時間と手間がかかり、穴埋めでなくてもいい箇所をプルダウンメニュー等にする事で、少しでも利用者の負担を減らせるようにする狙いがあるからである。また、形式を1つではなく複数にすることで、できるだけ利用者を飽きさせないようにすることも狙っている。授業は内容上毎回は一気に進められないため課題も一度に多くは出ないと思うが、それでも1つの問いに対しての解答欄が多いので、私は全てが穴埋めだと利用者が飽きてくるのではないだろうかと考えた。私は繰り返し復習に使ってもらえるようにするためにも、少しでも飽きさせないようにすることが必要だと思い至ったのである。

(i) オンラインの教材にする理由

オンラインの教材にする理由は、先生にとっても学生にとっても紙と比べて手間が省けると考えたからである。紙にすると先生は課題を出すたびに該当ファイルを学生の人数分印刷しなければならず、その時に学生が休んでしまうとその学生は課題を受け取れず次の授業まで復習ができなくなる。正誤判定までやろうとすれば、先生が全て添削するか学生が栗津さんの教材や自分が作成したファイルを元に確認するしかなく、前者だと先生に多大な負担がかかり、後者だと学生の時間と労力を費やすことになるのである。

しかしこれをオンラインの教材にすれば、どれも解決する。課題の指定は「 (ページ名)の 問目まで」と指示するだけで済み、学生が休んだ場合も一緒に受講している友人がいればその人から教えてもらい自宅や大学で課題ができる。友人がいなくても、課題ができていなければ先生から注意のメールが届くので学生は課題の存在を知ることが可能である。正誤判定は教材上で「正解・間違い」の表示をすれば十分である。

(ii) ログイン機能をつける理由

ログイン機能を付ける理由は第1章の第2節(p.2)で述べたように、学生の課題の進行状況を確認できるようにするためである。その学生ごとに全問正解しなければ次のページの問題内容を表示できないようにし、アクセスできる最新ページを課題ページとして、先生だけがアクセスできる管理者用ページにそのページ名とページ内で正解済みの問い番号を表示することで、どの学生が課題をきちんとやっているか・やっていないかの確認が可能となる。

課題をやっていない学生には注意のメールを送れるようにしたいとの先生からの要望により、私は個人情報の観点から学生ごとに大学から付

与されている大学用のアドレスを使用することにした。そのアドレスは「学籍番号+アルファベット2文字@hi.otani.ac.jp」なので、@以前をIDとして学生に登録してもらうことで、メールアドレスを先生が把握できるようにする。そのIDはWindowsのアカウントにログインする時のIDと同じなので、学生にとっても忘却を防ぐことに繋がる。

更に最終ログイン日時も記録することで、ちゃんと学生が教材を利用しているか先生に分かるようにもする。

(iii) 出題形式の選択理由

<textarea>の中にプログラムを丸ごと書いてもらう形式も考えたが、タブの入れ方やスペースの入れ方が人それぞれであることを踏まえ、私はその内容が正解か不正解か判定するプログラムの作成が難しいと判断し断念した。それにテストのように丸覚えすることが重要ではなく、この関数や変数はどういう時に使用し、どういう流れでプログラムが動いているか学生が理解できればいいのである。後期の授業で作成するものは、新規作成で登録し、一覧表示して編集や削除を行うといったようにほぼ流れは同じであり、関数・変数の使い方や流れの理解が重要となる。

主な形式を関数や変数の穴埋め問題とすればタブや半角スペースを気にする必要がなく正誤判定がしやすくなり、ポイントとなる箇所を空白にして出題することで、私は学生に重要な関数とプログラムの流れを覚えてもらおうと考えた。学生は答えが分からなければ課題が完了しないので、自然と栗津さんの教材や自分のファイルを確認する作業に繋がれるだろう。そうして何度も見ることで、学生が忘れてしまうことを少しでも防げるのではないかと考えている。

(2) ブラウザの指定

本教材ではデザインや動作の確認を行うブラウザを『Firefox』とする。

理由は大学のパソコン（Macintosh、Windows 共に）と自宅のパソコンにインストール済みで共通するものが当ブラウザであったからである。Macintoshにもう1つインストールされていた『Safari』にしなかったのは、JPCERT コーディネーションセンター⁽⁴⁾と独立行政法人情報処理推進機構（IPA）⁽⁵⁾が共同で運営している『Japan Vulnerability Notes』の記事で同年10月23日の時点で脆弱性が報告されているにも関わらず、Windows用の最新版の投稿日が「2012年5月9日」で止まっていることによる。

(3) 参照した資料

本教材を作成するには授業で得た知識だけでは足りず、授業では行わないようなPHPに触れるためにも、福田先生と相談し『Webサイト制作者のためのPHP入門講座』で勉強することにした。

この資料では初めにPHPやWebアプリケーションとは何かの説明がされ、続いてXAMPPのインストール方法、PHPの基礎知識、残りの章でアプリケーションの作成といった流れで書かれている。アプリケーションは1から自分で作るのではなく、指定されたURLからサンプルプログラムをダウンロードし資料に沿ってPHP部分の穴埋めをしていく方式を取っていたため、HTMLはある程度理解しておりPHPだけを勉強したいという私の目的に合っていた。

実際にこの資料を使ってみた感想としては、「入門講座」とあるがアプリケーションの作成はあまり初心者向けとは言えないのではないだろうかと感じた。資料にはきちんと「なぜこうなるのか」と説明されて

いたが、1つのアプリケーションで行う処理が多く複雑で（アンケートフォームなら、配列を利用した繰り返しや外部ファイルに結果を保存など）、私はどうしてこの結果になるのか、どういう流れなのかを理解するのに大変苦労した。その分本教材の作成に使いそうな配列の使い方やセッション、クッキーについて学べたので、この資料で勉強して良かったと思う。だが上述の通りこの資料は初心者には難しいと感じ、授業で使用しない関数が多く使われるので授業の復習課題には向かないものと考える。

（4）構成と実装する機能

（i）ページ構成

1. インデックスページ
2. ログイン・ログアウトページ
3. 問題一覧ページ
4. 問題ページ
5. 登録情報の確認・パスワード変更のページ
6. 管理者用ページ

問題ページのタイトルは利用者が一目で内容が分かるように、基本は「作成するもの（機能）」とする。作成するものとは書籍データベース等のことであり、その箇所を栗津さんの教材の見出しと一致させることで利用者にギャップを感じさせないようにする。機能とは新規作成や編集のことである。

（ii）実装する機能

第4節（i）（p.8）の1・2・5はそのままなので、ここではそれ以外について述べる。

3では、問題ページ名とページへのリンクを表示させる。また、第2章第1節(ii)(p.5)で述べたように、利用者が課題ページを全問正解する前に次のページのURLを直接入力してもページ内容が表示できないようにすることで不正を防ぐ。

4では解答欄ごとに正誤判定を行い、正解すれば解答を赤文字で表示し、間違えていれば空欄のままとする。課題で「問目まで」と指定される可能性を考え、問いを完答すればページの移動やログアウトしても正解状況が残るようにする。復習ページでも同様にし、最初から解きなおせるようにページごとにリセットボタンを設置する。

6では第2章第1節(ii)(p.5)で述べたように学生の課題の進行状況を表示し、先生が注意のメールを送りたい学生の欄にチェックを入れれば<textarea>の中にその学生のメールアドレスが表示されるようにする。

(iii) データベースの構造

データベース名は「php_kyozaisite」、使用するテーブルは「member」と「kaito」、「mondai_menu」とする。以下で各テーブルのフィールドやその役割について述べるが、テーブルに必須であるidフィールドは省略する。また、このデータベースは松森有希さんが制作する前期分の復習用教材サイトと共用するので、この教材では使用しないフィールドも省略する。

member テーブルではアカウントの管理を行う。mode フィールドは、先生のレコードを特定できるようにするために用意した。なぜなら管理者用ページへのアクセス許可に、ログインした人が先生であるか学生であるか判別しておく必要があるからである。このフィールドの値は学生なら「NULL」、先生なら「sensei」とする。

- gakuseki_num ... ID
- name ... 名前
- password ... パスワード
- login_time ... 最終ログイン日時
- unix_mondai ... 課題ページ番号
- mode ... 先生であるかの判別

kaito テーブルでは学生ごとに問題ページの正解状況を保存し、kaito_mondai フィールドは解答中の復習ページを変えた時の判別に使用する。復習ページ用と課題ページ用で正解状況の保存先を分けることにより、課題が「 問目まで」と出された場合でも課題ページの正解状況を残した上で自由に復習ページを解くことができる。

- gakuseki_num ... ID (どの学生のデータかの判別)
- kaito_mondai ... 現在解答中の復習ページ番号
- kaito_zumi ... 復習ページでの解答済み問い番号
- kadai_kaito ... 課題ページでの解答済み問い番号

mondai_menu テーブルでは問題ページの管理を行う。問題ページの情報データベースで管理することで、ページ番号やタイトルの変更時にデータベースの情報を更新するだけで済み、問題一覧ページは修正しなくてよくなる。また、このページ番号を利用して復習ページか課題ページか判断する。

- url ... 問題ページのURL (「 .php」の のみ)
- title ... 問題ページのタイトル
- unix_num ... 問題ページの番号

3 本論

(1) 枠組み

まずデザインの枠組みに取りかかった。デザインを考える上で表示に不具合が出ないように利用者のディスプレイの解像度を意識しなければならないが、『StatCounter Global Stats』と『F.Ko-Jiの「一秒後は未来」』によれば「1366 × 768」が1位となっていたのでそれを基準とする。枠組みは wrapper、header、unix_content、footer とし、header と footer においては内容をそれぞれ別ファイルで作成し、require で呼び出す(図1、コード1)。

(2) インデックスページ

インデックスページは誰でもアクセスできるページとし、このサイトは何であるかとログイン方法を説明している(図1)。ここで、『Firefox』以外で表示するとデザインが崩れる可能性があることを表記した。

(3) ログイン機能

(i) ログイン

【login.php】でIDとパスワードを入力してもらい(図2)、それらを【check_login.php】に送信してログイン成功か否かを判断する。最終ログイン日時はこのページで取得し、hidden を使用することでページ上には表示させず送信している。

【check_login.php】の作成にあたり『つくって覚える PHP 入門』を参考にした。ログイン情報はセッション⁽⁶⁾で管理する。まず session_start(); でセッションを作成し、【login.php】から送られてきたデータを受け取った後で htmlspecialchars 関数でIDかパス

ワードに「'」があった場合「'」に変換する(コード2)。次にデータベースに接続した後で member テーブルから先生の ID を取り出し、\$sensei_id に代入しておく(コード3)。次に、入力された ID とパスワードが一致する member テーブルのレコードを呼び出し、mysql_num_rows 関数で呼び出されたレコードの数を数え、1 つならログイン成功とする。なぜなら ID とパスワードが一致するレコードは 1 つしかないためである。

ここで、利用者の識別に使用する ID を\$_SESSION["gakuseki_num"] に代入してセッションで管理する。そうすればログイン中、利用者のレコードの指定に使う ID がこのセッションを呼び出すだけで使用可能となる。また、ID が\$sensei_id と一致すれば\$login に「sensei」と代入し、一致しなければ「OK」と別の値を代入することにより、先生と学生の区別をする。その後で、header でログインしている利用者の名前を表示するために member テーブルから名前のデータを呼び出し、この値もセッション変数 name として管理する。最後に\$login の値もセッション変数 login で管理し、ログイン中であるか確認する時に利用する。ログイン成功なら問題一覧ページに移動し、失敗なら【login.php】に戻る仕組みとなっている。

ログイン後の全てのページでも最初にセッションを開始し、セッション変数 login の値が「OK」でも「sensei」でもなければログインしていない状態であるため自動的に【login.php】に飛ばす。なお、ログイン時間の有効期限は 2 時間とし、ページをリロードすれば有効期限が更新される仕組みにしている。

(ii) ログアウト

ここでもまず最初に `session_start()`; でセッションを開始し、このサイトにおける全てのセッション変数を削除するために空の配列を `$_SESSION` に代入する。続いて、セッション ID を保存しているクッキーの有効期限を現在より 1 時間前としてセッション ID を削除する。最後にセッションに登録されている値を全て破棄することで、ログアウトできる。ログアウト画面の `unix_content` には「ログアウトしました。」とだけ表示する(図 3)。

(4) 問題一覧ページ

ここからはログイン後のページとなる。ログイン後の header では【`unix_header.php`】を呼び出し、サイトタイトルと名前とログアウトのリンクを表示させる。なお、先生がログインした場合は管理者用ページへのリンクも表示させた(図 4、コード 4)。ログインした人が先生か学生であるかの判断方法については第 3 節 (i)(p.11) 参照。

最初は 1 ページ目のリンクのみ表示し、そのページの問いを全て正解すれば一覧に次の課題ページを表示させるようにしていた。だが同じ方式を取っていた松森さんのアンケート結果に「問題ページの数が分からず終わりが見えないので不安」という意見があったので、最初から全ての問題ページのタイトルを表示しておき解き終われば次のページのタイトルをリンク表示させるよう変更した(図 5)。「課題」と赤文字で表示している点については、第 4 章第 2 節 (i)(p.21) 参照。

(5) 問題ページ

まず URL を直接入力して不正にアクセスしようとした時はエラー表示し(図6)、アクセス可能なページであれば問題内容を表示させるようにした(図7)。また、問題ページ上でも復習ページであるのか課題ページであるのか分かるように背景画像を変えている(図8)。

正誤判定は解答欄ごとに行い結果を表示させ、どこか間違っていれば「間違いです。もう一度!」、完答すれば「正解です!」と表示させる(図9)。完答すれば kaito テーブルの、課題ページなら課題用(kadai_kaito)、復習ページなら復習用(kaito_zumi)のフィールドにその問い番号を登録していく。そうすることで、利用者がページ移動やログアウトを行ってもデータベース上のデータによって正解状況が保てるのである。このデータをリセットするには、課題ページならページ内の全ての問いに正解するかページ下部に設置したリセットボタン(図10)を押し、復習ページなら先ほどの2つの方法に加え他の復習ページの問いを解けば可能である。このようにして最初から解きなおしたくなった場合にも対応した。

上述の通り問いを完答すれば正解状況が保存されるが、それまでの解答欄ごとの正解状況は配列に保存しているためその問いの解答ボタン以外でページを移動すれば最初からになってしまう。これも保つには、データベース上で問いごとに正解状況を管理するかセッションで管理しなければならず、時間の都合とシステムの複雑化を考え実装しなかった。

問題を解けばすぐに結果が見られるよう、解答ボタンを押せば JavaScript でその問いの見出しに飛ぶよう設定した。また、多くのページが縦に長いので、ページの上部を表示している状態からページの

下部にあるコンテンツ（リセットボタンや前後のページへのリンク）への移動がスムーズにできるよう、下部に移動する「下へ」と上部に移動する「上へ」という内部リンクを設置している。

「9. 書籍データベースのPHP化4（一覧）」の問1のような項目数が多く正しい順を選択する問題のプルダウンメニューでは、一目でどれが使用済みか分かりづらいので使用済みの項目に disabled 属性を付けマウスで選択できないようにし、使用済みか否か判別しやすくした（図11）。

（6）登録情報の確認・パスワード変更

ヘッダー部分の自分の名前のリンクから、登録情報ページへ飛ぶようにした。ここではIDと名前、パスワード変更ページへ飛ぶボタンを表示させる（図12）。

パスワード変更ページでは、現在のパスワードと新しいパスワードを入力してもらう（図13）。利用者に新しいパスワードを2回入力してもらうことで、登録ミスを防ぐ。現在のパスワードも入力してもらうのは、共有のパソコンでログアウトし忘れた時等に他の人に勝手に変更されてしまうことを防ぐためである。

実際の更新ページでは成功すれば「変更できました。」と表示し、失敗なら、現在のパスワードが間違っている場合・現在のパスワードは合っているが新しいパスワードが空欄の場合・新しいパスワードは空欄でなかったが確認用と値が違った場合と、エラーの内容にあった文章を表示させる（図14）。

(7) 管理者用ページ

管理者用ページは先生のみアクセスでき、万が一学生がこのページのURLを知りアクセスしてきても問題一覧ページに飛ばすようにする。

ここで学生の利用状況と課題進行状況が確認でき(図15)、課題ができていない学生のデータにチェックを入れて送信ボタンを押せば、移動先でその学生のデータと<textarea>にメールアドレスを表示させる(図16)。全て選択ボタンを押せば全てのデータにチェックが入り、リセットボタンを押せば全てのチェックが消える。

送信した後も再度学生のデータを表示することにより見間違いがないか確認でき、<textarea>に `onfocus='this.select();'` と記述することでクリックすればメールアドレスを全文選択できるようにした(図17)。また、先生がメールのアドレス欄にそのまま貼り付けられるよう「アドレス1, アドレス2, ...」と表示させる。チェックがない状態で送信されていれば、戻るボタンのみ表示させる。

4 結論

(1) アンケートによる評価

教材が8割方完成したところでアンケートの協力をお願いした。対象者は後期で福田先生のプログラミングの授業を受講している福田ゼミ3回生の4人と、この授業と一緒にアシスタントとして入っている福田ゼミ4回生の2人である。

1. 操作しやすかったか
2. 問題の難易度は適切だったか
3. レイアウト、色合いなどデザイン面での見やすさ

4. 良かった点

5. 悪かった点

1~3の質問は5段階評価(数字が大きいほど良い)と何かあれば書いてもらう欄を作成し、4~5は何でも良いので具体的な意見や感想を大き目の欄に書いてもらうことにした。

(i) 結果

1の質問では、「5」が2名で残りの4名が全て「4」とまあまあ良い評価だった。

- ・説明書がいらなくらい分かりやすかった。
- ・問いの全ての欄に正解していれば、ページを移動してもその問いは正解済みで残ったままだったので作業を途中で止めても再開しやすかった。
- ・一番下に次の課題のリンクがあってもいいと思う。

2の質問では、「5」が1名、「4」が4名、「3」が1名とまあまあ良い評価だった。

- ・丁度良いと思う。
- ・全部じゃなくていいが、間違った時にヒントが出ると良い。
- ・自分が覚えていないだけだった。
- ・やや高い難易度であっても、一箇所が合っていればそこは表示が変わるので解きやすく感じた。

3の質問では、「5」が4名、「4」が1名、「3」が1名と良い評価だった。

- ・可愛かった。
- ・統一感があって良かった。プルダウンメニューが選びにくいので文字の大きさが一回り程大きい方が良い。

- ・プルダウンメニューの margin を指定すれば尚良い。

4の良かった点についてのコメントは、この質問内だけでなく、上記の1~3の質問でのコメントと似ているものも省略している。

- ・一度やり終えたページは画像で分かるようになっているので、うっかり二度手間になることがなさそうなので良い。
- ・すぐに答えを出すのではなく、自分で調べたりしながらできるので身に付きやすいと思った。
- ・色をあまり多用せず、強調している箇所がはっきりと分かった。
- ・課題を完了しないと次の課題が出ないシステムは良かった。
- ・プルダウンメニューで既に使用済みのものとそうでないものが分かりやすかった。
- ・正解済みの問いのデータが保存されること。
- ・入力する問題、選択する問題と解くところによって変わるところ。

5の悪かった点についてのコメントは、4の箇所と同様に類似しているものは省略し、一部、4に書かれていたが改善点に繋がるコメントもこちらに列挙する。

- ・トップページから課題進行状況が一目で分かればもっと良くなるかと思う。
- ・一覧ページでも解答済みのページには「済」と表示されるとより分かりやすくなると思う。
- ・一覧ページで一番新しいものが課題で、それ以外が復習になっていることがすぐに分かるようにした方が良いと思った。
- ・同じページで解答途中で別の問いを解くと、その前に解いていた問いの正誤状況がリセットされてしまうのは少しやりにくく感じた。
- ・同じページなら一斉解答できるようになっていた方がまとめて答え

ることができ、すっきりしそう。

- ・「解答」と「リセット」ボタンが正しく表示されない時があった。
- ・「完」と「まだ」の画像が最初から「完」になっているページがあった。
- ・プルダウンメニュー(使用済みとそうでないものが分かる箇所)で、選択済みのところは黒くするか、無くすかすればやりやすい気がする。
- ・一覧ページの箇条書きの番号が、1ケタと2ケタで位置が揃っていないことが気になった。
- ・もう少しヒントがあれば良かった。隠し ON・OFF のような機能があれば良い。
- ・echo でないと正解できない。(print は不可)
- ・「16. 書籍データベースの PHP 化 まとめ」のように、順番通りでないと正解にならないところがある。

(ii) 分析

1の操作性についての質問では、本論(p.11)で述べたように利用者に直接関係するところでも様々な機能をつけていたので複雑になっていないか少々心配であったが、分かりやすいとの結果で安心した。問題ページへの移動が一覧からしかできない現状に不便さが残っていることが判明したため、解決策を考えていきたい。

2の難易度の適切さについての質問では、問題の難易度は適切であり、欄ごとに正誤結果が分かる機能が評価されていることが分かった。問題の作成に当たり、私はアシスタントとして授業中に学生に教えながら、実際学生はどういう箇所が理解しづらいと思っているのか、忘れやすいのかも見て回っていたのでそれが功を奏したのだと感じた。

3のデザイン面での見やすさについての質問では、利用者にも見やす
いと感じてもらえたことが分かった。見やすさは個人によって差がある
が、文章のレイアウトや色合いについての修正点は無しと考える。しか
しプルダウンメニューにおいては選びにくいとの意見があった。確かに
1つ1つの項目の高さは文字の大きさとほぼ同じなため、現状では項目
の高さはあまりなく少しマウスのカーソルを動かすだけで次の項目を
選択してしまう。更に<select>にmarginを指定していないためプル
ダウンメニュー同士が引っ付いており、メニュー自体もクリックしづら
くなっていた。利用者がスムーズに解答できるようにするためにもこの
点は改善しなければならない。

4の良かった点についての質問では、利用者が本教材を使用するに
あたってできるだけ不便さを感じないように工夫した点が評価されて
いた。

5の悪かった点についての質問では様々な箇所が指摘された。しか
し、ボタンが表示されない時があるとの指摘と最初から「完」の画像に
なっているページがあったとの指摘は、前者はPHPの記述ミスであり、
後者はアンケート時のこちらの説明不足だと考える。ボタンの表示につ
いては修正を行い、画像についても念のため確認を行うようにする。そ
の2つを除き残った意見を大きく分けると、課題の進行状況を一覧ペー
ジ上でより分かりやすくすることと、解答方法、機能の不足の3点であ
る。機能の不足については言い換えればヒントが足りないということ
であり、2の質問でのコメントにもヒントについて触れられているので、
ヒントの量を増やした方が良いと判断した。進行状況の表示と解答方法
の変更・修正についても解決策を考えていきたい。

(2) 修正した点としなかった点

修正した点と修正しなかった点を、それぞれ第1節(i)(p.17)のコメント順に述べていく。ただし修正内容や修正しなかった理由が被る場合は、後のコメントも先のコメントと同じ箇所で述べることとする。また、修正しなかった点では説明の都合上一部順が前後している箇所がある。

(i) 修正した点

まず、もともとページ下部にあった一覧ページに戻るリンクを消し次の課題ページのリンクを作成し、それだけでなく前のページにも移動できるリンクも作成した(図18)。「1. データベースの作成」は前にページが無いので、ページ番号(`$unix_num`)が1ならリンク表示せず打消し線を引くようにし、そうでなければページ番号から1を引いた数字と一致するページ番号のURLを呼び出すようにしている(コード5)。条件文を使っているのは、後で先生が前にページを追加する場合を考えてのことである。それ以外のページでは、条件文を使用せずelseの中身のみとなる。次のページへのリンクはまず課題ページ番号(`$unix_mondai`)を呼び出し、それが現在表示しているページの番号より大きければ次のページに進めるので、表示しているページ番号に1を足した数字と一致するページ番号のURLを呼び出す(コード6)。表示しているページが課題ページであれば、リンク表示せず打消し線を引くようにしている。

次に、プルダウンメニューの項目が選びやすくなるようpaddingとmarginで余白を増やし、マウスで選択しやすいようにした(図11)。

次に、利用者が一覧ページ上でどれが課題ページであるかより分かりやすくするために、ページ名の横に「課題」と赤色の文字で表示するよ

う修正した(図19)。また、一覧ページ上ではないが登録情報のページに課題ページ名と解答済みの問い番号の表示を追加した(図20)。これにより、利用者がいちいち課題ページを上から下まで確認しなくてもどの問いが解答済みか一目で分かるようにしている。もし全ての課題が終われば課題ページ名が表示される所には「完!お疲れ様でした。」、解答済みの問い番号がなければその箇所に「なし」と表示させる(図21)。

次に、一覧ページでの箇条書きの番号を桁が変わっても揃うよう変更した。今まではで箇条書きしていたが、<table>を使い<th>にページ番号、<td>に問題ページ名を入れ、<th>にCSSでtext-align: right;と設定し右寄せ表示することでページ番号の表示を揃えている(図22)。

最後に、一部で解答を複数用意するよう変更した。PHPの外の<form>で変数を使用する問いの解答欄では授業でechoを使用していたので正解はそれのみとしていたが、printも使用でき、そちらで作業している学生がいる可能性を考え修正することに決めた。方法は、その解答欄の模範解答はechoとしておき、正誤判定した直後にifで「もしその解答欄に入力されたものがprintなら」その解答欄を正解とするようにした(コード7)。正解の場合「echo(printでも可)」と表示させる(図23)。

(ii) 修正しなかった点

まず、間違った時にヒントを出す機能、並びにヒント表示をボタンで切り替える機能は実装しなかった。理由は問題ページ数が多いゆえに修正に手間がかかり、時間が足りないと判断したからである。その代わりに最初から表示するヒントの量を増やし、自分で確認してもらえるようにヒントとなるページのリンク(「こちら」)や粟津さんの教材に記され

ている箇所を示した(図 24、25)。なぜその解答なのか理解してもらえよう、一部の解答欄で正解すれば補足表示がされるようにもしている(図 26)。

次に、問 A を解答中に問 B も解答できるようにする機能も実装しなかった。解答欄ごとの正解状況を記録する配列の\$hantei と、その問いの解答中に完答するまで「間違いです」と表示するための\$NG をセッションで管理することで他の問いを解いても正誤状況が残るようにはできたが、セッションを使ったことにより、全問正解する前に他のページに移動すればそのページ上でも正誤結果が表示されてしまったのである。それを解決するには他の問題ページを表示した時点で上記のセッション 2 つを初期化すれば可能だが、そうしてしまうと、そのページを解く前に先ほどのページの続きを解きたいとなった場合に正誤状況のデータは初期化され先ほどのページは 1 から解きなおしとなる。これでは操作において制約ができ、現状でも復習は可能であるため実装は諦めたのである。

次に、同じページなら一斉解答できるようにする機能も実装しなかった。一斉解答する場合と問いごとに解答する場合を考えねばならず、一斉解答してから間違っただ箇所を個別に解答する場合は問いごとに正誤状況をセッションで管理する方法しか思いつかなかったのである。セッションを使えば上述(2 つ目の点)と同じ状況になるので実装は諦めた。

次に、プルダウンメニューでの選択済みの箇所の表示も実装しなかった。現状でも、どれが選択済みか分からなくなってもそこで一旦解答することで、正解した項目は選択できなくなり選択肢が減らせるからである。

次に、「16. 書籍データベースの PHP 化 まとめ」で順番どおりでな

いと正解にならない箇所はそのままとした。理由は、問題文に「同じ列にある選択肢は、作成した順に選択してください。」と注意書きしているからである(図 27)。授業ではまず【henshu.php】から作成し、その後で【delete.php】を作成するので問題文どおりに選択すれば正解できると判断した。

最後に、ボタンが表示されない時があったことと最初から「完」の画像になっていたとの指摘だが、ボタンについては作業中の修正ミスで if の条件が一部足りていなかったのため修正し、正常に表示されることを確認した。画像については全てのページを確認したが、正常に表示されていたため私の説明不足が原因だろうと思われる。

(3) 自己評価と今後の課題

アンケートの結果にも「自分で調べながらできるので身に付きやすい」との意見があったことから、本教材を使用することで学生に PHP や MySQL をより理解してもらうことと、栗津さんの教材や自分のファイルを確認してもらうという目的を達成できたのではないかと考える。また第 2 章第 1 節 (i) (p.5) にて挙げた紙の教材の場合の問題点も、本教材の完成と解答欄ごとの正誤判定の実装により解決できたものと考ええる。

問題ページでは、利用者は必ず問 1 から解くとは限らないのでどの問いからでも解けるようにしたり、自分が受講者側だった時の経験等から分かりづらかった箇所についてヒントを出したりした。そうしたからこそアンケートでは良い評価を得られたのだと考える。

しかし、正誤判定の実装に時間を費やしすぎてしまったため実装できない機能があった。『Unix 演習 1・2』は毎年少しずつ細かいとこ

るが変わっているので問題内容の変更、またはページの追加が考えられる。変更は PHP ファイルを修正すれば可能だが、ページの追加は `mondai_menu` テーブルの問題ページの番号 (`unix_num`) を振りなおす必要があり現状ではそれが自動ではできない。また、アンケートの結果にあったようにページ内で一斉解答できず問 A の解答途中に問 B の解答には移れないので少々不便さが残っている。もしこの先後輩が卒業制作で本教材を改良することになれば、そのような機能をつけるとより良いものになるだろう。

このように課題はあるが、問題も授業内容の最初から最後まで全て作成したので最後の授業まで復習に使い、万が一を考え学生の不正を防ぐ措置も取っているので課題としては十分に使えるであろうと考える。

5 おわりに

本教材を作成するにあたり、私は常に利用者の視点に立ち作業を行った。そうしたからこそ苦労したこともある。このような機能があれば便利だろうと思いついても知識が追いついていなかったために作成に時間がかかり、スケジュールの調整の大切さや取舍選択の必要さを痛感した。また、いくら利用者の視点に立っても PHP と MySQL に触れた期間は受講者と比べて 1 年多いためアンケートを取るまで問題の難易度の設定に不安があり、今まで教材を利用する側だったが教材を制作する側の大変さを学んだ。

本教材を繰り返し利用してもらうことで、少しでも授業がスムーズに進み、受講する学生の PHP と MySQL に対する理解が深まることを願う。

注

- (1) HTML ファイル内にスクリプトを記述し、処理結果に応じて文書を返してくるプログラミング言語。
- (2) オープンソースのリレーショナルデータベース管理システム。
- (3) <http://tibat.que.ne.jp/otani/lecture2011/2011ProgrammingUnixClass.epub>
- (4) 民間の非営利団体。日本国内のサイトでインターネットを介して起きる侵入やサービス妨害等の報告を受け付け、対応の支援や手口の分析、再発防止のための対策の検討と助言などを技術的な立場から行っている。(<https://www.jpccert.or.jp/>)
- (5) 経済産業省所管の独立行政法人。日本の情報セキュリティ対策と、情報処理システムの信頼性向上、IT 人材の発掘・育成・排出の促進を担う政策実施機関である。
(<http://www.ipa.go.jp/security/index.html>)
- (6) セッションは保存した値をサーバー側で管理し、ログアウトするまでその値は保持される。クライアント側には、どのセッションを使っているかを識別するためのセッション ID のみクッキーで保存される。

文献表

栗津明里

2011 『PHP 教材の電子書籍化について』

鈴木憲治、安藤建一、小出範明

2011 『Web サイト制作者のための PHP 入門講座』 技術評論社

オフィス加減

2010 『つくって覚える PHP 入門』 アスキー・メディアワークス
JPCERT/CC、独立行政法人情報処理推進機構

『Japan Vulnerability Notes』

<http://jvn.jp/jp/JVN42676559/index.html>

(更新日：2012年10月23日、最終アクセス日：2014年1月4日)

StatCounter

『StatCounter Global Stats』

<http://gs.statcounter.com/#resolution-JP-monthly-201301-201401-bar>

(最終アクセス日：2014年1月6日)

F.Ko-Ji

『F.Ko-Ji の「一秒後は未来」』

<http://blog.fkoji.com/2013/01091306.html>

(更新日：2013年01月09日、最終アクセス日：2014年1月6日)