**EXPRESS 02: TEMPLATE ENGINE EJS**

|  |
| --- |
| * Template Engine EJS * Xử lý dữ liệu trong Form với Module Body Parser * Bóc tách Layout * Xây dựng chức năng Login với Tài khoản cố định |

A – LÝ THUYẾT

I – TEMPLATE ENGINE EJS

**1. Template Engine EJS**

*Template Engine có tác dụng loại bỏ đi phần lớn các đoạn code xử lý phía Server ở trong View. Nó giúp tách biệt hoàn toàn các đoạn mã xử lý logic ra khỏi giao diện HTML, CSS*

*Hiện tại có rất nhiều Template Engine, và EJS là một trong những Template Engine đó.*

**2. Cài đặt**

|  |
| --- |
| npm install ejs --save |

**3. Cấu hình**

|  |
| --- |
| // config ejs template engine  app.set("views", \_\_dirname+"/apps/views");  app.set("view engine", "ejs"); |

II – CÁC CÚ PHÁP SỬ DỤNG TRONG EJS

**1. Sử dụng Template Engine EJS**

*Tất cả các file chỉ được gọi là Template Engine EJS và sử dụng được cú pháp EJS khi nó có thành phần mở rộng (đuôi file) là .ejs*

* Vùng làm việc

|  |
| --- |
| <%  var num = 10;  %> |

* Xuất giá trị

|  |
| --- |
| <%=num%> |

**2. Phương thức Render trong EJS**

|  |
| --- |
| function login(req, res){    res.render('admin/login');  } |

**3. Phương thức Redirect trong EJS**

|  |
| --- |
| function login(req, res){    res.redirect('admin/dashboard');  } |

**4. Hàm Include trong EJS**

|  |
| --- |
| <% include layout/header%> |

III – THAO TÁC VỚI FORM TRONG NODEJS

**1. Phương thức POST của Router**

|  |
| --- |
| router.post('/login', function(req, res){  res.render('admin/login');  }); |

**2. Module Body-Parser**

*Module Body-Parser giúp chúng ta có thể truyền và nhận dữ liệu thông qua Form (Phương thức POST)*

* Cài đặt:

|  |
| --- |
| npm install body-parser --save |

* Cấu hình:

|  |
| --- |
| var body\_parser = require(‘body-parser’);  // Sử dụng Body-Parser với dữ liệu được truyền từ Form  app.use(body\_parser.urlencoded({ extended: true }))    // Sử dụng Body-Parser với dữ liệu dạng Json  app.use(body\_parser.json()) |

* Sử dụng:

|  |
| --- |
| var a = req.body.input\_name; |

**3. Truyền và nhận dữ liệu từ Controller sang View**

Để truyền dữ liệu từ Controller sang View (EJS) chúng ta cần truyền một đối tượng từ Controller, khi sang View các thuộc tính lưu trữ giá trị của đối tượng đó sẽ trở thành biến lưu trữ giá trị tương ứng.

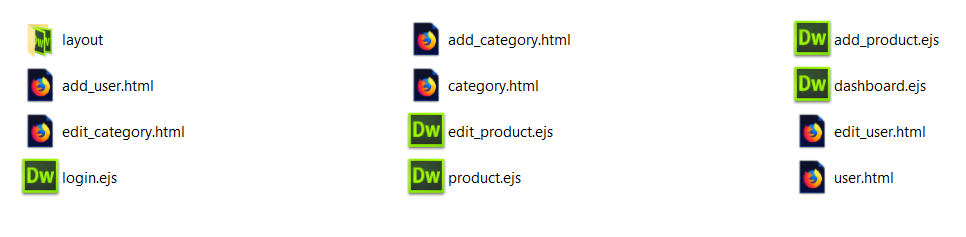
|  |
| --- |
| res.render('admin/login', {data: {error: 'Tài khoản không hợp lệ !'}}); |

B – TRAINING DỰ ÁN

I – CÔNG VIỆC TRÊN LỚP

**1. Bóc tách giao diện**

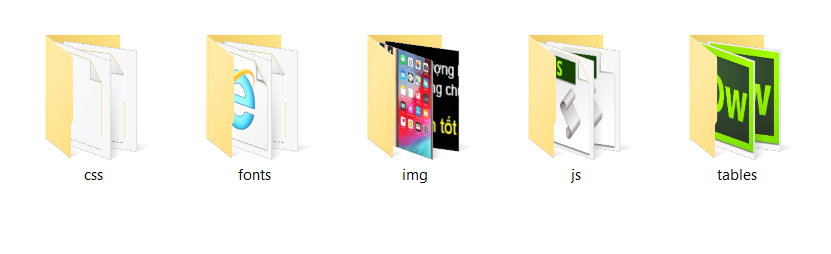
* Thư mục view



* Thư mục layout của view



* Thư mục admin của theme



**2. Xây dựng tính năng Login (đăng nhập)**

* Sử dụng tài khoản cố định với email: [vietpro.edu.vn@gmail.com](mailto:vietpro.edu.vn@gmail.com) – password: 123456
* Đăng nhập thành công thì chuyển hướng tới trang Dashboard

**3. Bóc phần cấu hình các Module**

Tách phần cấu hình các Module cài mới ra file kernal nằm theo đường dẫn apps/kernal

|  |
| --- |
| module.exports = (express, app, bodyParser)=>{  // Config EJS  app.set("views", \_\_dirname+"/views")  app.set("view engine", "ejs")  // Config Static Folder  app.use("/static", express.static(\_\_dirname+"/../public"))  // Config Body Parser  app.use(bodyParser.urlencoded({ extended: true }))  } |

II – CÔNG VIỆC VỀ NHÀ

**1. Bóc tác giao diện**

Tiếp tục bóc tách các giao diện cho chức năng quản lý User và Category (tương tự như với Product đã thực hành trên lớp)

**2. Xây dựng View**

Tiếp tục hoàn thiện các View cho chức năng quản lý User và Category (tương tự như với Product đã thực hành trên lớp)