

# 11 Vue.js を用いた Web アプリケーションの作成

相馬加奈

指導教員 小野 陽子

## 1. はじめに

以前から JavaScript を使ったフレームワークでのアプリケーション開発に興味があった。そのため、Vue.js を使用した Web アプリケーションを開発したいと考えた。

## 2. 研究の意義

フレームワークの導入は授業では学習しないため、その学習例を卒業研究の形で残すことで、他の生徒にとって、フレームワークやライブラリを使うハードルが下がるのではないかと考えた。今回は学生にとって手軽で使いやすい学習支援アプリを作成することにした。

### 2.1 本研究の概要

当初から計画していた学習支援アプリケーションについては、追加機能の実装を考えた時、GoogleAPI 等の連携など、追加で学習する内容が増え、本筋から逸れた内容になることが懸念されたため、より Vue.js を使って作ることに意味のあるアプリケーションも作成したいと考えた。

そのため、「標語コンクールの集計アプリケーション（以下標語アプリと表記）」の開発も同時進行で進めることとした。

### 2.2 開発環境

以下に開発環境を示す。

表 1 開発環境

IDE	Visual Studio Code
プラグイン	Vue-CLI Vuex Vue-router

### 2.3 学習支援アプリケーションについて

ToDo アプリ機能と単語帳機能を作成した。単語帳について、以下にリストページの UI と、追加画面の UI を記載する。



図 1 登録済単語のリスト表示画面

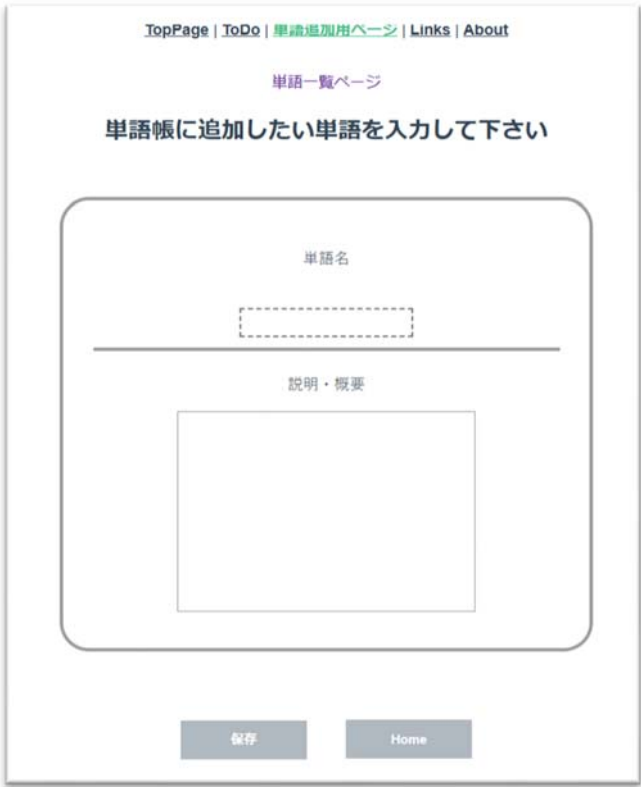


図 2 単語の追加画面

### 2.4 標語アプリについて

標語の登録画面と、投票するシステム、Node.js を使ったログイン機能を作成した。

リスト表示画面について、ジャンルごとの切り替えが出来るよう、タイプ別で切り分けをし、タブ

形式で表示ができるようにしている。

リスト表示画面で投票する標語をジャンルごとに各1つ仮選択し、投票確認画面に遷移する形になっている。

以下に標語追加画面とリストページのUIを記載する。

図3 標語アプリ 標語登録画面

ID	タイトル	タイプ	投票数	状態	投票ボタン
122	安全標語安全標語安全標語	safe	3	<input type="radio"/>	投票
121	安全標語安全標語安全標語	safe	0	<input type="radio"/>	投票
119	安全標語安全標語安全標語	safe	2	<input type="radio"/>	投票
118	安全標語安全標語安全標語	safe	2	<input type="radio"/>	投票
115	安全標語安全標語安全標語	safe	0	<input type="radio"/>	投票
113	安全標語安全標語安全標語	safe	2	<input type="radio"/>	投票

図4 標語アプリ リスト表示画面

### 3. 同様のアプリをHTML等で作成した場合との比較

#### 3.1 全体としての利点

Vue.jsを使うことの利点は、特定のページの内容をコンポーネントとして部品化して、他のページでも読み込むことが出来る点である。今回のアプリを作成する中でも、コンポーネントとビューを分けることで、ページの一部を変更したい時、どこかのコードを変更したら良いかが分かり易いと感じた。

また、ビューの中では、簡単に複数のコンポーネントを読み込むことが出来るので、表示形式なども工夫できると感じた。

#### 3.2 学習支援アプリを制作した時の利点

まず、Vue.js を用いて開発する都合上、SPA に近い構造になるのでページ遷移が速くなる。今回はローカルストレージを使用してデータの保存をしているが、Node.js との連携も可能なので、一般的なデータベースを利用することもできる。

#### 3.3 標語アプリを制作した時の利点

現在、標語を応募するフォームをコンポーネント化し、タブ上に並べる形式で表示しているが、これを JavaScript で再現しようとする、HTML コード上に全てのコンテンツの内容を記載した後に、JavaScript で表示する内容を選択し、表示するといった形になる。

Vue.js を使った場合と比べて、個々のコンテンツ内容を変更したい時にどこを変更するのかが分かり辛くなると感じた。

Vue.js を用いた場合、個々のコンポーネントが別ページになっているため、どこを変更すべきかが分かり易く、またビューページではコンポーネントの読み込みと切り替え表示のみのコードを記載するだけで良いため、動作が軽く、変更もしやすいと考える。新しい項目を増やしたい場合の拡張性にも優れている。

また標語リスト表示画面では、選択中の単語 ID を下部にリアルタイムで表示できるため、どの項目を選択しているかが視覚的に分かり易い。仕組み自体は JavaScript を用いても作成することは可能だが、Vue.js を使うとコード数も短く、簡単に同じ機能を実装できる。

#### 4. おわりに

Vue.js を用いたアプリケーション作成の一連の流れを知ることができたので今後役立つと考えた。フレームワークを使った開発は通常のページ作成より作業自体は少ないが、開発例などを調べても資料が見つかり辛いため、注意が必要だと感じた。

特に今回用いた Vue.js については、Vue3 での開発例が少なかったが、文法は分かり易く、一度理解したらページを作成するのは簡単だったため効率的に開発が出来たのではないと思う。