

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Nắm vững cách sử dụng Vue Router để điều hướng giữa các component trong ứng dụng Vue.
- ✓ Hiểu và áp dụng các phương thức xác thực (token-based) trong Vue.js.

NỘI DUNG

PHẦN I:

Bài 1 (2 điểm): Thực hành router cơ bản

Yêu cầu:

1. Tạo một ứng dụng Vue cơ bản với Vue Router.
2. Thiết lập các route cho hai trang: **Home** và **About**.
3. Sử dụng **router-link** để điều hướng giữa các trang.

Hướng dẫn

Bước 1: Tạo component **Labs/Lab-7/Bai-1/Home.vue**

```
<!-- Home.vue -->
<template>
<div class="container">
<h1>Trang chủ</h1>
<p>Chào mừng bạn đến với trang chủ của ứng dụng!</p>
</div>
</template>
```

Bước 2: Tạo component **Labs/Lab-7/Bai-1/About.vue**

```
<!-- About.vue -->
<template>
<div class="container">
<h1>Giới thiệu</h1>
<p>Đây là trang giới thiệu về ứng dụng.</p>
</div>
</template>
```

Bước 3: Nhúng **Home.vue** và **About.vue** vào **App.vue**

```
<!-- App.vue -->
<template>
<div>
<nav class="navbar navbar-expand-lg navbar-light bg-light">
    <router-link to="/" class="navbar-brand">Trang chủ</router-link>
```

```
<div class="collapse navbar-collapse">
<ul class="navbar-nav">
    <li class="nav-item"><router-link to="/" class="nav-link">Home</router-link></li>
    <li class="nav-item"><router-link to="/about" class="nav-link">About</router-link></li>
</ul>
</div>
</nav>
<router-view />
</div>
</template>

<script setup>
import { createRouter, createWebHistory } from "vue-router";
import Home from "./components/Home.vue";
import About from "./components/About.vue";
const routes = [
    { path: "/", component: Home },
    { path: "/about", component: About },
];
const router = createRouter({
    history: createWebHistory(),
    routes,
});
export default router;
</script>
```

Bài 2: Dynamic Route Matching với Params

Yêu cầu:

- Tạo một trang chi tiết sản phẩm ProductDetail.vue sử dụng dynamic route matching.
- Sử dụng params để hiển thị thông tin chi tiết sản phẩm dựa trên ID sản phẩm.

Hướng dẫn

Bước 1: Tạo component **Labs/Lab-7/Bai-2/ProductDetail.vue**. tham khảo code sau:

```
<template>
<div class="container">
<h2>Chi tiết sản phẩm</h2>
<div v-if="product">
    <h3>{{ product.name }}</h3>
    <p>Giá: {{ product.price }} VND</p>
    <p>Mô tả: {{ product.description }}</p>
    <p>Thể loại: {{ product.category }}</p>
    <p>Số lượng: {{ product.quantity }}</p>
    
</div>
<div v-else>
    <p>Đang tải dữ liệu sản phẩm...</p>
</div>
```

```
</div>
</template>

<script setup>
import { ref, onMounted } from "vue";
import { useRoute } from "vue-router";
import axios from "axios";
const product = ref(null);
const route = useRoute();
const fetchProduct = async () => {
const response = await axios.get(`http://localhost:3000/products/${route.params.id}`);
product.value = response.data;
};
onMounted(() => {
    fetchProduct();
});
</script>
```

Bước 2: cập nhật App.vue, cho thêm route ProductDetail

```
<script setup>
import ProductDetail from "./ProductDetail.vue";
const routes = [
    { path: "/", component: Home },
    { path: "/about", component: About },
    { path: "/product/:id", component: ProductDetail }, // Dynamic route
];
</script>
```

PHẦN II:

Bài 3 (2 điểm): Xác thực với Token Based

Yêu cầu:

1. Tạo một trang Login.vue cho phép người dùng đăng nhập với xác thực token-based.
2. Sử dụng axios để gửi thông tin đăng nhập và lưu trữ token khi đăng nhập thành công.

Hướng dẫn:

Bước 1: Cài đặt json-server-auth: [npm i json-server@0.17.4 json-server-auth -D](#)

Bước 2: Cấu hình lại package.json:

```
"scripts": {
    "server": "json-server-auth db.json --watch",
    "dev": "vite",
    "build": "vite build",
    "preview": "vite preview"
```

Bước 3: Tạo component Labs/Lab-7/Bai3.vue

```
<!-- Login.vue -->
<template>
<div class="container">
<h2>Đăng nhập</h2>
<form @submit.prevent="login">
<div class="mb-3">
    <input v-model="username" placeholder="Tên đăng nhập" class="form-control" />
</div>
<div class="mb-3">
    <input type="email" v-model="email" placeholder="Email" class="form-control" />
</div>
<div class="mb-3">
<input
    type="password"
    v-model="password"
    placeholder="Mật khẩu"
    class="form-control"
/>
</div>
<button type="submit" class="btn btn-primary">Đăng nhập</button>
</form>
<p v-if="error" class="text-danger">{{ error }}</p>
</div>
</template>
<script setup>
import { ref } from "vue";
import axios from "axios";
import { useRouter } from "vue-router";
const username = ref("");
const email = ref("");
const password = ref("");
const error = ref("");
const router = useRouter();
const login = async () => {
    try {
        const response = await axios.post("http://localhost:3000/login", {
            username: username.value,
            email: email.value,
            password: password.value,
        });
        localStorage.setItem("token", response.data.token);
        router.push("/");
    } catch (err) {
        error.value = "Đăng nhập không thành công";
    }
};
</script>
```

Bước 2: Nhúng Login.vue ở App.vue

```
<!-- App.vue -->

<template>
<div class="container mt-4">
    <Bai3/>
</div>
</template>

<script setup>

import Bai3 from "./Labs/Lab-7/Bai3.vue";
</script>
```

Bài 4 (3 điểm): Kiểm tra xác thực khi người dùng truy cập
Yêu cầu:

1. Tạo một trang Dashboard.vue chỉ hiển thị cho người dùng đã đăng nhập.
2. Kiểm tra xác thực khi người dùng truy cập vào trang này. Nếu chưa đăng nhập, chuyển hướng về trang đăng nhập.

Hướng dẫn:
Bước 1: Tạo component Labs/Lab-7/Bai-4/Dashboard.vue

```
<!-- Dashboard.vue -->
<template>
<div class="container">
<h2>Trang Dashboard</h2>
<p>Chào mừng đến với trang Dashboard!</p>
</div>
</template>

<script setup>
import { ref } from "vue";
const user = ref(JSON.parse(localStorage.getItem("user")));
if (!user.value) {
    // Nếu không có thông tin người dùng, chuyển hướng về trang đăng nhập
    window.location.href = "/login";
}
</script>
```

Bước 2: Truy cập App.vue cập nhật code sau

```
<!-- App.vue -->
<script setup>
import { createRouter, createWebHistory } from "vue-router";
import Home from "./components/Home.vue";
import About from "./components/About.vue";
import Login from "./components/Login.vue";
import Dashboard from "./components/Dashboard.vue";

const routes = [
    { path: "/", component: Home },
    { path: "/about", component: About },
    { path: "/login", component: Login },
    { path: "/dashboard", component: Dashboard, meta: { requiresAuth: true } },
];
const router = createRouter({
    history: createWebHistory(),
    routes,
});
// Navigation guard
router.beforeEach((to, from, next) => {
    if (to.matched.some((record) => record.meta.requiresAuth) && !localStorage.getItem("token")) {
        next({ path: "/login" });
    } else {
        next();
    }
});
</script>
<template>
<router-view></router-view>
</template>
```

**** Giảng viên có thể cho các bài tập phù hợp phục vụ assignment.**

***** Yêu cầu nộp bài:**

SV nén file (*hoặc share thư mục google drive*) bao gồm các yêu cầu đã thực hiện

trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM

--- Hết ---