**LAB 17 Express-handlebars**

**17.1. Server-side Template Engine là gì?**

* Tạo **HTML động từ server** bằng cách kết hợp:
  + **Template**: HTML + placeholder ({{name}}, {{#if}}, {{#each}})
  + **Dữ liệu**: lấy từ DB, API hoặc xử lý logic
  + **Kết xuất (render)**: engine xử lý & chèn dữ liệu
  + **Kết quả**: HTML hoàn chỉnh trả về client

**Ưu điểm**:

* Giao diện động
* Tách biệt logic & giao diện
* Hỗ trợ SEO tốt
* Dễ dùng, tải nhanh lần đầu

**Nhược điểm**:

* Không phù hợp với SPA (phải tải lại trang)

**17.2. Giới thiệu Express-handlebars**

* Là 1 **template engine** tích hợp với Express.
* Dùng cú pháp Mustache: {{variable}}, {{#each}}, {{#if}}

**Cài đặt**:

pnpm i -s express-handlebars

**Kết quả trong package.json**:

"express": "^4.21.2",

"express-handlebars": "^8.0.1"

**17.3. Cách sử dụng với Express**

**Bước 1: Khai báo gói**

const expressHandlebars = require('express-handlebars');

**Bước 2: Cấu trúc thư mục**

TeoShop/

├── views/

│ ├── layouts/ -> main.hbs

│ ├── partials/ -> header.hbs, footer.hbs

│ ├── index.hbs

└── index.js

**Bước 3: Cấu hình**

app.engine('hbs', expressHandlebars.engine({

layoutsDir: \_\_dirname + '/views/layouts',

partialsDir: \_\_dirname + '/views/partials',

extname: 'hbs',

defaultLayout: 'main'

}));

app.set('view engine', 'hbs');

**Bước 4: Tạo main.hbs và index.hbs**

* main.hbs: chứa header, footer, và {{{ body }}}
* index.hbs: nội dung riêng của trang index
* Render:

app.get('/', (req, res) => {

res.render('index');

});

**Sau đó truy cập**: http://localhost:9000/

**17.4. Bài tập ôn lại**

**Câu 17.1**: Express-handlebars là một \_\_\_\_\_\_\_\_\_ được tích hợp vào framework Express để tạo ra “HTML động” từ phía server.

**Đáp án đúng**: **D. server-side template engine**