

# ブラウザの開発者ツールを用いたオンラインレビューツールの提案

橋浦研究室

1175143 高橋 黎

1175246 吉成 知晃

## 1. はじめに

Web 制作において、コンテンツレイアウトの修正の際にブラウザの開発者ツールが広く用いられている。しかしながら、ブラウザの開発者ツールは、コードを保存するための機能やレビューのための機能を備えていない。このため、開発者がブラウザの開発者ツールを用いてコンテンツレイアウトの修正を行ったとしても、その修正結果をエディタ等で転記しなければならない。また、Web 制作を複数人で行う際などに、修正結果をレビューする必要がある。この際に、修正結果をテストサーバに反映させる工程が必要であった。

## 2. 研究目的

本研究の目的はコンテンツレイアウト修正時の負担を軽減すること、また、ブラウザ上からでもエディタと同じような修正を可能にすることである。

## 3. 関連研究

関連研究として西ら[1]の研究を挙げる。西らは Web ページ作成時にかかる負担を解決するためにオンライン編集可能な Web ページ作成編集支援システムを提案している。このシステムは HTML エディタ等を使わずに Web ページ上で HTML を簡単に作成・編集できるものである。西らは小学生を対象にシステムの評価を行い、Web ページを作成できることを確認している。

西らが開発したシステムは簡単に Web ページ上から作成できる一方、表示される Web ページは実際の内容とは大きく異なっているという問題がある。また、レビューに関する機能を備えていないため、複数人での開発には向いていないという問題があった。これに対して、本研究ではブラウザの開発者ツールを使用することで実際に表示される見目で編集でき、レビューを行う環境を整えることで Web ページ開発を支援する。

## 4. 提案手法

本研究では、前述の問題を解決するために、ブラウザの開発者ツールを用いて修正したコンテンツレイアウトの修正結果を保存し、直接 GitHub でレビューできる環境を Code Modify and Review Tools for Chrome Devtools(以下、本ツール)で提案する。

### 4.1. GitHub[2]

ここで、本研究で利用する GitHub について述

べる。GitHub はソースコードを管理するサービスの 1 つである。以下に GitHub で使用するプルリクエストとリポジトリ、マージ機能の概要について述べる。

- プルリクエスト: 変更した内容をオリジナルに反映する際の通知方法
- リポジトリ: ソースコードの管理場所
- マージ: 分岐した変更を 1 つに統合する作業のこと

### 4.2. GitHub Pages[3]

Web ページの公開は GitHub Pages を使用する。GitHub Pages は GitHub のリポジトリをそのまま Web ページのサーバとして利用できるサービスである。このサービスを利用することによりプルリクエストされた HTML のファイルを公開しているリポジトリにマージすることで Web ページの更新をすることが可能になる。

## 5. 実現手法

本ツールは GitHub Pages で公開している Web ページに対して修正を行う場合の利用を想定している。このため、公開している Web ページ上において、ブラウザの開発者ツールから修正したコードを直接 GitHub に送信しプルリクエストを用いたレビューが可能になる機能を作成する。

また、Web ページの見た目をレビューする際、現状ではテストサーバで確認するかローカル上にコードをファイルとして保存して確認する必要がある。このため、本ツールでは GitHub 上において Web ページの見た目が確認できる機能を作成する。これらの機能は Google Chrome の拡張機能[4]として実装する。これらを実現したツールの利用の流れを図 1 に示す。

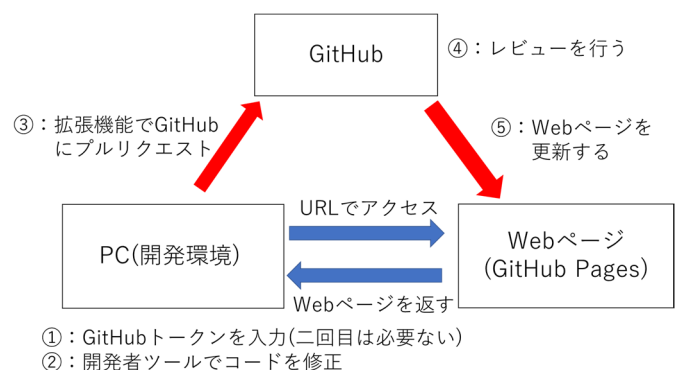


図 1 ツールの利用の流れ

図1の①はプルリクエストを送る際に必要となる情報についての初期設定を行うためのもので、入力された情報についてはローカルストレージに保存する。利用者はコード修正(②)を行い、図2のプルリクエスト送信画面を用いてGitHubに送信(③)する。プルリクエスト送信画面は、プルリクエスト時に必要な情報を修正したコード(②)のプレビューを行いながら入力する画面であり、図2のプルリクエストを行うボタンを押すことで利用者に代わってツールがプルリクエストを行う。

これによってブラウザ上から修正を行ったコードを直接保存し、プルリクエストを送信する機能を実現した。

図2 プルリクエストの送信画面

また、図1の④でレビューを行う際には、送られたコードと編集前のコードをGitHub[2]から取得し、ブラウザ上で並べて表示する。これにより修正したファイルをローカル上にダウンロードしたり、テストサーバに修正したファイルを適用したりせず、見た目のレビューを行うことが可能となった。

## 6. 実験

ツールの効果を確認するために、日本工業大学でHTMLについての講義を受けたもの10名を被験者として実験を行った。実験の概要はツールを用いてHTMLの修正作業とそれに伴うレビュー作業を行うものである。

HTMLの修正作業については、一般的な開発環境と比較を行うために統制群には6.2で後述するVS Codeの拡張機能を使用させ、本ツールとの比較を行うこととした。

### 6.1. リサーチクエストション(RQ)

実験を実施するにあたって、以下の2つのRQを設定した。

**RQ1: ツールは機能適合性に影響を与えるか**

**RQ2: ツールは開発者の負担軽減に寄与するのか**

RQ1の機能適合性の定義は、ISO/IEC 25010[7]で定義されているFunctional Suitabilityの定義を用いる。今回の実験において修正したWebページが用意した正解のWebページにどれだけ近づけられたかということである。具体的な評価指標としては、HTMLに対する正しい修正数に着目し、その数を比較することとした。

RQ2を評価するための指標として、作業時間に着目しすることとし、本ツールによって作業時間が短縮しているかを確認する。作業時間は修正準備時間、修正時間、プルリクエスト時間(以下、PR時間)、レビュー準備時間、レビュー時間を計る。修正準備時間は作業開始からコードの修正に取り掛かるまでの時間を計る。修正時間はコードの修正する時間を計る。PR時間はコードの修正後からプルリクエスト送信までの時間を計る。レビュー準備時間は作業開始からレビューに取り掛かるまでの時間を計る。レビュー時間はレビューをしてコメントを送信するまでの時間を計る。

### 6.2. Microsoft Edge Tools for VSCode[6]

今回、実験の統制群ではVisual Studio Code[5](以下、VS Code)の機能拡張であるMicrosoft Edge Tools for VSCode(以下、VS Codeの拡張機能)を使用する。これは、開発したツールを現在のソフトウェア開発では一般的に使われている考えられる環境と比較するためである。

この拡張機能はVS Code上でURLを指定することで、指定したWebページをVS Code上で表示し、編集することが可能である。これによってローカル上にあるHTMLのファイルを開くことで、VS Code上でWebページを確認しながらコードを編集することが可能である。

### 6.3. 実験方法

前述のとおり、本実験の内容はツールを用いてHTMLを修正し、修正したものに対してレビューを行うものである。これを被験者が修正完了と判断するまで繰り返す。

修正対象となるHTMLにはあらかじめ作成した仕様書をもとに14個の誤りが挿入されており、被験者は仕様書を読み解き、この14個の誤りを検出し、HTMLの修正を行う。レビュー作業については、仕様書とチェックリストを基に行う。これを、本ツールを使用した場合と使用しない場合で行う。今回使用するチェックリストはISO/IEC 2501で定義されている満足性に着目し作成した。

チェックリストによって、図や写真の挿入、文字やフォントサイズなど見た目の誤りや HTML コードの誤りを網羅的に検証できる。

#### 6.4. 評価

実験結果の概要を表 1, 2 に示す。

表 1 修正に関する実験結果

種類	修正				
	修正数 [箇所]	準備時間 [s]	修正時間 [s]	PR 時間 [s]	合計時間 [s]
本ツール	133	736	10735	4266	15624
VS Code	133	1522	8436	3868	13826

表 2 レビューに関する実験結果

種類	レビュー		
	準備時間[s]	レビュー時間[s]	合計時間[s]
本ツール	231	5820	6164
VS Code	867	6569	7436

全体として、提案ツールは VS Code と同等の能力があることが確認でき、準備時間に関しては、修正とレビュー両方で削減できているという結果となった。

ここで前者の RQ1 について議論する。表 1 の修正数に着目すると、本ツールを使用した場合と VS Code の拡張機能を使用した場合の修正数がどちらも 133 となっている。このことから、本ツールを使用した場合と VS Code の拡張機能を使用した場合差がなく、RQ2 について本ツールは影響を与えていないことが明らかになった。

次に RQ2 について議論する。修正準備時間に関して、本ツールを使用した場合は 736 秒なのに対し、VS Code の拡張機能を使用した場合は 1522 秒となっている。この結果に対して t 検定を行ったところ有意差が認められた ( $\alpha=0.05$ ,  $p=0.009$ )。これにより、本ツールによって負担軽減効果があったことが確認できた。

また、修正における準備時間分散 V に関しては、本ツールが 244.64, VS Code が 6118.16 となった。この結果に対して F 検定を行ったところ、有意差が認められた ( $\alpha = 0.05$ ,  $p = 0.00005$ )。これにより、本ツールを用いた方が VS Code よりも被験者間で修正準備時間のばらつきが小さくなっていることが確認できた。これは、本ツールによって作業が標準化されたためであると考えられる。

さらにレビューにかかった時間のうちレビュー準備時間に関して、本ツールを使用した場合は 231 秒、本ツールを使用しない場合は 867 秒とな

っている。この結果に対して、t 検定を行ったところ、有意差が認められた ( $\alpha = 0.05$ ,  $p = 0.000001$ )。これにより本ツールが開発者の負担を軽減できる可能性が示唆された。

#### 7. 考察

実験によって得られた結果から、レビュー準備時間の短縮効果が明らかになった。実際の Web サイト開発では、レビューの前に開発成果をテストサーバに反映することが必要である。この作業はレビューのたびに必要になるため、開発の作業効率化への大きな貢献が期待できる。

加えて、複数人での開発時に修正を行う際、本ツールを使用することでオンライン上から修正、レビューを行えることが確認できた。

一方で、修正のしやすさに関しては、VS Code で行った修正の方が本ツールと比較して修正時間が短い結果となった。このため、本ツールに対して今後、VS Code が持つ修正時間の短縮効果が高い機能の実装が必要である。

#### 8. まとめ

本研究はブラウザの開発者ツールを用いたオンラインレビューツールの提案を行い、実際にツールを実現して評価を行い、HTML の修正作業に対して開発者の負担を軽減できる可能性が示唆された。

今後の課題として、修正にかかる時間を短縮するためにブラウザの拡張機能上でのコード補完機能を作成することで短縮することが挙げられる。さらに、CSS や JavaScript, PHP など HTML 以外の Web ページを構成する要素を修正できるようにすることでブラウザ上からの修正の幅を広げる必要がある。

#### 文 献

- [1] 西 健太郎, 新谷 虎松, 松尾 徳朗, 田代 慎治, 伊藤 孝行, "既存 Web ブラウザを利用したオンライン編集可能な Web ページの実現," 電気学会論文誌 C (電子・情報・システム部門誌), Vol. 125, No. 4, pp. 660-665, Jul. 2005.
- [2] GitHub, Inc., "GitHub," <<https://github.com/>> (Accessed : 2021-1-14).
- [3] GitHub, Inc., "GitHub Pages について," <<https://docs.github.com/ja/free-pro-team@latest/github/working-with-github-pages/about-github-pages>> (Accessed : 2021-1-14).
- [4] Google LLC, "Extend the Browser," <<https://developer.chrome.com/extensions>> (Accessed : 2021-1-14).
- [5] Microsoft Corp., "Visual Studio Code," <<https://code.visualstudio.com>> (Accessed : 2021-1-

14).

- [6] Microsoft Corp., "Microsoft Edge Tools for VSCode," <<https://marketplace.visualstudio.com/items?itemName=ms-edgedevtools.VSCode-edge-devtools> > (Accessed : 2021-1-14).
- [7] ISO, "ISO/IEC 25010:2011," <<https://www.iso.org/standard/35733.html>>(Accessed : 2021-1-14).