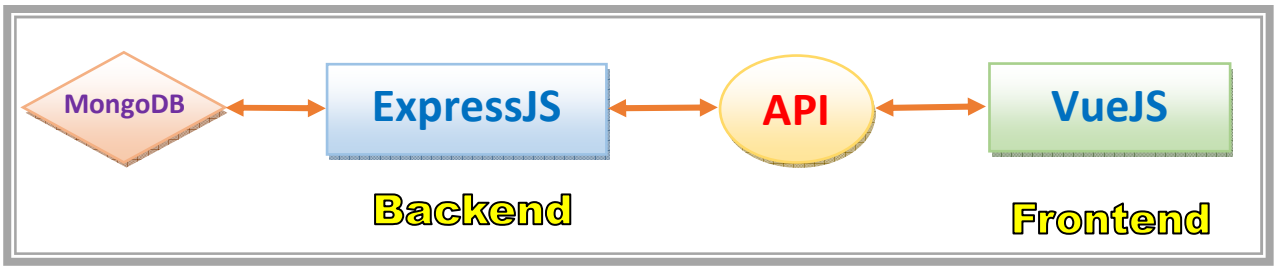
LAB 4

❖ CONTENT

- Enable Restful API exchange in backend with "cors" package
- Initialize new VueJS project as frontend
- Consume Restful APIs from backend with "axios" package

❖ INTRODUCTION

- CORS: Cross-Origin Resource Sharing. It is a security feature implemented in web browsers to control how web applications running at one origin can interact with resources from a different origin.
- VueJS is a progressive JavaScript framework used for building user interfaces and single-page applications. Vue.js is known for its simplicity, flexibility, and fine-grained reactivity.
- Some alternatives to VueS: ReactJS, AngularJS, SvelteJS
- Some ways to consume Restful API with VueJS:
  - *Axios: a promise-based HTTP client*
  - Fetch API: a browser built-in web API
- System architecture diagram:



❖ INSTRUCTION

1. Continue with previous project and set it as backend project

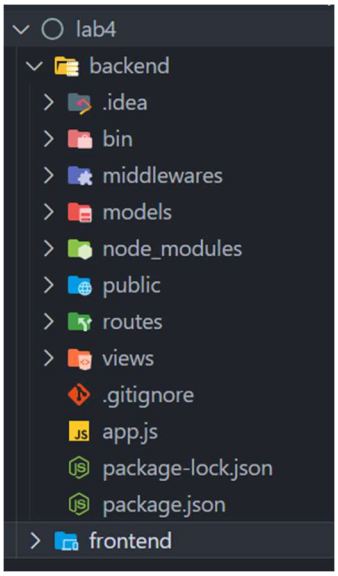


Figure 1 - Project structure

2. Install new package: CORS

```
npm install cors
```

Figure 2 - Install new package

3. Enable CORS in backend side for Restful API exchange

```
//import "cors" library
var cors = require('cors');

//usage 1: enable CORS requests for all domains
app.use(cors());

//usage 2: enable CORS requests for a single route
app.get('/product', cors(), (req, res) => {
  //codes go here
})

var corsOptions = {
  origin: 'https://mywebsite.com.vn',
  optionsSuccessStatus: 200
}

//usage 3: enable CORS requests for a single domain
app.get('/product', cors(corsOptions), (req, res) => {
  //codes go here
})
```

*Figure 3 - Config CORS in backend (select 1 method only) (**app.js**)*

4. Start the backend server and keep it running. Do not stop it.

   Note: Default web server address for **ExpressJS** is **http://localhost:3000**

```
\lab4\backend>npm start
```

Figure 4 - Start the backend server

5. Initialize the frontend side with VueJS.

   Install **axios** package to consume API from backend.

   Start VueJS web server: **npm run dev**

   Note: Default web server address for **VueJS** is **http://localhost:5173**

```
npm create vue@latest frontend
cd frontend
npm install axios
npm run dev
```

Figure 5 - Initialize VueJS project in new Terminal

```
√ Add TypeScript? ... No / Yes
√ Add JSX Support? ... No / Yes
√ Add Vue Router for Single Page Application development? ... No / Yes
√ Add Pinia for state management? ... No / Yes
√ Add Vitest for Unit Testing? ... No / Yes
√ Add an End-to-End Testing Solution? » No
√ Add ESLint for code quality? ... No / Yes
```

Figure 6 - Config VueJS project

6. Update the default view page **src/App.vue** to fetch data from backend

```
<script setup>
//import "axios" library to consume API from backend
import axios from "axios";
</script>
```

*Figure 7 - import "**axios**" library*

```
<script>
var backendAPI = "http://localhost:3000/api/product";
export default {
  data() {
    return {
      data: null,
    };
  },
  mounted() {
    axios
      .get(backendAPI)
      .then((response) => {
        this.data = response.data;
      })
      .catch((error) => {
        console.log(error);
      });
  },
};
</script>
```

*Figure 8 - config **backendAPI url** and fetch data **with axios.get()***

```
<template>
  <div>
    <table>
      <thead>
        <tr>
          <th colspan="4">Product List</th>
        </tr>
        <tr>
          <th>Product name</th>
          <th>Product price</th>
          <th>Product image</th>
          <th>Product category</th>
          <th>Menu</th>
        </tr>
      </thead>
      <tbody>
        <tr v-for="product in data" :key="product._id">
          <td>{{ product.name }}</td>
          <td>${{ product.price }}</td>
          <td>
            <img :src="product.image" width="100" height="100" />
          </td>
          <td>{{ product.category.name }}</td>
        </tr>
      </tbody>
    </table>
  </div>
</template>
```

Figure 9 - Load fetched data to table

```
  <title>Product Management System</title>
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js"></script>
</head>
```

*Figure 10 - Import Materialize CSS framework to file s**rc/index.html***

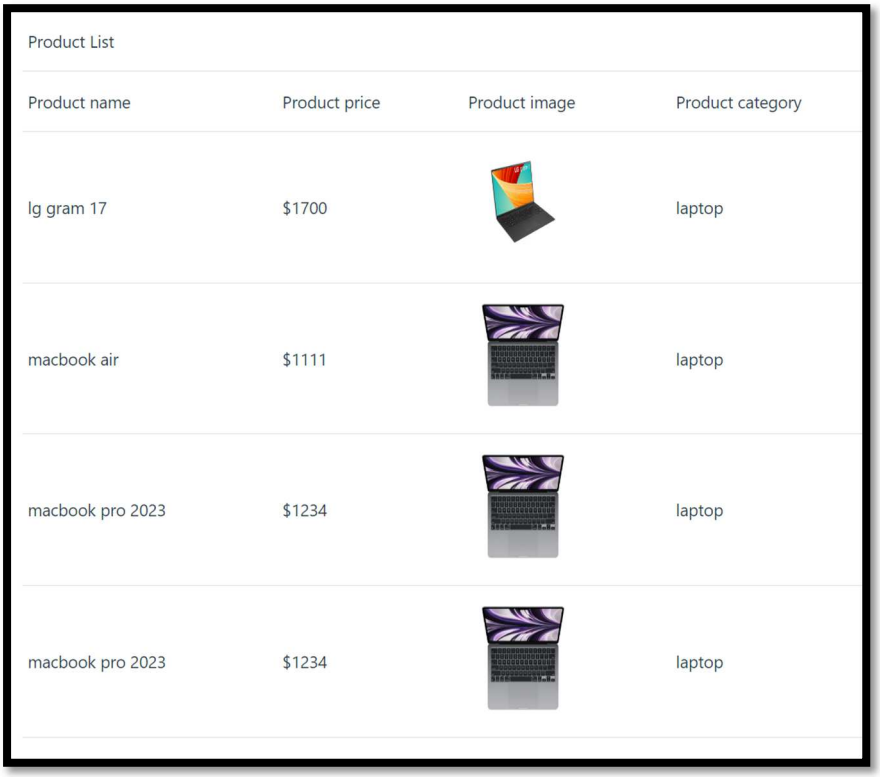


Figure 11 - Result of fetching data from backend with axios

- **<u>TODO:</u>** Insert the "**Add**", "**Edit**", "**Delete**" button in table then consume other Restful APIs from backend side