

Halo rekan-rekan mahasiswa, bagaimana kabar kalian? Semoga rekan-rekan mahasiswa dalam keadaan baik. Nah, kali ini ada Tugas Praktik 3 untuk mengimplementasikan apa saja yang sudah rekan-rekan mahasiswa pelajari mengenai **Vue.js**.

Secara garis besar, tujuan dari tugas praktikum ini adalah:

Mahasiswa mampu mengimplementasikan sistem perorganisasian kode Javascript Framework **Vue.js** dengan konsep **Vue Component** dan Template untuk membangun antarmuka dan alur interaktif dari aplikasi pemesanan bahan ajar *SITTA* untuk sistem kebutuhan proses pemesanan dan distribusi bahan ajar di UT.

Pertama, Tugas Praktik 3 ini memiliki delapan Indikator Capaian untuk menilai Hasil Belajar Mahasiswa, yaitu:

1. Mahasiswa dapat menerapkan sistem perorganisasian kode *Javascript Framework Vue.js* dengan memanfaatkan konsep **Vue Component** dan **Vue Template**
2. Mahasiswa dapat menerapkan fungsi untuk menampilkan data (*mustaches/directive v-text/directive v-text*) untuk menghasilkan output yang diharapkan.
3. Mahasiswa dapat menerapkan fungsi *conditional* atau pengandaian (*directive v-if/directive v-else/directive v-else-if/directive v-show*) dengan menerapkan *operator condition* yang sesuai untuk menghasilkan output yang diharapkan.
4. Mahasiswa mampu menerapkan data binding (*one-way data binding/two-way data binding*) melalui jenis *directive data binding* (*v-bind,v-model*) dan penggunaan *property* untuk mengolah data (*computed property/methods property*) yang sesuai untuk menghasilkan output yang diharapkan
5. Mahasiswa mampu menerapkan fungsi *watcher* yang sesuai untuk menghasilkan output yang diharapkan
6. Mahasiswa mampu menerapkan pengolahan data *Array* melalui *directive v-for*, baik untuk *zero-based number index* maupun *name based index*
7. Mahasiswa mampu menerapkan *formatting* data teks menggunakan konsep filter
8. Mahasiswa mampu menerapkan penggunaan *custom element*, **Vue Component**, dan **Property Template**

Soal Tugas

Anda diminta untuk membuat sebuah aplikasi website sederhana untuk pemesanan Bahan Ajar di Universitas Terbuka. Pada Tugas Praktik Kedua ini, mahasiswa berfokus untuk membuat aplikasi *website* berbasis **Vue.js** sebagai *framework* yang digunakan sepanjang penggerjaan Tugas Praktik 3 Sehingga, data-data yang dibutuhkan tidak dibaca melalui *database*, tapi melalui json file sebagai *dummy data* yang akan dilampirkan bersamaan dengan Tugas Praktik ini.

Pada Tugas Praktik 2, kita sudah membuat beberapa fitur pada sistem Bahan Ajar Universitas Terbuka menggunakan *Vue.js* yang masih sederhana. Sekarang, tugas teman-teman adalah merefaktor halaman web yang sudah dibuat menjadi custom element dalam

bentuk **Vue Component** dan **Template**. Namun, tentunya ada beberapa hal yang perlu ditambahkan untuk mencapai indikator untuk penilaian hasil belajar teman-teman.

Untuk struktur proyek **Vue** pada Tugas Praktik 3 adalah sebagai berikut:

tugas3-vue-ut/

```
|── index.html          # Root: mount #app  
|  
|── /assets/  
|   |── css/  
|   |   |── style.css    # Gaya global  
|   |── img/           # Ikon/gambar (opsional)  
|  
|── /data/  
|   |── dataBahanAjar.json  # Sumber data (JSON)  
|  
|── /js/  
|   |── app.js         # Inisialisasi Vue root  
|   |── components/  
|   |   |── stock-table.js  # <ba-stock-table> (ini hanya contoh saja)  
|   |   |── do-tracking.js # <do-tracking> (ini hanya contoh saja)  
|   |   |── order-form.js  # <order-form> (ini hanya contoh saja)  
|   |   |── status-badge.js # <status-badge> (ini hanya contoh saja)  
|   |   |── app-modal.js   # <app-modal> (ini hanya contoh saja)  
|   |── services/  
|   |   |── api.js        # fetch JSON (data service), menangani data akses  
└── /templates/  
    |── stock-table.html # (ini hanya contoh saja)  
    |── do-tracking.html # (ini hanya contoh saja)  
    |── order-form.html # (ini hanya contoh saja)  
    |── status-badge.html # (ini hanya contoh saja)
```

└─ app-modal.html # (ini hanya contoh saja)

Catatan:

Nama *file* untuk nama komponen menggunakan gaya *kebab-case* untuk *naming convention style* yang digunakan dalam tugas ini.

Contoh:

File component `stock-table.js` diregistrasikan menjadi *Vue Component* dengan perintah sebagai berikut: `Vue.component('ba-stock-table', ...)`

Setelah itu, id template juga perlu konsisten. Sebagai contoh: `tpl-stock`, `tpl-tracking`, dsb., agar mudah dicari.

Routing ke halaman lain cukup tab state di root (`index.html`)

Misal: `tab: 'stok' | 'tracking' | 'order'`

Sebagai contoh, berikut adalah cuplikan ringkas pada **index.html**

```
<body>
  <div id="app">
    <nav>
      <button @click="tab='stok'">Stok</button>
      <button @click="tab='tracking'">Tracking D0</button>
      <button @click="tab='order'">Pemesanan</button>
    </nav>

    <section v-show="tab==='stok'">
      <ba-stock-table :items="state.stok"></ba-stock-table>
    </section>
    <section v-show="tab==='tracking'">
      <do-tracking :data="state.tracking"></do-tracking>
    </section>
    <section v-show="tab==='order'">
      <order-form :paket="state.paket" :ekspedisi="state.pengirimanList"
        @created="handleNewD0"></order-form>
    </section>

    <app-modal ref="modal"></app-modal>
  </div>

  <!-- Templates -->
  <template id="tpl-stock"> ... </template>
  <template id="tpl-tracking"> ... </template>
  <template id="tpl-order"> ... </template>
  <template id="tpl-badge"> ... </template>
  <template id="tpl-modal"> ... </template>

  <!-- Scripts (urutan penting) -->
  <script src="https://cdn.jsdelivr.net/npm/vue@2/dist/vue.js"></script>
  <script src="/js/services/api.js"></script>
  <script src="/js/components/stock-table.js"></script>
  <script src="/js/components/do-tracking.js"></script>
  <script src="/js/components/order-form.js"></script>
  <script src="/js/components/status-badge.js"></script>
  <script src="/js/components/app-modal.js"></script>
  <script src="/js/app.js"></script>
</body>
```

1. Halaman Stok Bahan Ajar

- Halaman ini menampilkan daftar stok bahan ajar untuk seluruh UT daerah (silakan bebas menggunakan *table/list/bentuk lain* yang menampilkan *list* data dalam jumlah banyak dan memiliki data-data field). Teman-teman mahasiswa perlu menampilkan daftar stok bahan ajar yang terdiri dari kolom data field sebagai berikut
 - Kode Mata Kuliah/Nama Mata Kuliah → `kode, judul`
 - Kategori Mata Kuliah → `Kategori`
 - UT-Daerah → `upbjj`
 - Lokasi Rak → `lokasiRak`
 - Harga → `harga`
 - Buatlah *formatting* teks harga dengan menambahkan currency (Rp)
 - Jumlah Stok Bahan Ajar → `qty`
 - Buatlah *formatting* teks Jumlah Stok Bahan Ajar dengan menambahkan satuan (buah)
 - Jumlah Stok Safety Bahan Ajar → `safety`
 - Buatlah *formatting* teks Jumlah Stok Bahan Ajar dengan menambahkan satuan (buah)
 - Status

Untuk *data field* Catatan → `catatanHTML`, ditampilkan saat mentrigger *event mouse hover* pada kolom ‘Status’ dan menampilkan *tooltip/preview*

- Terdapat Fitur untuk mengedit stok data bahan ajar
- Terdapat fitur untuk melakukan *filter* dan *sort* data stock (silakan bebas menggunakan elemen *select/radio/checkbox*)
 - Terdapat *filter* stok berdasarkan:
 - ‘UT-Daerah’: terdiri pilihan dari seluruh UT-daerah, dan daftar setiap UT-daerah (`upbjjList`)
 - ‘Kategori Mata Kuliah’ (`upbjjList`)
 - Terdapat *filter* yang hanya menampilkan jumlah stock (`qty`) < dari jumlah *safety* stock (*safety*) dan jumlah stock = 0 untuk menjadi daftar yang dapat mengingatkan untuk melakukan *re-order* bahan ajar.
 - Terdapat fitur *sort* yang mengurutkan berdasarkan judul, stock, dan harga
 - Fitur filter dapat di-set ulang (*reset*)
 - Terapkan fitur di vue.js di mana di semua filter tidak perlu *recompute* kembali.
 - Implementasikan *dependent options* seperti misalkan ketika memilih filter stok berdasarkan ‘UT-daerah’, baru akan memunculkan filter untuk ‘Kategori Mata Kuliah’
- Menampilkan status dari stok bahan ajar
 - Memberikan status ‘Aman dengan menggunakan simbol yang memberikan tanda yang memberikan kesan ‘aman’ ataupun teks berwarna hijau apabila stok \geq safety stock

- Memberikan status ‘Menipis’ dengan menggunakan simbol yang memberikan kesan adanya ‘warning’ ataupun teks berwarna orange apabila stok < safety stock.
 - Memberikan status ‘Kosong’ dengan menggunakan simbol yang memberikan kesan adanya ‘bahaya’ ataupun teks berwarna merah apabila stok = 0
- Terdapat Fitur untuk menambahkan data bahan ajar baru (*create*) melalui formulir kecil dan melakukan validasi sederhana untuk setiap data field.
 - Ketika mentrigger *event keyboard* ‘Enter’, maka data bahan ajar baru langsung disimpan dan diperbarui pada tampilan daftar
- Terdapat Fitur untuk memperbarui (*update*) data bahan ajar baru melalui formulir kecil dan melakukan validasi sederhana untuk setiap data field.
 - Ketika mentrigger *event keyboard* ‘Enter’, maka data bahan ajar akan langsung ter-*update* dengan nilai terbaru
- Terdapat Fitur untuk menghapus data bahan ajar melalui event click tombol/icon “delete”
 - Sebelum menghapus data bahan ajar, terdapat pop up yang mengkonfirmasi penghapusan data kepada pengguna

2. Tracking *Delivery Order* (DO)

- Halaman ini menampilkan halaman untuk tracking status pengiriman DO bahan ajar seperti pada Tugas Praktik 1, tetapi di sini perlu menerapkan penggunaan framework **Vue.js**.
 - Menampilkan fitur pencarian berdasarkan:
 - Nomor DO, atau
 - NIM
 - Pengguna dapat melakukan submit pencarian dengan cara menekan tombol *keyboard* ‘Enter’
 - Pengguna dapat melakukan *clear/reset* pencarian dengan cara menekan tombol *keyboard* ‘Esc’
- Terdapat fitur untuk menambahkan *input delivery order* yang baru. Berikut adalah data field yang perlu diisi.
 - Nomor DO, di mana tergenerate otomatis. Penomorannya terdiri dari:
 - DO
 - Tahun Berjalan
 - Sequence Number

Misal:

DO2025-001 untuk data awal

Selanjutnya jika sudah ada data tracking terbaru, menjadi DO2025-002 dst.

- NIM, diisi secara manual oleh pengguna
- Nama, diisi secara manual oleh pengguna
- Ekspedisi (JNE Regular/JNE Express), dipilih oleh pengguna berdasarkan data pada **upbjjList**

- Paket Bahan Ajar, dipilih oleh pengguna berdasarkan data pada **paket**
 - Gunakan elemen `<select>` dan menampilkan di halaman web informasi kode paket dan nama paket
 - Setelah memilih paket, akan muncul detail isi paket bahan ajar tersebut di bawah elemen `<select>` untuk memberikan informasi
- Tanggal Kirim, dapat diisi manual ataupun menggunakan fungsi **Date** untuk mengambil *local time*. Pada tanggal kirim, perlu diformat sebagai berikut: tanggal bulan tahun. Contoh: 25 Agustus 2025
- Total Harga, diambil dari data pada *Array of Objects* **paket** → **harga**. Pada tanggal kirim, perlu diformat dengan menambahkan *currency* (Rp)

Buatlah juga validasi input yang sederhana saat memasukkan data *delivery order* baru.

- Terdapat fitur untuk menambah status *progress* perjalanan pengiriman
 - Data “waktu” diambil berdasarkan *local time* menggunakan **Date**.
 - Data “keterangan” diambil berdasarkan *input* pengguna.

Petunjuk Penggeraan dan Poin untuk Video Penjelasan Tugas Praktik 3

1. Kriteria penilaian dalam tugas ini adalah bagaimana teman-teman mengimplementasikan:
 - 1.1. Arsitektur dan Struktur Proyek Vue.js: struktur folder, struktur penamaan file, struktur komponen, menerapkan pemakaian *vue component* dan *template* (**20 Poin**)
 - 1.2. Penggunaan *Data Binding & Directive*, *Array*, dan *Filter* sebagai *formatting data teks* untuk *List Rendering*: penggunaan fungsi menampilkan data (*mustaches*, *v-text*, *v-html*) dan fungsi *data binding* (*v-bind*, *v-model*) yang tepat dan sesuai dengan *output* yang diharapkan untuk membuat daftar, menerapkan *v-for* untuk pengolahan data array (**10 Poin**)
 - 1.3. Penggunaan *Conditional*: fungsi *conditional* (*v-if*, *v-else*, *v-else-if*, *v-show*) yang efektif dan sesuai dengan *output* yang diharapkan (**7 Poin**)
 - 1.4. Penggunaan *Property (Computed* ataupun *Methods*): menerapkan dengan sesuai (**10 Poin**)
 - 1.5. *Watchers*: minimal menggunakan 2 *watcher* untuk memantau perubahan data (**10 Poin**)
 - 1.6. Formulir input dan validasi input sederhana, serta penggunaan *event handler* melalui *mouse* dan *keyboard*: menampilkan element-element yang berinteraksi dengan pengguna atau yang perlu diisi pengguna yang tepat, sesuai, dan memudahkan pengguna, menerapkan *event handler* melalui *mouse* dan *keyboard* sebagai pengalaman interaksi yang lebih mudah untuk pengguna (**20 Poin**)
 - 1.7. Kreativitas dalam menampilkan interface yang mudah bagi pengguna dan memberikan pengalaman penggunaan yang nyaman (**8 Poin**)
 - 1.8. Penjelasan dalam **video** untuk menyelesaikan tugas dengan sistematika dan alur berpikir yang dapat menjelaskan penerapan fungsi-fungsi (beserta argumentasi pendukung) yang digunakan dalam merancang dan membuat aplikasi website menggunakan Vue.js (**15 Poin**)
2. Durasi video penjelasan maksimal 15 menit

Setelah menyelesaikan Tugas Praktik 3, rekan-rekan dapat mengirimkan file source code dan link rekaman video penjelasan penggeraan Tugas Praktik 3.

Berikut adalah sumber-sumber yang dapat membantu teman-teman dalam mengerjakan tugas ini:

1. https://www.w3schools.com/howto/howto_css_modals.asp
2. https://www.w3schools.com/howto/howto_js_popup_form.asp
3. https://www.w3schools.com/js/js_popup.asp
5. https://www.w3schools.com/js/js_date_methods.asp

Sumber Referensi:

Sufandi, U. U., Aprijani, D. A., & Pandiangan, P. (2021). Evaluasi dan hasil review desain user interface prototype aplikasi mobile SITTA Universitas Terbuka. *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, 10(3), 147-156. <https://doi.org/10.23887/janapati.v10i3.40281>

Sufandi, U. U. (2022). Analisis Kebutuhan dan Dokumentasi Sistem Informasi Tiras dan Transaksi Bahan Ajar Universitas Terbuka. *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, 11(2), 112-122. <https://doi.org/10.23887/janapati.v11i2.42966>