

Sweetie Framework: Simple but Practical Web Application Development Environment

Motoki Miura

Department of Basic Sciences, Faculty of Engineering, Kyushu Institute of Technology
1-1 Sensui, Tobata, Kitakyushu, Fukuoka, 804-8550 JAPAN

Email: miuramo@mns.kyutech.ac.jp

Web: <http://ist.mns.kyutech.ac.jp/>

Abstract—In order to build a web application, understanding of the programming language, database, SQL, and HTTP is indispensable. Therefore, for beginners, the required skills of the web application development are higher than that of the web page authoring. We propose Sweetie framework that can easily learn basic concept of simple web applications and construct them. Sweetie framework is designed to work with Sweetie, a lightweight web development editor that runs on a web browser. By utilizing the Sweetie framework, novice developers can easily handle, retrieve, and display data in the database. In addition, the Sweetie framework provides user authentication and user registration function for practical applications. We have conducted a web application development lecture using Sweetie and Sweetie framework.

Index Terms—Web-based Creative Task, Novice Web Developer, Web Framework, IDE

1 INTRODUCTION

Authoring web documents is one of the popular activities in introductory computer literacy lecture courses. By authoring web documents, learners can understand the concept of markup languages, structure of hypertext, and representation of the web pages. Developing web applications is an advanced course of the static web document authoring. Knowledge and skills of developing web application is helpful for realizing their own collaborative shared virtual workplaces. Once learners learned such skills, they can design a creative, effective, and efficient communication method based on web infrastructure.

Generally, in order to construct a web application, it is necessary to understand the programming language for dynamic page generation. In addition, knowledge about databases and SQL for saving data and understanding about HTTP are indispensable. Therefore, for beginners, the required skills of the web application development are higher than that of the web page authoring.

There are many web frameworks that help developers to build and manage web applications. Ruby on rails [1] (RoR) is a popular web framework that provides ActiveRecord and MVC architecture. However, mastering the RoR requires much time and efforts especially for novice learners. Supaartagorn also proposed a simple PHP-based web framework based on MVC architecture [2]. The framework provides simple CRUD (create, read, update, delete) methods for data manipulation. However, the framework does not consider the authentication function that is crucial for developing a web application with actual users.

We propose Sweetie framework, a web framework that makes it easy to construct simple but practical web applications. The Sweetie framework provides CRUD and authentication functions with administrative pages. Since the Sweetie framework employs PHP-based notation rather

than HTML, the learner can develop a web application in a short period of time. For improving usability, we bundled the Sweetie framework with web-based online editor named Sweetie. Therefore, the teacher can provide optimized web IDEs for novice learners.

2 SWEETIE

Before explaining the Sweetie framework, we introduce Sweetie (Simple Web Editor Utilities) [3], which is a simple lightweight web editor that runs on a web browser. By utilizing Sweetie, users can edit server-side files (HTML, css, Javascript, Markdown, etc.) easily. Sweetie was originally developed to help novice learners construct static web pages efficiently. Therefore, Sweetie provides syntax highlighting, short-cut keys, auto-indent and quick file/image upload functions.

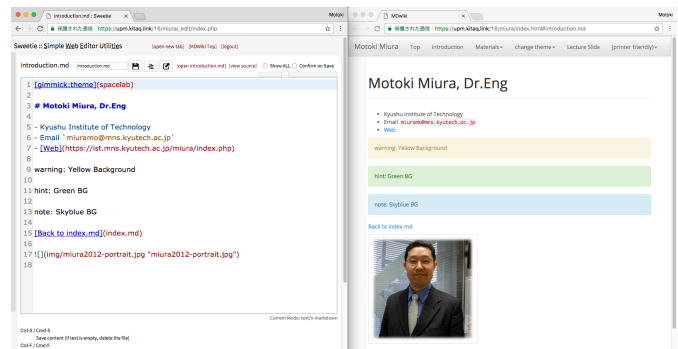


Fig. 1. Sweetie editor editing a markdown file (left), and the page view (right).

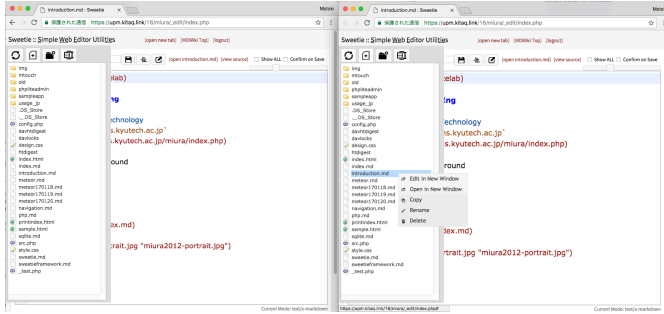


Fig. 2. Sweetie file manager (left), and its context menu (right).

Figure 1 left shows the Sweetie editor editing a simple markdown document. The editor supports syntax highlighting of multiple file formats, e.g. HTML, CSS, JavaScript, Markdown, and so on¹. After editing a file, learners can use either the save icon button or a keyboard shortcut (CTRL+S) to save the file directly on the server. Learners can use the confirm icon button (or CTRL+R keyboard shortcut) to confirm the content of the saved file. This operation opens a new web browser tab and displays the edited content. The learner can add images by drag and dropping files on the editor, or pasting clipboard image. Sweetie editor uploads the image files to “img” folder on the server, and automatically inserts a markdown text (or HTML tags) to show the images. Figure 1 right shows the rendered page view of the markdown file. The markdown files with an .md extension are dynamically converted to HTML using MDWiki [5]. Since the MDWiki utilizes Bootstrap styles, the converted page is also suitable for small screen devices, such as smart-phones and tablets.

The Sweetie editor provides CTRL+/ shortcut key to toggle commenting. It can toggle (un-)commenting of markdown, HTML, and PHP scripts. When multiple lines were selected, the shortcut key toggles these selected lines. This function is useful for learners to test the code snippets by enabling/disabling with a simple operation.

The learner can open multiple files from a file manager (see Figure 2 left). The file manager only appears when the learner hovers on the hidden file manager. The file manager allows the learner to confirm the server-side files. When the user clicks a file name with the right-button, the context menu appears (see Figure 2 right). The context menu can be used to change the file name, copy, and delete on the server-side.

Sweetie simplifies the web authoring work by eliminating file-transferring tasks. The learners can immediately check the rendered page view after editing the contents. Teachers and administrators can also benefit from Sweetie, because no text editors are necessary for conducting lectures. Using a starter script of Sweetie, the teacher can allow the learners to create Sweetie folder on the server with their favorite name.

Regarding related works and systems, there are several web services similar to the Sweetie environment. JSFiddle [6] and jsdo.it [7] can share HTML, CSS, and JavaScript

1. The syntax highlighting function is provided by the CodeMirror [4] editor.

snippets using a simple interface. The snippets written by others can be reused by forking items. These services are suitable for sharing simple snippets but not for compound pages and sites because the URL for the snippets is assigned randomly. With Sweetie, users can utilize an authoring environment that is similar to JSFiddle and jsdo.it on their own managed servers. Google Docs and ShareLaTeX [8], enable groups of users to edit content simultaneously. Even though such services require registration, their approach is similar to our motivation. The Stypi [9] web service, formerly EtherPad, does not require registration. With Stypi, multiple users can write text files, such as HTML and Markdown, collaboratively. The Stypi live-viewing function renders a preview of the content immediately. However, the service is specialized for text files, and users cannot construct a bundle of web pages that include figures and images.

3 SWEETIE FRAMEWORK

We constructed the Sweetie framework coupled with the Sweetie web editor, for novice learners to quickly understand the outline of the technology required for web applications. With the Sweetie framework, the novice learner mainly writes small PHP scripts that call the framework functions, rather than HTML tags. Therefore, the users of the Sweetie framework are required a rudimentary knowledge of PHP, but detailed syntax knowledge of HTML is not required.

3.1 Features

Sweetie framework includes the following features to reduce the extra burdens of web application development for novice learners.

- To make the inserting and updating of the DB easier, basic functions for handling SQLite databases are provided. The functions also support BLOB (binary large objects), so the novice learner can develop a file/image upload form as easy as storing text data.
- The framework provides a function to create simple CRUD (create, read, update, and delete) interfaces for any SQLite tables. The created interfaces involve AJAX for simple and intuitive operations.
- The framework provides a simple HTML helper and a form helper written by PHP language. For example, `br(5)` is converted to the five `
` tags, and `heading("H3", 3)` inserts `<h3>H3</h3>`. For developing of smart-phone friendly pages, embedding QR code function is also provided.
- Functions of user authentication and user registration can be easily incorporated. For the administrator pages, an authentication mechanism of Sweetie editor can be adopted by a simple function call.
- We provide functions to make graphs from table data using Google Graph API.

We adopt PHP language for the development of the web applications with the Sweetie framework. The main reason to adopt the PHP code style rather than HTML tags is simplicity. Of course, we can build a dynamic web page by mostly HTML notations with some PHP embedding code. However, writing the HTML tags requires more texts

`require_sweetielogin()` (line 9) requires the same authority as Sweetie's editor privileges. By calling this function, the developer can easily control access from unauthorized anonymous users.

`showtable_withdeleedit()` (line 17) displays a table with links of deletion and editing correspond to each row. `jqaddform()` (line 18) embeds an invisible form using JQuery and Ajax. When the user presses "Add tweets", the invisible form appears as Figure 7. Pressing "Edit" link on each row inserts an edit form as Figure 8. The contents of these forms are dynamically organized by Sweetie framework considering the fields and attributes of the table schema.

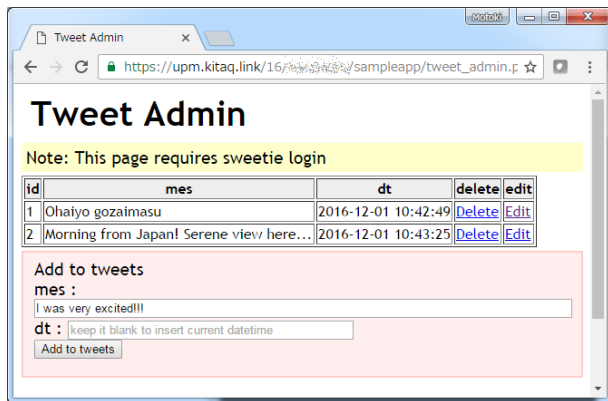


Fig. 7. Pressing "Add tweets" button appears the input form for "Add". If the date-time (dt) field was left blank, current date-time is inserted.

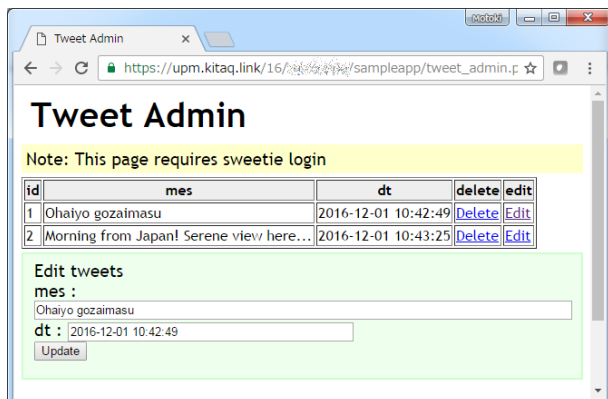


Fig. 8. Pressing "edit" link appears the input form for "Edit".

3.2.2 Shop application

Figure 9 shows an administrative interface of a shop application. The shop application consists of three tables (users, items, and logs). The interface not only shows these table contents but also a joined table for the purpose of explaining a relational database. The novice learner can understand the merit of the relational database through utilizing the typical application. The administrative interface is also controlled by `require_sweetielogin()`.

Figure 10 shows a user interface of the shop application. This interface requires a preliminary user registration which is realized by calling `require_login("DBfilename")` function at the top of the PHP script. This function checks a session, and

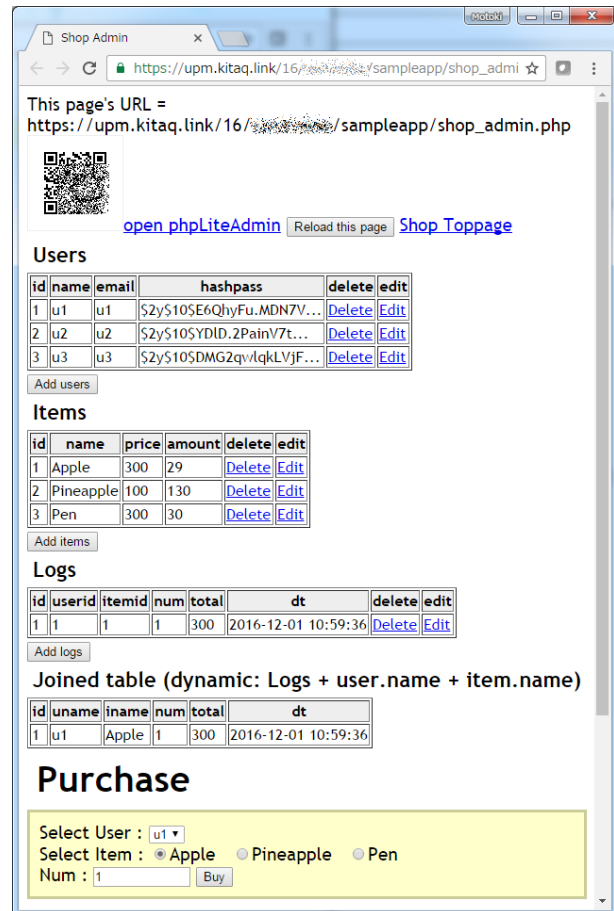


Fig. 9. An administrative interface of the shop application

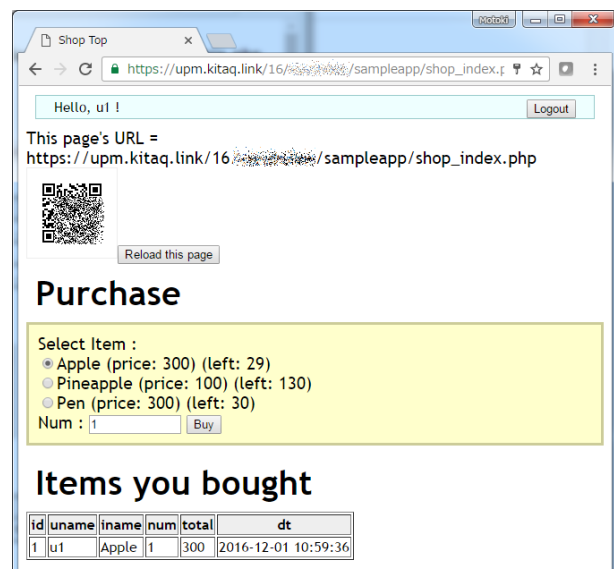


Fig. 10. A user interface of the shop application

redirects to the login/register page (see Figure 11) if the session is invalid. Unregistered users create their account and register through an email authentication process. Figure 10 shows a screen that the user successfully logged in. The user name and logout button shown in the top of Figure 10 are inserted by the `require_login("DBfilename")` function. For de-

velopers, `$_SESSION['u']['id']` and `$_SESSION['u']['name']` can be utilized for the user ID and user name respectively in their PHP script. By providing these login/authentication functions, novice learners can focus on the learning of the fundamental part rather than complicated techniques. Note that the QR code can be easily inserted with the `showqr-code("URL")` function. The function eases the web access from smart-phones.

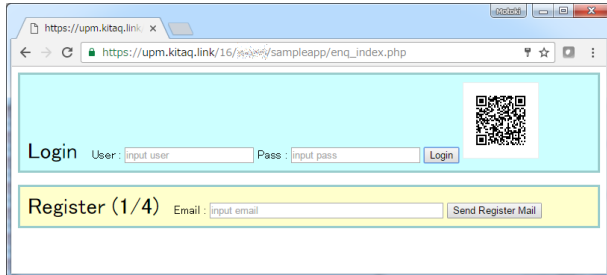


Fig. 11. Login page of the shop application

3.3 Additional functions

This section describes the additional functions of the Sweetie framework.

3.3.1 Modifying table contents

The Sweetie framework provides functions to modify table contents. The table content is normally defined by two-dimensional associative array of PHP. `aryinscol(array, string, column name [, column number])` inserts a key-value pair of "column name" and "string" for each inner array of the array. The "column name" is utilized for the table header. The "string" can contain "column name" variables for inserting the values (such as `"$id"`). `arymapcol(array, target column name, function)` performs rewriting by the function with respect to each row data of the specified column. By using the function, the developer can customize table contents with minimal description. Examples of these function calls are shown in Figure 12.

```

54 $ut = tbl($db,"users"); // get table contents as two-dimensional associative array
55 $table = "users";
56 $ak = getaccesskey($dbfn,$table); // get special token for deleting row
57 aryinscol($ut, "<a href='_delete.php?ak={$ak}&id={\$id}'>Delete</a>", "delete"); // add [Delete] link
58 aryinscol($ut, "<a href='#{$table}{\$id}' onclick='jqedit(\"{$ak}\", \"{\$id}\", \"{\$table}\");'>Edit</a>", "edit"); // add [Edit] link
59
60 //arydelcol($ut,"hashpass"); // drop column
61 arymapcol($ut,"hashpass",function($a){ return substr($a, 0, 20)."..." ; }); // truncate long hash string
62

```

Fig. 12. Modifying table contents

4 RELATED WORKS AND SYSTEMS

Amaya [11] is a web editor/browser to promote collaboration on the web. With Amaya, users can easily create web pages and upload them to the server. The basic concept of editing with a browser is similar to Sweetie. However, Amaya targets static document editing such as HTML, CSS, XML, MathML, SVG. We aim for Sweetie and Sweetie framework to make it easy to build web applications that behave dynamically, even for beginners, with browser only.

CodeSync [12] is a collaborative coding environment for novice web developers. With CodeSync, the learners can simultaneously and collaboratively edit a shared source code with sharing modifications like Stypi. Therefore, the learners can work with a web development project with real-time sharing of file contents. CodeSync was implemented using Node.js, Ajax, and MongoDB with Socket.io technology. CodeSync was applied for static web development (HTML, CSS, and Javascript). The Sweetie framework was intended to reduce the burdens of learning of initial web application development.

As PHP frameworks for lowering the development load of web applications, CakePHP, Laravel and Codeigniter are commonly known. Since these frameworks have many functions to fulfill any type of requirements for complete web applications. Thus, developers have to take much time to understand the frameworks. The Sweetie framework consists of only six files in the framework alone. Although the functions are restricted, it is suitable for novice learners to build their first web applications using PHP.

5 APPLIED LECTURES

5.1 Three-hour experimental lecture

We conducted a web application development lecture using Sweetie and Sweetie framework as a part of short-term mobility program. Nineteen short-term visiting students at Putra University in Malaysia² (Figure 13) joined the lecture. The total lecture time was 3 hours and 10 minutes, 90 minutes \times 2 consecutive periods, including a 10 minutes break.

First, students created Sweetie folders with names of their choice in their browsers. Figure 14 left shows the interface of creating Sweetie folder. When the student inputted a folder name, a complete set of templates including sample web applications and lecture materials is copied to the specified folder. Then, the student were notified his/her web site URL, the editor URL, and password (see Figure 14 right). Next, I described my self-introduction page with Markdown and confirmed that it is reflected on the web. Through these operations, students got accustomed to the operation of the Sweetie editor and shortcut keys. After that, we explained fundamentals of PHP, such as echo, date, variables, and associative array during the first period.

At the beginning of the second period, we introduced a tweet application, photo upload application, shop application as examples of simple web application. We explained SQLite database with sample databases by phpLiteAdmin [10]. We showed how to confirm table schemes and modify them. We also introduced the authentication functions through the PHP scripts of the sample applications. Most of the students had their smart-phones. Thus we recommend the students to utilize QR codes, and memorize experiences of stay in Japan. We emphasized the importance and merits of web application, and demonstrated the easiness of the development through the Sweetie editor. We also explained how to add field to the existing table for storing extra text data. Figure 15 shows an example of using the photo upload application that students have modified. The student

2. <http://www.kyutech.ac.jp/whats-new/topics/entry-4088.html#prettyPhoto>

added a comment field to the table of the photo uploading application, and uploaded some pictures with comment.

Actually, not all students could understand the syntax of PHP, and modify the sample web applications in the short-term lecture. To make fully understand the functions and mechanisms, we need much time at least five hours. However, the students could recognize the summary of the web application development through the experience of utilizing sample applications on the Sweetie editor. Also, we could provide the online learning/development environment continuously after the short-term lecture.



Fig. 13. Three-hours experimental lecture

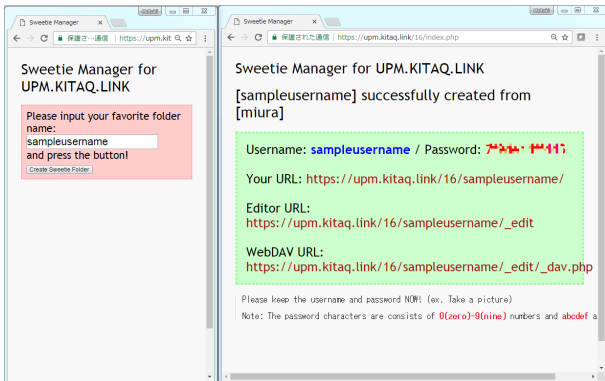


Fig. 14. Creating Sweetie folder (We also provides WebDAV URL for advanced students.)

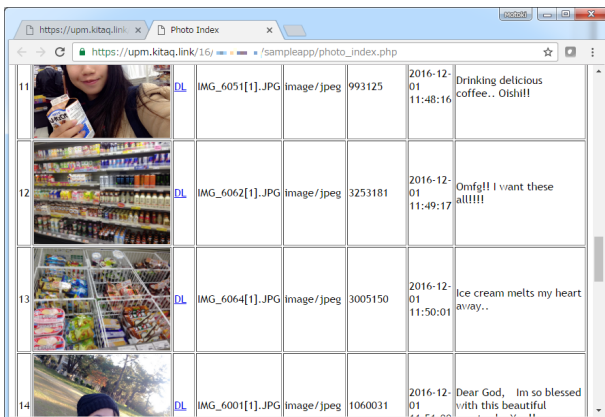


Fig. 15. Modified photo upload web application by a student

6 CONCLUSION

We introduced Sweetie framework, a web application framework that runs on Sweetie, along with use cases. Web

application development requires a wide range of knowledge and skills such as databases, SQL, HTTP and forms. Sweetie and Sweetie framework can ease the burdens of the first web application development that uses PHP and SQLite databases with access controlling by user authentications. We believe that the Sweetie framework is suitable for the first stage learning of the web application development because several sample applications can be easily provided. The Sweetie framework hides the complicated part of web applications such as session, HTTP, and PDO database access. Thus the novice learners can focus on the principal topics and knowledge.

There are several disadvantage of learning by Sweetie framework. One possible issue is a validity of the HTML generated by the PHP script. Since we mainly utilize PHP helper functions for web authoring, the output might not fulfill an agreement of HTML. For the future work, we will introduce a mechanism that automatically corrects output content to valid HTML. Also, there are arguments that Sweetie framework approach will encourage learners to neglect efforts to fulfill an agreement of HTML. We consider that the moral could be trained after the learning experiences with the Sweetie framework.

Sweetie and Sweetie framework can be downloaded from the URL. <http://ist.mns.kyutech.ac.jp/miura/sweetie/>

ACKNOWLEDGMENT

The part of this research was supported by the fund of Telecommunication Advancement Foundation and JSPS KAKENHI Grant-in-Aid for Scientific Research (C): Grant Number 15K00485.

REFERENCES

- [1] M. Bächle and P. Kirchberg, "Ruby on rails," *IEEE software*, vol. 24, no. 6, 2007.
- [2] C. Supaartagorn, "Php framework for database management based on mvc pattern," *International Journal of Computer Science & Information Technology (IJCSIT)*, vol. 3, no. 2, pp. 251–258, 2011.
- [3] M. Miura, "Lightweight Web Authoring Environment," in *20th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES2016)*, vol. 96, Sep. 2016, pp. 887–895.
- [4] M. Haverbeke, "CodeMirror – versatile text editor implemented in JavaScript for the browser," <https://codemirror.net/>, (Visited: March 21, 2017).
- [5] T. Drr, "MDWiki – Markdown based wiki done 100% on the client via javascript," <http://mdwiki.info/>, (Visited: March 21, 2017).
- [6] "JSFiddle: Online code editor and playground. Front End developers hub." <https://jsfiddle.net/>, (Visited: March 21, 2017).
- [7] K. Inc., "jsdo.it — Share JavaScript, HTML5 and CSS," <http://jsdo.it/>, (Visited: March 21, 2017).
- [8] H. Oswald, J. Allen, and B. Gough, "ShareLaTeX, the Online LaTeX Editor," <https://www.sharelatex.com/>, (Visited: March 21, 2017).
- [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/>, (Visited: December 17, 2015).
- [10] "phpLiteAdmin: The Web-based Database Management Tool for SQLite." <http://www.phpliteadmin.org/>, (Visited: March 21, 2017).
- [11] V. Quint and I. Vatton, "An Introduction to Amaya," W3C NOTE 20-February-1997, <https://www.w3.org/TR/NOTE-amaya-970220>, Feb. 1997.
- [12] M. S. Feldman, "Codesync: A collaborative coding environment for novice web developers," http://cs.wellesley.edu/~webdb/lectures/Node/Monica_Starr_Feldman_Thesis.pdf, Apr. 2014.