

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

三浦 元喜^{1,a)}

概要: 本稿では、とくに Web サイト構築演習を対象とした、学生計算機上で動作する協働作業に適した軽量 Web エディタ環境の開発と試用結果について報告する。一般に、プログラミングや Web サイト作成演習においては、シンタックスハイライト機能付きのエディタを使ったほうがソースコードの編集ミスに気づきやすい。しかし、教育用の計算機システムは運用管理やセキュリティ確保の都合上、学生や教員が自由にソフトウェアをインストールしたり設定をしたりできない場合がある。また、グループで作業させたい場合もあるが、Web サイトの作成においては一般に編集作業結果を統合しにくい。このような場面に対処するため、我々は学生計算機上で Web サーバを起動し、Web ブラウザ内でローカルファイルを編集可能な Web アプリケーションを構築した。学生はローカル環境にインストールしたテキストエディタと同等の、シンタックスハイライト機能やインデント機能が備わったテキストエディタを各自の Web ブラウザ内で動作させることができる。Web サーバ上で動作するため、グループによる協調編集作業にも適用可能である。講義において試用した結果、日本語フォルダ名を含む場所では起動できない、漢字変換中の下線が表示されないといった問題点がみつかったが、グループによる協調編集作業ができる、情報共有用の Wiki サイトを併設できるといった利点について一定の評価が得られた。

キーワード: HTML 演習, シンタックスハイライトエディタ, ポータブル Web サーバ

Sweetie: Lightweight Web Editor Environment for Collaborative Web Building Exercise

MIURA MOTOKI^{1,a)}

Abstract: We have developed a lightweight simple web editor suitable for collaborative website building exercises. Generally, a syntax highlighting editor helps the learners to notice mistakes of their editing. However, most of computer systems for learning are prohibited to install/setup softwares freely. This may reduce the usability and efficiency of the exercise. In addition, the gathering of individual works during the website building exercises is difficult for the novice learners. To solve these issues, we have chosen to run local web server on each student computer. The local web server provides an editor function as a web service, as well as the website contents built by the learners. The learners can utilize the editor with useful functions such as syntax highlighting, indentation, and file/image upload functions. Preliminary experiment revealed issues regarding Japanese folder name and Japanese IME. But we confirmed the effectiveness of the collaborative group editing environment.

Keywords: HTML exercise, Syntax highlighting editor, Portable Web server

1. はじめに

プログラミングや Web サイト作成演習では、利用する

¹ 九州工業大学 基礎科学研究系

^{a)} miuramo@mns.kyutech.ac.jp

言語や環境、エディタ、ツール等の要因が学習効果に影響しやすい。とくに初学者においては、情報教育ならびにコンピュータに対する印象や、その後の学習意欲に影響を及ぼすことがある。そのため、初学者向けのプログラミング環境やエディタ、ツールについては様々な研究がなされている。たとえばドリトル [1] では、初学者のつまづきを防ぐための日本語記述を用いた簡潔な構文、編集画面と実行画面の統合が図られている。また Nigari System[2] でも、Java を簡素化した言語や、コンパイル環境と実行環境の統合が図られている。

しかし初学者が Web ページを作成したり、Web サイトを構築したりするための環境については、プログラミング環境に比べるとあまり研究がなされていない。この原因として、なにかしらのエディタがあれば HTML ファイルを記述できることや、HTML ファイルをブラウザで開けば簡単に動作確認できることなどが考えられる。しかし、我々は、プログラミングと同様に、可能であれば初学者に配慮した環境を提供したほうが望ましいと考えている。

近年では、Sublime Text^{*1} や Atom^{*2} といった、プラグインによって柔軟にカスタマイズでき、拡張性に富んだテキストエディタが登場している。また、Brackets^{*3} という Web 開発に最適化した、高機能なエディタも存在する。これらの高機能エディタは、シンタックスハイライト機能やインデント調整機能など、初学者が HTML の文法や構造を理解することを促進する機能を備えており、有用性は高い。

しかし、これらのエディタを導入するにはインストールするための管理者権限が必要であることが多い。一般に教育用の計算機システムは運用管理やセキュリティ確保の都合上、学生や教員が自由にソフトウェアをインストールできない場合があるため、簡単に導入することが難しい場合がある。ソフトウェアがインストールされていても、設定やプラグインの導入など、利用者が個別に行えるカスタマイズが制限され、ソフトウェアの操作性向上に影響する場合もある。

そこで我々は、初学者が Web サイト構築演習を行う際に、管理者権限を必要とせずに学生計算機上で動作させることができる軽量 Web エディタ環境 Sweetie を開発した。

2. 軽量 Web エディタ環境 Sweetie

本章では我々が設計・構築した Web サイト構築演習用の軽量 Web エディタ環境 Sweetie について、設計指針と実装について述べる。

2.1 設計指針

我々は、Web サイト構築演習用の軽量 Web エディタ環

境 Sweetie を構築するにあたり、以下の設計指針を設けた。

- 学生計算機上のローカルファイルを編集することができる。
- シンタックスハイライト機能、インデント機能を備える。
- グループ (2~6 人程度) が協働して編集作業できる。
- 管理者権限がなくても、学生計算機に導入して動作させることができる。
- ファイルの保存や全文選択といった、基本的なショートカットキーを備えている。
- ローカルファイルとして編集・構築した Web サイトを簡単に確認できる。
- ローカルファイルとして編集・構築した Web サイトを簡単にサーバにアップロードできる。

上記の設計指針を実現するため、我々は学生計算機上で Web サーバを起動し、Web ブラウザ内でローカルファイルを編集可能な Web アプリケーションとして構築することにした。

2.2 機能

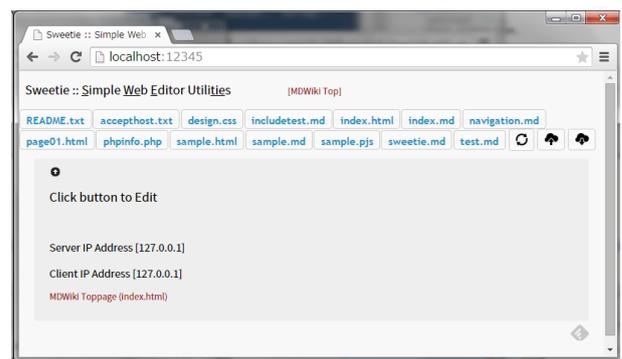


図 1 Sweetie 起動画面

Sweetie は Simple Web Editor Utilities の略称で、シンプルな Web 開発エディタとして実装した。インストール不要のため、フォルダ毎コピーするか、または圧縮ファイルを展開するだけで使用できる。Sweetie は PHP スクリプトで記述された、Web ブラウザ内でローカルファイルを編集可能な Web アプリケーションである。そのため、Sweetie のフォルダをドキュメントルートとした簡易 Web サーバを起動する必要がある。WindowsOS においては、QuickPHP[3] が利用できる。

起動と編集、確認

簡易 Web サーバを起動した後、任意のブラウザで、Web サーバを起動したときに設定したポート番号を指定して開くと、図 1 に示すように、Sweetie の起動画面が表示される。起動画面の上部には編集可能なファイル名が表示されたファイルボタンと、ファイルボタンの再読み込み、Web アップロードとダウンロード用のアイコン付きボタンを表

*1 <http://www.sublimetext.com/>

*2 <https://atom.io/>

*3 <http://brackets.io/>

示している。また画面の中央には、WebサーバとクライアントのIPアドレスを表示するようにした。ファイルボタンには、Sweetieのフォルダ直下にあるファイル群のうち、ファイル名がアンダースコアで始まるものと実行・設定ファイル、バッチファイル類を除いた、編集可能ファイルを表示する。

図1において、上部のファイルボタンをクリックすると、図2に示すように、ファイルを編集するためのエディタが画面中央部に表示される。

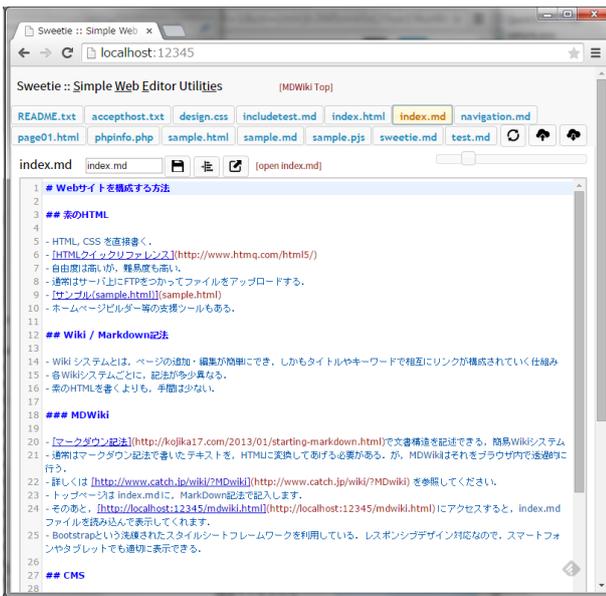


図2 Sweetie エディタ表示画面

このエディタは、HTML/CSS/Javascript/Markdown等のシンタックスハイライト機能に対応している。学習者は、エディタ内の内容を書き換えることで、Sweetieのフォルダ内のローカルファイルを直接修正することができる。また、エディタ画面上部にある「出力を確認」ボタン(図2における、3つ並んだアイコンボタンの一番右)を押すと、現在編集しているファイルの出力結果を別タブに表示できる。

ファイル名変更とファイル削除

現在編集しているファイルを別名で保存したい場合は、エディタ上部のテキストフィールドに表示されている、ファイル名を修正してEnterをおす。すると、図3に示すような確認ダイアログが表示される。ここでOKを押すと、別名保存が実行される。

また、現在編集しているファイルを削除したい場合は、ファイルの内容を空にして、保存ボタンを押す。すると「The content is empty. Delete this file?」というダイアログが表示される(図4)。ここでOKを押すと、ファイルの削除が実行される。または、直接Sweetieのフォルダ内にあるローカルファイルを削除して、ファイルボタンの再読み込みを押してもかまわない。

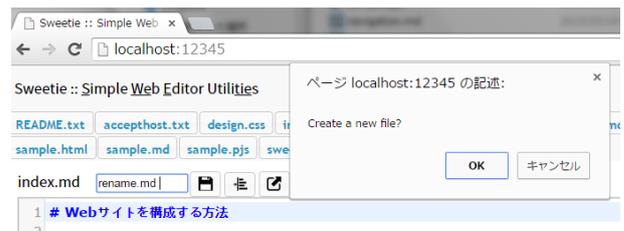


図3 ファイル名を指定して保存(コピー)

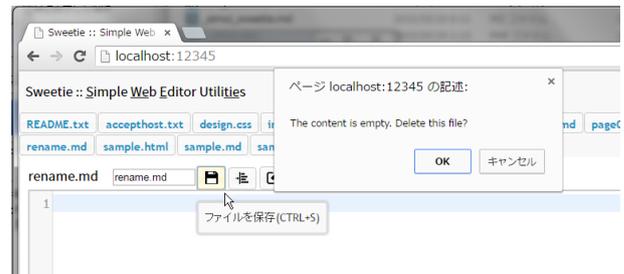


図4 ファイルの削除

インデント調整・フォントサイズ変更

Sweetieでは、図2で示したシンタックスハイライト機能に加え、HTML/CSS/Javascript/Markdown等のインデントを自動的に調整する機能を備える。学習者は、「字下げをそろえる」ボタンを押すだけで、現在編集しているファイルに対してインデント機能を適用できる。一例として、HTMLファイルに対してインデント機能を実行した後の画面を図5に示す。インデント機能により、学習者はHTMLのリスト構造やテーブル構造を記述するときに、ミスがないかどうかを確認することができる。

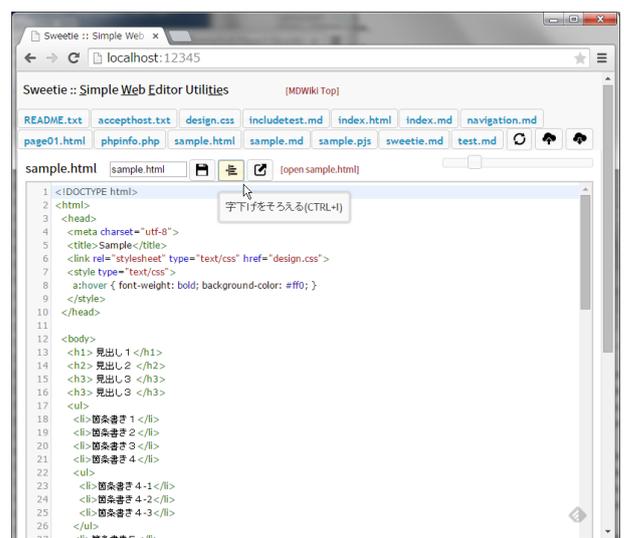


図5 インデント(字下げをそろえる)機能

またSweetieでは、学習者の読みやすさの調整や、教員がプロジェクトや教師画面で編集しているファイルの内容を提示することを鑑み、編集時の文字サイズをスライダによって自由に調整する機能を備えている。一例として、CSSファ

イルに対してスライダによる文字サイズ調整を実行した画面を図 6 に示す。文字サイズは 5 ピクセルから 50 ピクセルまで、0.25 ピクセル刻みでなめらかに調整することができる。

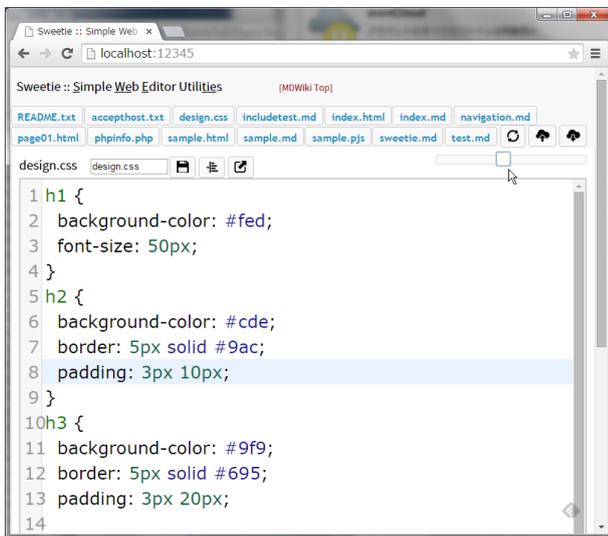


図 6 スライダによる文字サイズ調整

Markdown 対応

Sweetie は HTML/CSS/Javascript 等を編集するだけでなく、Markdown 形式のファイル (.md) を作成するだけで、Web ページに変換して閲覧することもできる。これは、MDWiki[4] の機能を利用している。MDWiki は、Markdown 形式のファイルを読み込み、HTML に変換して表示することができる特殊な HTML ファイル (Javascript と CSS が埋め込まれている) を提供している。MDWiki を用いることで、図 7 のように記述した Markdown 形式のテキストから、図 8 の表示を得ることができる。なお、MDWiki は Bootstrap を使用しているため、レスポンシブデザインに対応しており、スマートフォンやタブレット等の小さな画面でも閲覧しやすいサイトを簡便に作成することができる。



図 7 Markdown エディタ



図 8 Markdown ページの表示 (MDWiki)

画像ペースト・ドロップ操作によるファイルアップロードとタグ挿入

その他、HTML や Markdown による Web サイト構築を支援する機能として、クリップボードに画像が保存されている状態で、HTML/Markdown 編集エディタ画面で「貼り付け (CTRL+V)」操作を行うとクリップボード画像の内容からサーバ側に PNG ファイルを生成したうえで、画像埋め込みタグ () を挿入する機能を備える*4。関連して、画像ファイルをエディタ画面にドラッグ&ドロップすると、自動的にファイルをアップロードして画像埋め込みタグを挿入する機能や、画像以外のファイルをエディタ画面にドラッグ&ドロップすると、ファイルをアップロードしたのち、ダウンロードリンク (<a>) を挿入する機能を備える。これらの画像・ファイルアップロード機能は、複数人で 1 つの Web サイトを構築する演習場面ではファイルの転送と配置を意識する必要がないため、学習者は文書生成タスクに注力することができる。

発展としての PHP や Processing プログラム対応

Sweetie は初学者のための Web 開発環境として設計しているが、発展として、PHP を用いた動的な Web ページ作成や、Web 版の Processing プログラミングを行うことも可能である。拡張子を [.pjs] としたファイルを作成すると、Processing.js の編集に適したシンタックスハイライトが行われる。また「出力を確認」ボタンをクリックするか CTRL+R を押すことで、編集中のプログラムを別タブで実行することができる。これにより、インタラクティブなアニメーションを作成し、サイトに反映することも簡単に行える。

Web アップロード機能

学習者が作成した Web サイト「作品」を教師が集約したり、学習者が提出しやすくするため、我々は Web アップロード機能を追加した。Web アップロード機能は、学習者が編集したファイル群を、教師が準備しておいたインター

*4 (注) クリップボードからの画像生成機能は、PHP の GD ライブラリが利用できる場合に限られる。

ネット上のサーバにアップロードする機能である。ファイル送信先の URL は Sweetie のスクリプトにあらかじめ書き込んでおき、グループ名とパスワードは機能呼び出すときに入力する。集約サーバ側には、専用の PHP スクリプトを配置し、ファイルのパーミッション設定を済ませておく。これにより、学習者は FTP クライアントを用いずに、手軽に Web サイトをサーバにアップロードすることができる。

2.3 実装の詳細

Sweetie は、PHP 言語で書かれたスクリプトとして Web アプリケーションを実装している。そのため、PHP を有効にした Web サーバをローカルホストで起動する必要がある。WindowsOS が動作する学生計算機上で、PHP を有効にした Web サーバを起動する方法として、我々は QuickPHP [3] を利用した。QuickPHP はインストール不要で動作する軽量 Web サーバであり、任意のフォルダをドキュメントルートとして起動することができる。我々は、**図 9** に示すように、カレントディレクトリをドキュメントルートとしてローカルホストの 12,345 番ポートを使用して起動した。これは以下の内容を記述したバッチファイルによっても実行できる。

```
QuickPHP.exe /Port=12345 /Root=. /Start
```

なお、QuickPHP はライブラリと実行ファイルを含めたファイルのサイズが 7MB 程度と比較的小さいため、各学生がダウンロードしても支障は少ない。

学生計算機が MacOS や Linux の場合は、以下のコマンドをドキュメントルートで実行することで、手軽に PHP を有効にした Web サーバ (PHP Built-in Server) を起動できる。ただし、この場合はあらかじめ学生計算機に PHP5.4.0 以降がインストールされている必要がある。

```
$ php -S localhost:12345
```

通常、Web サーバを起動したユーザと、ファイルを所有しているユーザは同一であるため、Web サーバを介したファイル読み書きや新規フォルダの生成には制限はかからない。そのため、ローカルファイルの編集やファイルの新規アップロード等を比較的自由に行うことができる。

他の計算機からの不用意な編集を防ぐため、Sweetie では編集機能を実行可能な計算機を IP アドレスによって制限する機能を備えている。初期状態ではサーバと同一の計算機のみを許可しているが、学生が必要に応じて、許可したい計算機の IP アドレスをテキストファイル [accepthost.txt] に追加することで、グループの他のメンバーに編集機能を公開することができる。なお、Web サイトの閲覧については現在制限を設けていない。Basic 認証機能をつかって保護するための仕組みも準備している。

Sweetie の実装にあたって、ファイルボタンやスライ

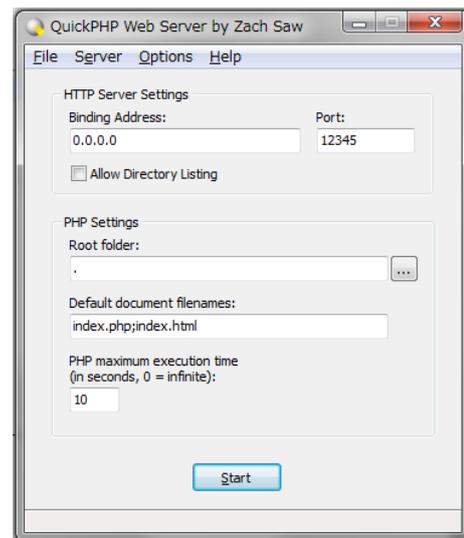


図 9 QuickPHP の設定画面

ダは JQueryUI*⁵、シンタックスハイライト機能やインデント機能を含むエディタ部分は CodeMirror[5] を用いた。CTRL+S で保存する、CTRL+R で出力確認するといったショートカット機能や、ファイル/クリップボード画像をドロップ/ペーストしたときのファイルアップロード処理およびタグ/スニペット挿入については、自作のスクリプトで実現した。

2.4 利点と欠点

ポータブルな Web サーバと、編集用の Web アプリケーションという方式を採用することの利点として、以下が挙げられる。

- (管理者権限不要) ソフトウェアをインストールするための管理者権限がない場合でも、シンタックスハイライトや自動インデント機能を備えたエディタを利用できる。
- (サーバ・ネットワーク負荷軽減) 学生計算機上で Web サーバを起動するため、1つのサーバですべての編集と保存の処理を賄うのに比べて処理が分散される。またネットワーク負荷も軽減できる。万一、外部ネットワークに接続できなくなった場合でもローカルネットワークが機能していれば、演習を継続できる。Web アップロード機能で成果物を集約する場合でも、サーバにかかる負荷は軽減できる。
- (ディスク容量) ポータブルな Web サーバが比較的軽量で、ファイルサイズが小さければ、各学生がコピーを所持したとしてもディスク容量をそれほど圧迫しない。MacOS や Linux の場合は、PHP5.4.0 以降をインストールする必要があるが、各学生がポータブル Web サーバのファイルを所持する必要がなくなるため、必要なディスク容量は軽減できる。

*5 <https://jqueryui.com/>

- (教育効果) 学生計算機上で Web サーバを起動することによって、ブラウザで指定する URL の IP アドレス部を変更すれば、他の人のサイトを閲覧することが可能となる。たくさんのサーバが同時に起動しており、ブラウザは URL によってサーバを特定し、コンテンツの入手が可能、という状況は、Web サーバの概念を理解させるうえでも有益であると考えられる。またローカルファイルの変更が設定によっては他人から行われてしまうことから、サーバに対する攻撃を理解する必要があるため、体験を通じてセキュリティの意識を高める効果が期待できる。

また、ローカルホストで実行できることと、機能を自由に追加したり、カスタマイズしやすいことから、以下の利点が得られる。

- (ローカルファイル) ファイルはローカルホストに保存されるため、どのような状況になっているのかをファイルエクスプローラ等で確認しやすい。ファイルのコピーやリネームも、直接ローカルのファイルを操作することで実行できる。
- (対応するファイル形式) HTML や CSS, Javascript に加え、Markdown や PHP, Processing プログラムコードなど、多様な形式のファイルを扱うことができる。また、それぞれの編集結果について、簡単に動作確認をおこなうするための仕組みが備わっている。
- (画像埋め込み支援) HTML や Markdown ファイルの編集時に、クリップボード画像をペーストしたり画像ファイルをドロップすることで、自動的に画像埋め込みのタグが追加できる。また Markdown 対応の Wiki が組み込まれているため、共有メモやドラフト版、構成検討用の Web ページを簡単に作成できる。

提案方式の欠点としては、もし学生用計算機のファイアウォール設定により、自由なポート番号での通信が許可されていない場合は、外部ホストからの接続ができないため、グループメンバー間での利用はできない。ただしその場合でも、ローカルホストでの編集やファイルの Web サーバアップロード機能は使用できると考えられる。

3. 関連研究・システム

Baggaley[6] は、USB メモリに入れて持ち運び可能な、教育に利用できるポータブルアプリケーションとして Firefox, Gaim, Filezilla 等を紹介し、その柔軟性と有効性について論じている。Kompen ら [7] は、Web2.0 的なツールやサービスを利用した個人学習環境 (Personal Learning Environment) を通じた学習方策とガイドラインを提案している。これらの研究は一般的に利用されているツールやネットワークサービスの利用を想定している。本研究では、Web サイト構築演習とそれに付随する学習を円滑にすすめることを目的として、ポータブルな Web サーバ環境

を用いたシステムを提案している。

複数人でのファイル共有を行うためのシステムとして、Dropbox や ownCloud, Bittorrent Sync 等が知られている。これらのファイル共有システムは基本的にインストールが必要になる。ownCloud は、サーバに Web ブラウザでアクセスして、ファイルを編集するシンタックスハイライトエディタ機能を備えている。しかしインデント機能やファイル貼り付け機能は提供していない。

Windows で動作する高機能な日本語テキストエディタのうち、サクラエディタ*6, TeraPad*7, xyzy*8 等は、インストール不要で動作する。これらのエディタは高機能かつ軽快に動作するが、複数人でファイルを共有して編集することはできない。

近年では Google ドキュメント*9 や、ShareLaTeX[8] のように、複数人で1つのドキュメントを同時に編集することが可能な Web サービスが登場している。しかし、これらのサービスは利用にあたって登録が必要である。EtherPad の後継である Stypi[9] という Web サービスは、登録が不要で手軽に HTML や Markdown の複数人同時編集が行え、またライブビュー機能により、編集結果が即座に閲覧画面が反映されるため、有用性は高い。ただしテキスト編集に特化しているため、画像を含む Web サイトを自由に構築することはできない。

4. 運用

筆者が担当している複数の講義において、Sweetie を用いた演習を行った。S 大学の「情報の表現法」の講義では、PowerPoint や Excel で図やグラフを作成する単元のあと、それらを活かした総合的な活動として、Web サイトを作成する演習を行っている。最初に、Web ページや Web サイト、URL 等の基本概念について説明したのち、Sweetie を自分のフォルダにコピーして、バッチファイルを実行して QuickPHP を起動してもらった。

受講生によっては、Sweetie のフォルダを、日本語のフォルダ名の下に含めてしまったことで、QuickPHP の動作が不安定になったり異常終了するといった事例がみられた。また、HTML ファイルを編集する段階で、IME を漢字入力モードにしてひらがな入力を行った際、変換中を示す下線が表示されなかったため、Enter を押して確定するまでは CTRL+S キーによる保存ができないといった問題がみられた。後者については、保存ボタンを追加し、これを押すことによって回避できるようにした。

その後、学生計算機の IP アドレス採番ルール (192.168.12.xxx) を説明して、隣の計算機で動作するサーバ

*6 http://osdn.jp/projects/sfnet_sakura-editor/

*7 <http://www5f.biglobe.ne.jp/~t-susumu/>

*8 <http://xyzy-022.github.io/>

*9 <https://www.google.com/docs/about/>

に接続する方法を説明した。その際、他の計算機からコンテンツを編集可能にする方法の1つとして、`accepthost.txt`に許可したい計算機のIPアドレスを指定する方法を説明した。なお、S大学の計算機環境ではセキュリティの観点から、一般ユーザはコマンドプロンプトを開けない設定となっていたため、`ipconfig`によって自分が使っている計算機のIPアドレスを調べる方法が使用できなかった。そこで、図1に示すように、Sweetie 起動画面にはサーバとクライアントのIPアドレスを表示して、自分が使っている計算機のIPアドレスを確認できるようにした。

Webサイト構築演習は、あるひとつのテーマについて、2~3名のグループで調べてまとめたものを、図やグラフを作成・参照しながら複数のページで構成するという課題となっている。作成方法としては、(1) 素のHTML, CSSで記述する、(2) MDWikiで記述する、(3) サーバ上に用意したWordPressで構築する、のいずれかを選択できるようにした。そのため、とくに(1)(2)について、グループで分担して作成・編集しやすい環境の提供は、全員が編集・活動に参加すべきという観点からも重要であると考えている。従来(昨年度)の講義では、ホームページビルダーを使用してWebサイト構築演習を行ったが、学生間でHTMLファイルや画像ファイルの移動やマージに手間取るケースが多かった。またホームページビルダーの雛形はデザインが洗練されており有益ではあるのだが、その半面として、ヘッダ画像やロゴ画像の差し替えなど、本質的でない部分に学生が多く時間を費やしている印象があった。

5. まとめと今後の課題

学生用計算機で動作するWebサーバを利用したファイル編集・確認環境 Sweetie を提案した。インストールのための権限が不要であり、またシンタックスハイライトやインデント機能等、必要最小限の編集支援機能を備えているため、初学者にとって最初の敷居は低い。Markdown記法を用いたサイト作成や、PHPの記述、Web版のProcessingプログラミングにも対応しており、静的なWebページに限定しない、様々なWeb開発用途に利用できる。また学生がWebサーバを起動することにより、サーバの役割を体験的に理解したり、セキュリティ意識をもたせることができることも教育的な観点からみたときのひとつの利点であると考えている。

Sweetieはファイルの拡張子によって様々な形式のファイルについてシンタックスハイライトを適用することができる。そのためWeb開発に限らず、C言語やJava言語のプログラムをグループで共有しながら編集するような用途にも適している。

現在のSweetieの実装では、ひとつのファイルを複数人が同時に編集し、保存すると問題が発生する場合がある。今後はStypi[9]やShareLaTeX[8]のように、ひとつのファ

イルを複数人で同時並行編集できる仕組みを導入し、提案手法におけるグループによる協働編集の有用性を高めたいと考えている。

Sweetieは以下のURLでダウンロードできる。<http://ist.mns.kyutech.ac.jp/miura/sweetie/>

謝辞 本研究の一部は公益財団法人電気通信普及財団の支援によるものです。

参考文献

- [1] 兼宗進, 中谷多哉子, 御手洗理英, 福井真吾, 久野靖. 初中等教育におけるオブジェクト指向プログラミングの実践と評価. 情報処理学会論文誌, Vol. 44, pp. 58-71, 2003.
- [2] 長慎也, 甲斐宗徳, 川合晶, 日野孝昭, 前島真一, 笥捷彦. プログラミング環境 Nigari: 初学者がJavaを習うまでの案内役. 情報処理学会論文誌: プログラミング, Vol. 45, No. 9, pp. 25-46, 2004.
- [3] Zach Saw. QuickPHP - Lightweight Standalone Bloat-free Portable PHP Web Server. http://www.zachsaw.com/?pg=quickphp_php_tester_debugger. (2015年5月19日確認).
- [4] Timo Drr. MDWiki - Markdown based wiki done 100% on the client via javascript. <http://mdwiki.info/>. (2015年5月19日確認).
- [5] Marijn Haverbeke. CodeMirror - versatile text editor implemented in JavaScript for the browser. <https://codemirror.net/>. (2015年5月19日確認).
- [6] Jon Baggaley. Portable applications in mobile education. *The International Review of Research in Open and Distributed Learning*, Vol. 7, No. 2, 2006.
- [7] Ricardo Torres Kompen, Palitha Edirisingha, and Richard Mobbs. Building Web 2.0-based Personal Learning Environments—A Conceptual Framework. In *EDEN Research Workshop 2008*, October 2008.
- [8] Henry Oswald, James Allen, and Brian Gough. ShareLaTeX, the Online LaTeX Editor. <https://www.sharelatex.com/>. (2015年5月19日確認).
- [9] Jason and Byron. Stypi - realtime editor that allows multiple users to make changes to a single document at the same time. <https://code.stypi.com/>. (2015年5月19日確認).