

## Praktikum 11: VueJS

### Tujuan

1. Mahasiswa mampu memahami konsep dasar API.
2. Mahasiswa mampu memahami konsep dasar Framework VueJS.
3. Mahasaswa mampu membuat Frontend API menggunakan Framework VueJS 3.

### Instruksi Praktikum

1. Persiapkan text editor misalnya **VSCode**.
2. Buat folder dengan nama **lab8\_vuejs** pada docroot webserver (**htdocs**)
3. Ikuti langkah-langkah praktikum yang akan dijelaskan berikutnya.

### Apa itu VueJS?

**VueJS** merupakan sebuah *framework* JavaScript untuk membangun aplikasi web atau tampilan *interface* website agar lebih interaktif. VueJS dapat digunakan untuk membangun aplikasi berbasis *user interface*, seperti halaman web, aplikasi mobile, dan aplikasi desktop.

*Framework* ini juga menawarkan berbagai fitur, seperti *reactive data binding*, *component-based architecture*, dan *tools* untuk membangun aplikasi skalabel. Fitur utamanya adalah *rendering* dan komposisi elemen, sehingga bila pengguna hendak membuat aplikasi yang lebih kompleks akan membutuhkan *routing*, *state management*, *template*, *build-tool*, dan lain sebagainya.

Adapun *library* VueJS berfokus pada *view layer* sehingga *framework* ini mudah untuk diimplementasikan dan diintegrasikan dengan *library* lain. Selain itu, VueJS juga terkenal mudah digunakan karena memiliki sintaksis yang sederhana dan intuitif, memungkinkan pengembang untuk membangun aplikasi web dengan mudah.

Untuk lebih lengkapnya dapat dipelajari pada dokumentasinya pada websitenya <https://vuejs.org/guide/introduction>

### Langkah-langkah Praktikum

#### Persiapan

Untuk memulai penggunaan framework Vuejs, dapat dilakukan dengan menggunakan npm, atau bisa juga dengan cara manual. Untuk praktikum kali ini kita akan gunakan cara manual. Yang diperlukan adalah library Vuejs, Axios untuk melakukan call API REST. Menggunakan CDN.

#### Library VueJS

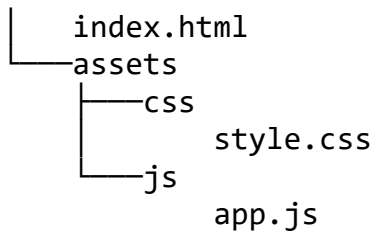
```
<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
```

#### Library Axios

```
<script src="https://unpkg.com/axios/dist/axios.min.js"></script>
```

## Struktur Direktori

Buat Project baru dengan struktur file dan directory seperti berikut:



## Menampilkan data

### File index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Frontend Vuejs</title>
  <script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
  <script src="https://unpkg.com/axios/dist/axios.min.js"></script>
  <link rel="stylesheet" href="assets/css/style.css">
</head>
<body>
  <div id="app">

    <h1>Daftar Artikel</h1>

    <table>
      <thead>
        <tr>
          <th>ID</th>
          <th>Judul</th>
          <th>Status</th>
          <th>Aksi</th>
        </tr>
      </thead>
      <tbody>
        <tr v-for="(row, index) in artikel">
          <td class="center-text">{{ row.id }}</td>
          <td>{{ row.judul }}</td>
          <td>{{ statusText(row.status) }}</td>
          <td class="center-text">
            <a href="#" @click="edit(row)">Edit</a>
            <a href="#" @click="hapus(index, row.id)">Hapus</a>
          </td>
        </tr>
      </tbody>
    </table>
```

```
    </div>
    <script src="assets/js/app.js"></script>
</body>
</html>
```

### File apps.js

```
const { createApp } = Vue

// tentukan lokasi API REST End Point
const apiUrl = 'http://localhost/labci4/public'

createApp({
  data() {
    return {
      artikel: ''
    }
  },
  mounted() {
    this.loadData()
  },
  methods: {
    loadData() {
      axios.get(apiUrl + '/post')
        .then(response => {
          this.artikel = response.data.artikel
        })
        .catch(error => console.log(error))
    },
    statusText(status) {
      if (!status) return ''
      return status == 1 ? 'Publish' : 'Draft'
    }
  },
}).mount('#app')
```

Hasil outputnya.

localhost/labvue/#

## Daftar Artikel

ID	Judul	Status	Aksi
11	ter judul penelitian baru	Publish	<a href="#">Edit</a> <a href="#">Hapus</a>
10	usulan penelitian yang lain	Draft	<a href="#">Edit</a> <a href="#">Hapus</a>
8	usulan penelitian yang lain	Draft	<a href="#">Edit</a> <a href="#">Hapus</a>
7	artikel 5	Draft	<a href="#">Edit</a> <a href="#">Hapus</a>
6	Artikel tigaaaaa	Publish	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Artikel satuuuu lagi	Publish	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Artikel 1	Publish	<a href="#">Edit</a> <a href="#">Hapus</a>

## Form Tambah dan Ubah Data

Pada file **index.html** sisipkan kode berikut sebelum table data.

```
<button id="btn-tambah" @click="tambah">Tambah Data</button>
<div class="modal" v-if="showForm">
  <div class="modal-content">
    <span class="close" @click="showForm = false">&times;</span>
    <form id="form-data" @submit.prevent="saveData">
      <h3 id="form-title">{{ formTitle }}</h3>
      <div><input type="text" name="judul" id="judul" v-
model="formData.judul" placeholder="Judul" required></div>
      <div><textarea name="isi" id="isi" rows="10" v-
model="formData.isi"></textarea></div>
      <div>
        <select name="status" id="status" v-
model="formData.status">
          <option v-for="option in statusOptions"
:value="option.value">
            {{ option.text }}
          </option>
        </select>
      </div>
      <input type="hidden" id="id" v-model="formData.id">
      <button type="submit" id="btnSimpan">Simpan</button>
      <button @click="showForm = false">Batal</button>
    </form>
  </div>
</div>
```

File app.js lengkapi kodenya.

```
const { createApp } = Vue

// tentukan lokasi API REST End Point
const apiUrl = 'http://localhost/labci4/public'

createApp({
  data() {
    return {
      artikel: '',
      formData: {
        id: null,
        judul: '',
        isi: '',
        status: 0
      },
      showForm: false,
      formTitle: 'Tambah Data',
      formTitles: [{text: 'Tambah Datas'}],
      statusOptions: [
        {text: 'Draft', value: 0},
        {text: 'Publish', value: 1},
      ],
    }
  },
  mounted() {
    this.loadData()
  },
  methods: {
    loadData() {
      axios.get(apiUrl + '/post')
        .then(response => {
          this.artikel = response.data.artikel
        })
        .catch(error => console.log(error))
    },
    tambah() {
      this.showForm = true
      this.formTitle = 'Tambah Data'
      this.formData = {
        id: null,
        judul: '',
        isi: '',
        status: 0
      }
    },
    hapus(index, id) {
```

```

        if (confirm('Yakin menghapus data?')) {
            axios.delete(apiUrl + '/post/' + id)
                .then(response => {
                    this.artikel.splice(index, 1)
                })
                .catch(error => console.log(error))
        }
    },
    edit(data) {
        this.showForm = true;
        this.formTitle = 'Ubah Data'
        this.formData = {
            id: data.id,
            judul: data.judul,
            isi: data.isi,
            status: data.status
        };
        console.log(data)
        console.log(this.formData)
    },
    saveData() {
        if (this.formData.id) {
            axios.put(apiUrl + '/post/' + this.formData.id, this.formData)
                .then(response => {
                    this.loadData()
                })
                .catch(error => console.log(error))
            console.log('Update item:', this.formData);
        } else {
            axios.post(apiUrl + '/post', this.formData)
                .then(response => {
                    this.loadData()
                })
                .catch(error => console.log(error))
            console.log('Tambah item:', this.formData);
        }

        // Reset form data
        this.formData = {
            id: null,
            judul: '',
            isi: '',
            status: 0
        };

        this.showForm = false;
    },
    statusText(status) {
        if (!status) return ''
    }
}

```

```
        return status == 1 ? 'Publish' : 'Draft'
    }
},
}).mount('#app')
```

File style.css

```
#app {
    margin: 0 auto;
    width: 900px;
}
table {
    min-width: 700px;
    width: 100%;
}
th {
    padding: 10px;
    background: #5778ff !important;
    color: #ffffff;
}
tr td {
    border-bottom: 1px solid #eff1ff;
}
tr:nth-child(odd) {
    background-color: #eff1ff;
}
td {
    padding: 10px;
}
.center-text {
    text-align: center;
}
td a {
    margin: 5px;
}
#form-data {
    width: 600px;
}
form input {
```

```
    width: 100%;
    margin-bottom: 5px;
    padding: 5px;
    box-sizing: border-box;
}

form select {
    margin-bottom: 5px;
    padding: 5px;
    box-sizing: border-box;
}

form textarea {
    width: 100%;
    margin-bottom: 5px;
    padding: 5px;
    box-sizing: border-box;
}

form div {
    margin-bottom: 5px;
    position: relative;
}

form button {
    padding: 10px 20px;
    margin-top: 10px;
    margin-bottom: 10px;
    margin-right: 10px;
    cursor: pointer;
}

#btn-tambah {
    margin-bottom: 15px;
    padding: 10px 20px;
    cursor: pointer;
    background-color: #3152d6;
    color: #ffffff;
    border: 1px solid #3152d6;
}

#btnSimpan {
    background-color: #3152d6;
    color: #ffffff;
    border: 1px solid #3152d6;
}

.modal {
```

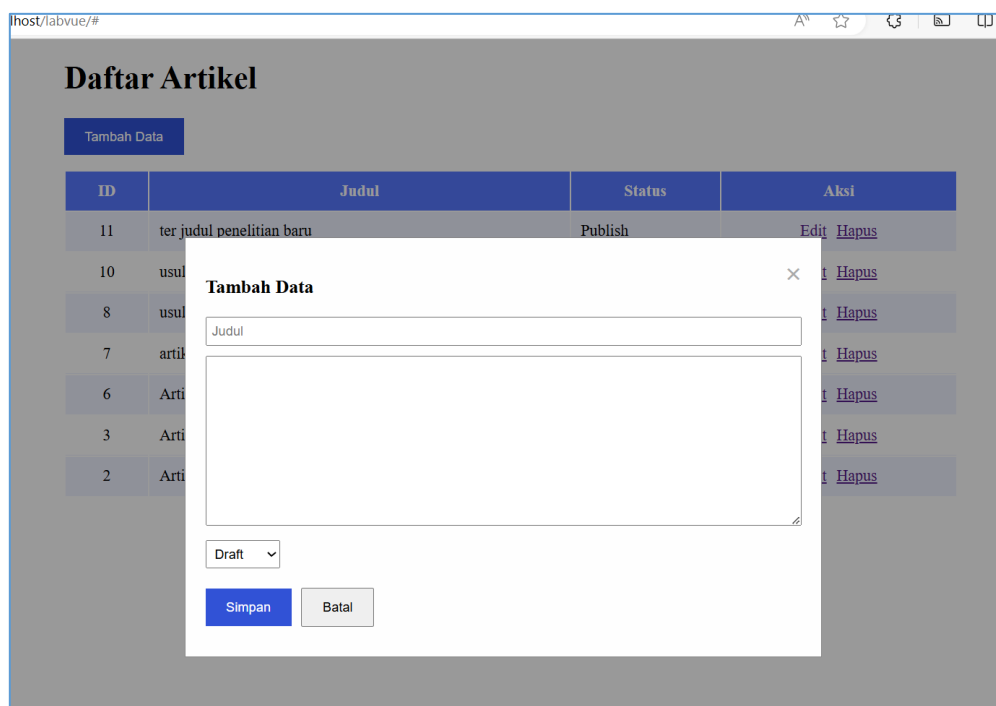


```
display: block;
position: fixed;
z-index: 1;
left: 0;
top: 0;
width: 100%;
height: 100%;
overflow: auto;
background-color: rgba(0, 0, 0, 0.4);
}

.modal-content {
background-color: #fefefe;
margin: 15% auto;
padding: 20px;
border: 1px solid #888;
width: 600px;
}

.close {
color: #aaa;
float: right;
font-size: 28px;
font-weight: bold;
cursor: pointer;
}
```

Hasil outputnya.



### Pertanyaan dan Tugas

Selesaikan programnya sesuai Langkah-langkah yang ada. Anda boleh melakukan improvisasi.

### Laporan Praktikum

1. Melanjutkan praktikum sebelumnya pada **repository** dengan nama **Lab11Web\_VueJS**.
2. Kerjakan semua latihan yang diberikan sesuai urutannya.
3. Screenshot setiap perubahannya.
4. Update file **README.md** dan tuliskan penjelasan dari setiap langkah praktikum beserta screenshotnya.
5. **Commit** hasilnya pada **repository** masing-masing.
6. Kirim **URL repository** pada **e-learning** ecampus