

Web 技術を活用したインタラクティブな情報教育環境の構築と実践

三浦 元喜[†]

[†]九州工業大学 〒804-8550 福岡県北九州市戸畑区仙水町 1-1

E-mail: †miuramo@mns.kyutech.ac.jp

あらまし Web ベースの学習環境は、ネットワーク接続とブラウザがあればどこでも学習が継続でき、また学習活動に最適化されたインタフェースが利用できるため効果的である。しかし、Web ベースの学習環境は既存の学習者用 PC における操作環境と異なる場合も多い。学習環境の使いやすさや有用性は、学習者の満足度に影響する重要な要因である。我々は、学習者が内容に集中して取り組めるようにするため、一貫性を保ちシンプルでわかりやすく、かつインタラクティブ性の高い Web ベースの学習環境を構築してきた。本報告では、我々が Processing プログラミングと、Web アプリケーション開発を対象とした授業で運用してきた Web ベースの学習システムを実践事例とともに紹介し、それらの運用から得られた利点と欠点に関する知見と、今後の方向性について述べる。

キーワード 学習管理システム (LMS), コース管理システム (CMS), Processing, Web インタフェース

Development and Practices of Web-based Interactive Computer Science Learning Systems

Motoki MIURA[†]

[†] Faculty of Engineering, Kyushu Institute of Technology
1-1, Sensui, Tobata-ku, Kitakyushu-shi, Fukuoka, 804-8550 Japan

E-mail: †miuramo@mns.kyutech.ac.jp

Abstract Web-based learning environment can provide sophisticated, task-specific interface for students. The students can utilize the interface at anytime and anywhere when a network and a web-browser are available. However, interfaces of web-based learning environment often differ from conventional desktop environment. The usability and effectiveness influence the learners' satisfaction. We have been developed a sophisticated, interactive web-based learning environment that helps Processing programming and web application building. In this report, we introduce our interfaces, and explain typical functions and practices of the interfaces. Based on the practices, we describe the merits/demerits and the future directions of the web-based learning environment.

Key words Learning Management System (LMS), Course Management System (CMS), Processing, Web Interface

1. はじめに

Web 技術の進展により、Web ベースの仮想学習環境 [1] や、コース管理システム [2] の研究と開発が行われてきた。とくにプログラミングについては、C 言語のソースコードをステップ実行して視覚的に表示する環境 [3] や、Javascript をベースとした独自言語を利用して開発する環境 [4] など、いくつか提案されている。近年では、HTML5 Canvas や WebGL など、Web ブラウザにおける視覚的でインタラクティブな表現の種類も多様化 [5] しており、今後さらに Web ベースの学習環境が一般的になると考えられる。

Web ベースの学習環境は、ネットワーク接続とブラウザさえ

あればどこでも学習が継続できる点や、特定の学習活動に適した設定があらかじめ整えられた環境を利用できる点において、学習者の負担を下げることができるため、効果的である [6]。しかし、Web ベースの学習環境はその特殊性により、通常の学習者用計算機における操作環境と異なる場合も多い。E ラーニングにおいてどのような要因が学習者の満足度に影響を及ぼすかについて、Sun らが調査した結果 [7] でも、システムの使いやすさや有用性は、学習者の満足度に影響する主要な要因の 1 つであると述べられている。

我々は、学習者が学習の内容に集中して取り組めるようにするため、シンプルでわかりやすく、かつインタラクティブ性の高い Web ベースの学習環境を構築し、学習者に提供してきた。

番号	題名	満点	状況	点差	入力文字数	操作	詳細
1.	絵を書こう	0.5	完了	0.5	347	見る 修正	
2.	前回の復習ミニテスト1	1	完了	1	407	見る 修正	
3.	練習用砂場:translate,rotate	0	回答中	--	207	回答修正 見る	
4.	練習用砂場:(ターゲツグラフィックス)	0	回答中	--	639	回答修正 見る	
5.	練習用砂場:mouseMoved	0	回答中	--	452	回答修正 見る	
6.	練習用砂場:keyPressed	0	回答中	--	154	回答修正 見る	
7.	練習用砂場:draw	0	回答中	--	237	回答修正 見る	
8.	練習用砂場:for	0	回答中	--	419	回答修正 見る	
9.	関数を使って絵を書こう	1	完了	1	38	見る 修正	
10.	前回の復習ミニテスト2	1	完了	1	53	見る 修正	
11.	短いプログラムで複雑な絵を書こう	1	完了	1	38	見る 修正	
12.	練習用砂場:for(その2)	0	未入力	--	0	回答	
13.	練習用砂場:if	0	回答中	--	262	回答修正 見る	
14.	練習用砂場:translate,rotate	0	未入力	--	0	回答	
15.	前回の復習ミニテスト3	1	完了	1	45	見る 修正	
16.	練習用砂場:移動・回転・pushMatrix	0	回答中	--	334	回答修正 見る	
17.	(練習用砂場) 往復運動と当たり判定	0	回答中	--	436	回答修正 見る	
18.	(練習用砂場) 放物運動	0	回答中	--	284	回答修正 見る	
19.	(練習用砂場) 配列	0	回答中	--	339	回答修正 見る	
20.	運動・当たり判定	1	回答中	--	150	回答修正 見る	2016-05-20 (金) 23:59 10 時間
21.	線分切り出し、すぐに相互判定モジュールに移ります 時計を揃えよう(移動と回転、開数、変数の復習) 線分切り出し、すぐに相互判定モジュールに移ります	1	完了	1	205	見る 修正 相互判定	
22.	(練習用砂場) 1次元配列	0	回答中	--	812	回答修正 見る	
23.	前回の復習ミニテスト4	1	完了	1	37	見る 修正	
24.	棒グラフ(or 円グラフ)	1	回答中	--	692	回答修正 見る	2016-05-27 (金) 23:59 7 日
合計点数		9.5	点差	7.5	点	(78.9%)	

図 1 演習システム (OurLMS) : 課題一覧

Fig. 1 List of exercises in OurLMS

本報告では、我々が Processing プログラミングと、Web アプリケーション開発を対象とした授業で運用してきた Web ベースの学習システムを実践事例とともに紹介し、それらの運用から得られた利点と欠点に関する知見と、今後の方向性について述べる。

2. 事例 1 : Processing.js と Web エディタを組み込んだ演習管理システム OurLMS

OurLMS は、我々が 2009 年から開発し、実際の講義で利用している演習管理システムである。学習者はログインすると、図 1 のような課題一覧が表示され、これまでの演習課題とスコア、今後の演習や締切を、一画面で統合的に確認することができる。当初は出席管理のほか、レポートファイルやプログラムソースコードを提出する機能しか備えていなかった。2012 年の 10 月に、Processing.js と Web エディタによる Web IDE を組み込み、講義の演習で継続的に用いてきた。

Processing.js [8] は、John Resig らによって開発された、Processing のプログラムを Javascript に変換し、HTML5 の Canvas 要素上で動作させることができる Javascript ライブラリである。現在では多くの Web ブラウザが HTML5 Canvas に対応しているため、Web ブラウザさえあれば視覚的かつインタラクティブなプログラムを動作させることが可能である。

Processing.js と Web エディタの組み込みにあたり、我々

```

1 int a0,a1,a2,a3,a4; //配列を使わない場合
2
3 void setup(){
4   size(200,200);
5   textSize(20);
6   textAlign(CENTER); //文字列のセンタリング
7 }
8
9 void draw(){
10  background(200);
11
12  for(int h=0;h<5;h++){
13    if(!is_inside(h)){ fill(255,255,0); } else { fill(255); }
14    rect(h*40, 100, 38,38);
15  }
16  fill(10);
17  text(a0, 20, 129); //配列を使わない場合
18  text(a1, 20+40, 129);
19  text(a2, 20+40*2, 129);
20  text(a3, 20+40*3, 129);
21  text(a4, 20+40*4, 129);
22 }
23
24 int is_inside(int x){
25  if ( mouseY < 100 || 138 < mouseY) return 0;
26  if (int(mouseX/40) == x) return 1;
27  else return 0;
28 }
29
30 void mousePressed(){
31  int pos = int(mouseX/40);
32  if (is_inside(pos)){
33    if (pos==0) a0++; //配列を使わない場合
34    else if (pos==1) a1++;
35    else if (pos==2) a2++;
36    else if (pos==3) a3++;
37    else if (pos==4) a4++;
38  }
39 }
40

```

図 2 演習システム (OurLMS) : for ループと配列の練習課題

Fig. 2 Exercise of “for” loop and array in OurLMS

は、Yutaka Kachi 氏が提供しているファイル^(注1)を参考に、Javascript で記述された CodeMirror エディタ [9] を組み込んだ。これに、フォントサイズ変更、ソースコード全選択、全角記号チェック、エラーチェック機能を追加することによって、Web ブラウザのみでプログラム編集と動作確認が行える環境を構築した [6]。エラーチェックについては、サーバ側で Processing (スタンドアロン版) を起動して、ソースコードをコンパイルしたときのエラーを取得して、学習者にフィードバックしている。また、インデント自動調整ボタンにより、学習者は自分が書いたソースコードを整形したり、ブラケットの対応関係を確認したりすることができる。

2.1 演習例

学習者は OurLMS にログインし、課題一覧 (図 1) の回答 (修正) ボタンを押すと、図 2 のような編集実行画面が表示される。画面右側に表示されたソースコードを修正して、CTRL+S または保存ボタンを押すと、サーバにプログラムを保存したあと、画面左側の領域でプログラムが動作する。

学習者は、教員が準備したサンプルプログラムを少しずつ修正し、動作確認しながら、条件分岐や繰り返し、配列といったプログラミングの基礎を学ぶことができる。たとえば、図 2 で示したプログラム (箱をクリックすると、対応する数字が増えていく) を修正していくことにより、配列を使った場合には mousePressed() 関数のなかの if 文が省略できることや、draw() 関数のなかで、箱のなかに数字を表示する text() 関数呼出しを for 文でまとめられることがわかるため、配列を使うことの利点や意義を理解できる。

(注1) : <http://www.catch.jp/program/processing.js/edit/>

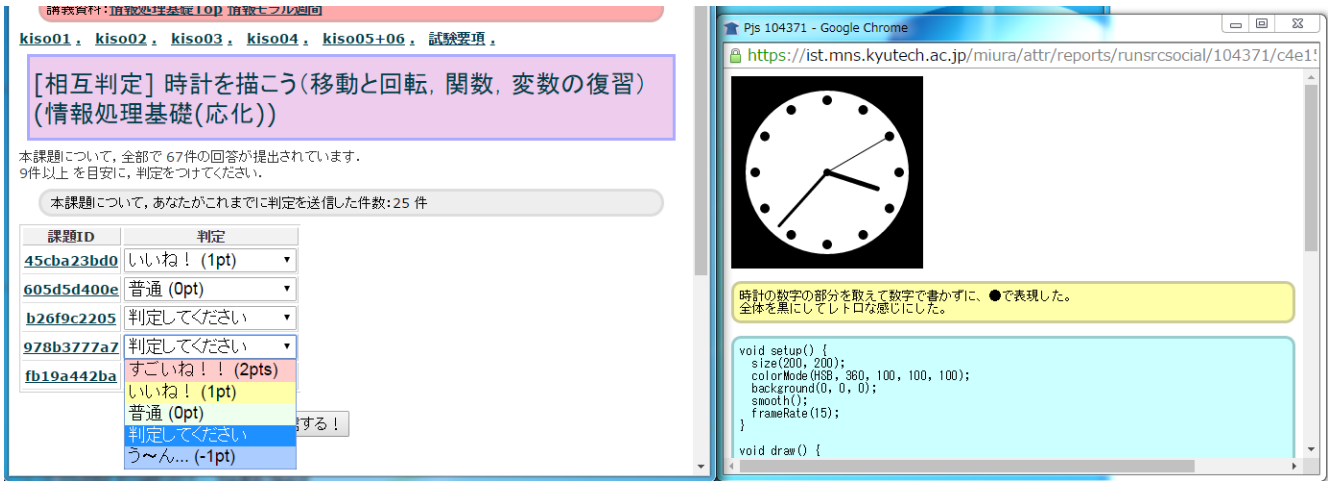


図 3 提出課題の相互判定

Fig.3 Social check of exercises

2.2 グループ課題

OurLMS では、グループによる演習や課題提出の機能も備えている。教員がグループ課題として作成した演習については、ソースコードや記述内容がグループメンバーで共有される。プログラミングをグループで行うことによって、Pair Programming [10] に準じた学習効果が期待できる。

2.3 相互評価・判定機能

演習課題について、他の学習者が提出したプログラムを動作させて、その評価点を入力するといった活動も行える。図 1 の課題一覧で、相互判定ボタンを押すと、図 3 の左側に示すようなウィンドウが画面に現れ、課題 ID のリンクが表示される。学習者が課題 ID のリンクをクリックすると、図 3 右側のウィンドウが表示され、他の学習者が提出した Processing プログラムが動作する。プログラムの下にはコメントとソースコードも表示される。学習者はマウスやキーボード操作によって実際の動作を試したり、またコメントやソースコードも考慮して、判定を入力する。相互判定で表示される課題は、自分以外が提出した課題のうち、評価総数が少ないものから優先的に、ランダムで選択される。このような相互評価・判定機能を導入することにより、学生が他者に評価されることを意識しながら演習することを促す。また他者のプログラムを正しく評価しようとする姿勢による学習効果も期待できる。なお判定結果は、教師が講義で学生に模範回答例を紹介するときや、提出物の個別評価を行う際にも役立つ。

2.4 統一的な管理と、シンプルで一貫性のあるインターフェース

OurLMS では、学習者が一旦ログインすれば、講義資料の Web ページや、講義で説明に用いたスライドに簡単にアクセスできるようにしている。スライド自体も Web ページとして作成しているため、学習者は PC だけでなくスマートフォンからでも復習ができる。また、出席や、小テスト、演習、レポート、試験、アンケート等、講義に関する情報をすべて統一的に管理しているため、シンプルで一貫性のある操作インターフェースを提供できる。

3. 事例 2: Web 開発オンラインエディタ Sweetie

Sweetie (Simple Web Editor Utilities) [11] は、Web サーバ上の特定のフォルダ以下のファイルを、Web ブラウザから直接編集することができるエディタ兼ファイルマネージャである。エディタ部分は、前述の演習システム (OurLMS) と同じ CodeMirror [9] を利用している。シンタックスハイライト機能やインデント機能、CTRL+S による保存や、CTRL+Z による Undo 等のショートカットキーも有効であるため、通常のエディタと同様の操作感で編集・保存できる。また図 4 に示すように、HTML の文法エラーを表示するため、初心者が犯しがちなタグの対応付けのミスについて、早い段階で気づかせることができる。

さらに Sweetie では、画像ファイルをエディタ内にドラッグ & ドロップしたり、クリップボード画像を CTRL+V でペーストすると、画像データをファイルとしてサーバにアップロードしたうえで、インライン表示するための HTML/Markdown 記述をエディタ内に挿入する機能を備えている。これにより、画像を含む Web ページを簡単な操作で記述できる。

3.1 開発の経緯と改良

当初 Sweetie は、学生計算機上で動作し、学生計算機上のローカル HTML/CSS ファイルを編集するための、軽量かつシンプルな Web 開発エディタ環境として設計・実装した [11]。管理者権限のない学生計算機において、シンタックスハイライト機能やインデント機能を備えたエディタを導入し、Web 開発演習を効果的に行うことが主目的であった。

最初に Sweetie を導入した 2015 年度前期の講義では、まず Sweetie とサンプルコードを含む圧縮ファイルを配布して学生計算機上で展開し、ローカルの Web サーバ (QuickPHP [12]) を各学生計算機上で起動したうえで、ローカルファイルを編集してもらった。そして、Web サイト構築がある程度進んだ段階で、教員が設定した公開用の Web サーバにアップロードする機能を使って送信してもらった。

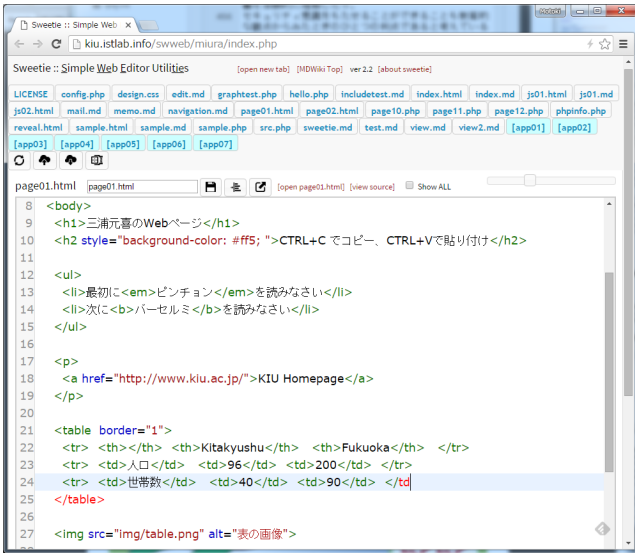


図 4 Web 開発オンラインエディタ Sweetie の画面：文法エラーの強調表示

Fig.4 Screenshot of Sweetie web editor : Highlighting a syntax error

Web サーバの概念や仕組みと、ローカルファイルを公開用 Web サーバにアップロードすることではじめて公開される、ということを経験的に理解してもらおう点では、学生計算機上で Web サーバを起動し、ソースを編集・アップロードするという一連の行為に意義はあった。しかし、作業効率の観点からは、ローカルファイルを編集して、随時アップロードして確認するよりも、直接サーバ上のファイルを Sweetie のエディタで編集したほうが効率的である。

そこで、2015 年度後期の講義では、教員が準備して配布するアカウント名とパスワードを用いて一旦アップロードした後は、ユーザを認証したうえで、すべて Web サーバ上で直接編集・確認作業を行ってもらうようにした [13]。これにより、学生はグループでの演習が行いやすくなり、またファイルの保存場所やバージョンを意識することなく、開発作業に集中できる。また、学生の最新の作業ファイルがすべてサーバ上に置かれていることにより、教員が学生の作業状況を把握しやすくなった。付加的なメリットとして、サンプルコードを含むフォルダを、授業の進捗にあわせて追加することも容易であった。

2016 年度前期からは、Sweetie による学習者主体の開発作業を促進させるため、OurLMS で希望のフォルダ名を入力したら、アカウント名とパスワード（乱数）を自動発行する仕組みを導入した（図 5）。この連携により、学生が OurLMS を用いて Sweetie フォルダ（Sweetie で編集可能な Web サーバ上のフォルダ）をいつでも簡単に追加したり、削除したりすることができるようになった。また個人用とグループ用について、それぞれ複数（現在はそれぞれ 4 つまで）の Sweetie フォルダを作成することができる。グループ用の Sweetie フォルダについては、共通のアカウント名とパスワードが設定・提供されるため、複数人による協調編集が可能である。学習者は、図 5 の画面において、「自動ログイン」ボタンを押すと、別ウィンドウで



図 5 Web 開発オンラインエディタ Sweetie の管理画面（学生）
Fig.5 Administrator page of Sweetie web editor for students

Sweetie フォルダの編集画面が表示される。ちなみに、Sweetie フォルダを設置する Web サーバや共通部分の URL（パス）については、講義毎に設定できる。Sweetie フォルダを置くサーバ側で設定をしておけば、OurLMS をホストしているサーバとは別のサーバに Sweetie フォルダを作成することもできる。

3.2 インタフェース上の改良

以前の Sweetie エディタ画面は、図 4 に示すように、ファイルやフォルダを表すボタンが画面上部に常時表示されていた。ファイルやフォルダが少ないうちは問題ないが、数が増えてくるとエディタ画面上の専有面積が増えてしまうことがあった。また、ファイルやフォルダを移動したり、ファイル名を変更したりする操作が提供されておらず、操作が煩雑であった。

そこで、我々は図 6 に示すように、ファイル選択・管理画面を必要ときに画面左から引き出して表示するように改良した。またファイルやフォルダ表示について、フォルダツリー表示を導入して、一覧性を高めた。これにより、通常の編集画面では図 7 に示すようにファイル一覧を隠し、表示をシンプルにすることができた。

さらにファイルやフォルダ名で右ボタンをクリックすると、メニューを表示するようにした。これにより、ファイルやフォルダのコピーや名前変更、削除をメニュー選択で簡単に実行できるようにした。またファイルを直接 Web ブラウザで表示 (Open in New Window) したり、ファイル編集画面を別タブ / ウィンドウで表示 (Edit in New Window) する機能を追加した。これにより、ローカルファイルを扱うときの操作に近くなったことで、学習者が意図した編集操作やファイル管理操作が行いやすくなった。

なお、Sweetie には、エディタ画面（例：図 4）の [view source] リンクを押すと、編集中のソースコードを表示するためのページ URL（例：図 8）を簡単に生成できる機能がある。この機能を用いると、編集権限がない利用者に対してもソースコードを

閲覧させることができる^(注2)ため、教員が学習者にサンプルプログラムを即座に提示したり、学習者がうまく動かないプログラムをグループで共有したりするときに、有効であった。

3.3 実際の利用事例

2015年度後期のWebアプリケーション開発講義において、HTML/CSS/PHPの導入とJavascriptの簡単な演習、SQLite/phpLiteAdmin[14]を用いたデータベース設計と、PHPからSQLで問い合わせをする方法までを効率的に説明し、アプリケーション構築演習につなげることができた。特にシンタックスハイライト表示や、文法エラーの強調表示(図4)、phpLiteAdminによって、受講生はHTMLやPHP、SQLiteの学習を効率的に行えた。

2016年度前期のWebアプリケーション開発プロジェクト学習において、OurLMSと連携してインタフェースを改良したSweetieを用いて、HTML/PHP/SQLiteの演習・開発を行っている。フォームで写真をアップロードしたら、SQLiteにBLOBとして保存し、一覧表示するアプリケーションも簡単に作成できることを確認した。教員が学生の編集結果を確認する際、HTMLについてはページソース表示で確認できるが、PHPのコードについてはページソース表示では確認できない。このようにときも、前述のソースコード表示・共有機能が役に立った。

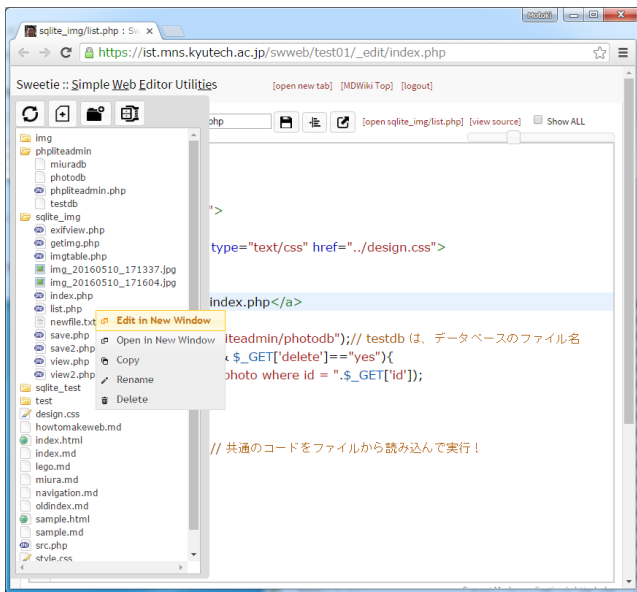


図6 Sweetieにおけるファイル選択・管理画面

Fig.6 File List of Sweetie

4. Webベースの学習環境を導入することによる利点と欠点

これまで述べてきたようなWebベースの学習環境を導入することによる利点として、1章で述べたどこでも学習が継続できる点や、調整された演習環境の提供ができる点に加えて、以下の利点があると考えている。

- (1) 学習者の演習結果や学習成果物を、Webに公開・共有

(注2)：表示に必要なパスキーがURLに含まれている。

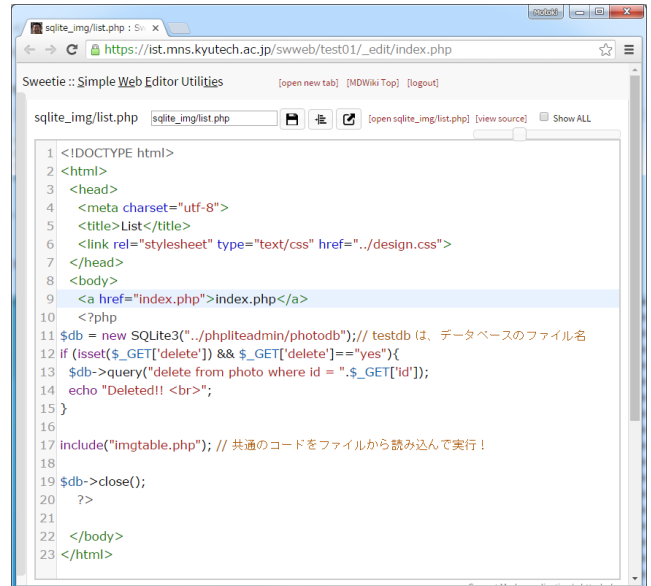


図7 SweetieによるPHPファイルの編集

Fig.7 Edit PHP file by Sweetie



図8 SweetieによるPHPコードの表示

Fig.8 View PHP file by Sweetie

する際のコストが下がることで、Web公開・共有を前提とした学習活動が容易に行えるようになる。

- (2) Web公開・共有を前提とした学習活動に、相互評価機能を取り入れることによって、学習者の意欲や関心を高めることができる。

- (3) 学習者の活動がWeb上で行われるため、活動の詳細を確認することができ、きめ細かな指導や授業改善につなげられる。

元来、学習者は他の学習者の活動に興味があり、参照したり参考にした、良いものは他者に見せたいという意識を持っていると考えられる。学習者の演習結果や学習成果物を、Webとの親和性が高いフォーマットで作成してもらうことで、Web公開・共有を簡単に行えるようになり、学習者が他の学習者の活動を参照できるようになる。それに相互評価の仕組みを含めることによって、Webで共有する効果をさらに高めて、学習効果に結びつけることができる。

また学習者のすべての活動をWebブラウザ上で行わせるように、学習環境をデザインすることができる。これまで我々が構築してきた学習環境ではあまり取り入れていないが、詳細な活動ログに基づいた学習者支援が容易になると考えられる。

逆に Web ベースの学習環境を導入する欠点や注意点として、以下が挙げられる。

(1) ブラウザの種類によって、提供されている機能が異なるため、動作に若干の差異が生じることがある。

(2) ネットワークやサーバに負荷がかかりすぎるような活動はできない。

(3) プログラミング演習において、無限ループを作成してしまうと、ブラウザが固まってしまうことがある。

(1) については、たとえば Sweetie におけるクリップボード画像のペーストによるファイルアップロードが、Safari では実現できないことが挙げられる。動作環境を制限したり制約を置くことによって緩和できる。(2) と (3) については、実装の工夫や事前チェックによって回避できる可能性がある。

5. 関連研究

Hwang らは、プログラミングに関する人間の認知能力を考慮した Web システム WPAS を開発し、評価を行っている [15]。オンラインコーディング機能や、他人のソースコードに対してハイライトによる色付けやコメントを付与するアノテーション機能（相互評価機能）を提供している。また相互評価機能の有用性について評価実験により検証している。我々の研究では、ソースコードのハイライト機能を備えた高機能な編集環境を提供したり、相互評価時にプログラムを自動実行して判定を行いやすくするなど、インタフェースの改善によって学習者の認知的な負荷を下げることによって、学習の効率を高めることをねらいとしている。

桑島ら [16] は、HTML5 を用いた Web ベースのシステムによって実現可能な教育・学習支援について議論している。WebSocket を用いた双方向性の高い支援の種類や、HTML Canvas による手書き入力の利用について述べている。本研究において現時点では AJAX を利用する程度であり、WebSocket のようなリアルタイム性の高い双方向データ通信技術は用いていないが、今後学習者の体験を向上させるために利用する可能性はある。我々は、プログラミングや Web 開発に Javascript や HTML Canvas の機能を利用し、シンプルで統一的な演習管理と一貫性のあるインタフェースを実現した。

6. まとめと今後の課題

Web ベースの学習環境について、我々が構築してきたプログラミングを対象とした環境と、Web アプリケーション開発を対象とした環境について事例を挙げ、それらの実践によって得られた有効性や効果について述べた。また、Web ベースの学習環境を導入することによる利点と欠点について考察した。

今後の課題として、Web 上で活動を行う利点を活かした、詳細な作業ログに基づくプログラミング支援や、補完や検索機能の強化が挙げられる。たとえば大森ら [17] が Eclipse IDE に対して実現したソースコード変更抽出の仕組みや、Example-Centric Programming [18] で提案されているような高度なプログラミング支援を OurLMS に組み込むことで、学習者がより効果的に学習できる環境を構築していきたいと考えている。

謝辞

本研究の一部は公益財団法人電気通信普及財団および JSPS 科研費（課題番号 15K00485）の支援によるものです。

文 献

- [1] G. Piccoli, R. Ahmad, and B. Ives, “Web-Based Virtual Learning Environments: A Research Framework and a Preliminary Assessment of Effectiveness in Basic IT Skills Training,” *MIS Quarterly*, vol.25, no.4, pp.401–426, 2009.
- [2] L. Lee, “Using Web-based instruction to promote active learning: Learners’ perspectives,” *CALICO Journal*, vol.23, no.1, pp.139–156, 2005.
- [3] K. Chansilp and R. Oliver, “Using multimedia to develop students’ programming concepts,” 2003.
- [4] 長島和平, 長 慎也, “Tonyu System 2 ゲーム制作を通じたプログラミング学習に適したフレームワーク,” 情報処理学会研究報告: コンピュータと教育 (CE), vol.2015, no.2, pp.1–8, March 2015. <http://ci.nii.ac.jp/naid/110009882608/>
- [5] H. Levkowitz and C. Kelleher, “Cloud and mobile web-based graphics and visualization,” *Proceedings: 25th SIBGRAP - Conference on Graphics, Patterns and Images Tutorials, SIBGRAP-T 2012*, pp.21–35, 2012.
- [6] 三浦元喜, “Processing Web IDE を用いたプログラミング基礎教育の試み,” 情報教育シンポジウム 2013 論文集, 第 2013 巻, pp.225–231, Aug. 2013.
- [7] P.C. Sun, R.J. Tsai, G. Finger, Y.Y. Chen, and D. Yeh, “What drives a successful e-Learning? An empirical investigation of the critical factors influencing learner satisfaction,” *Computers and Education*, vol.50, no.4, pp.1183–1202, 2008.
- [8] Processing.js Team, “Processing.js” <http://processingjs.org/>.
- [9] M. Haverbeke, “CodeMirror – versatile text editor implemented in JavaScript for the browser,” <https://codemirror.net/>.
- [10] H. Natsu, J. Favela, A.L. Morán, D. Decouchant, and A.M. Martinez-Enriquez, “Distributed pair programming on the Web,” *Proceedings of the Mexican International Conference on Computer Science*, vol.2003-January, no.1, pp.81–88, 2003.
- [11] 三浦元喜, “Sweetie: 学生計算機上で動作する協働作業に適した軽量 Web エディタ環境,” 情報処理学会情報教育シンポジウム (SSS2015), pp.117–123, Aug. 2015. <http://id.nii.ac.jp/1001/00144690/>.
- [12] Z. Saw, “QuickPHP – Lightweight Standalone Bloat-free Portable PHP Web Server,” http://www.zachsaw.com/?pg=quickphp_php_tester_debugger.
- [13] 三浦元喜, “お手軽 Web 開発環境 Sweetie,” 情報処理学会インタラクティブ 2016 シンポジウム, pp.281–284, March 2016.
- [14] “phpLiteAdmin: The Web-based Database Management Tool for SQLite.,” <http://www.phpliteadmin.org/>.
- [15] W.Y. Hwang, C.Y. Wang, G.J. Hwang, Y.M. Huang, and S. Huang, “A web-based programming learning environment to support cognitive development,” *Interacting with Computers*, vol.20, no.6, pp.524–534, 2008. <http://dx.doi.org/10.1016/j.intcom.2008.07.002>
- [16] 桑島聖大, 香川考司, “HTML5 を用いたリアルタイム性のある Web ベースの教育支援システムの開発,” 情報処理学会研究報告, vol.2011-CE-10, no.1, pp.1–7, 2011.
- [17] 大森隆行, 丸山勝久, “開発者による編集操作に基づくソースコード変更抽出,” 情報処理学会論文誌, vol.49, no.7, pp.2349–2359, July 2008.
- [18] J. Brandt, M. Dontcheva, M. Weskamp, S.R. Klemmer, and S. Francisco, “Example-Centric Programming: Integrating Web Search into the Development Environment,” *ACM Conference on Human Factors in Computing Systems*, pp.513–522, 2010.