



**FPT POLYTECHNIC**



**THỰC HỌC – THỰC NGHIỆP**



**Conceive Design Implement Operate**

**NODEJS & RESFUL WEB  
SERVICE**

**TEMPLATE ENGINE**

<http://www.poly.edu.vn>

- ⊙ Biết được Template là gì
- ⊙ Tạo và sử dụng được EJS template
- ⊙ Làm việc tốt với form



- 📖 Template là gì? Tại sao cần template
- 📖 Tổ chức và sử dụng template ESJ engine
- 📖 Form với template EJS





# PHẦN 1: TỔNG QUAN VÀ TEMPLATE EJS

- ❑ Là dạng view được sử dụng trong express nodejs
- ❑ Giúp người lập trình tạo các view phản hồi yêu cầu của client một cách dễ dàng

# TỔNG QUAN TEMPLATE ENGINE

EJS



```
<tag><%=biến%></tag>
```



Sử dụng HTML và  
Javascript đơn thuần

Pug(Jade)



```
<tag> #{biến}
```



Sử dụng HTML và  
ngôn ngữ của  
template

Handlebars



```
<tag>{{biến}}</tag>
```



Sử dụng HTML và  
ngôn ngữ của  
template

Cài đặt module ejs



Chuyển tới project->  
Npm install ejs

Tổ chức cấu trúc chứa view

```

  ✓ Lab02
    > node_modules
    ✓ views
      <> 404.ejs
      <> add-product.ejs
      <> shop.ejs
    {} package-lock.json
    {} package.json
    JS server.js
  
```

- ❑ Khai báo sử dụng ejs engine trong server và dùng hàm render để trả view cho client

```
Lab02 > JS server.js > ...
1   const express=require('express');
2   var app=express();
3   const port=3000;
4
5   //khai bao sử dụng template ejs
6   app.set("view engine", "ejs");
7   app.set("views", "./views");
8
9   //router
10  app.get("/", (req, res)=>{
11    |    res.render('shop');
12  })
13
14  app.listen(port, ()=>{
15    |    console.log(`ứng dụng đang chạy với port: ${port}`);
16  })
17
```



## ❑ Gắn kết dữ liệu trong EJS: <%= data %>

```
app.get("/", (req, res) => {  
  var today = new Date();  
  currentDay = today.getDay();  
  var day = '';  
  switch (currentDay) {  
    case 0:  
      day = 'Chủ nhật';  
      break;  
    case 1: ...  
    case 2: ...  
    case 3: ...  
    case 4: ...  
    case 5: ...  
    case 6: ...  
    default: ...  
  }  
  res.render('home', {kindOfDay: day});  
})
```

views > <> home.ejs > html

```
<main>  
  <h1>Chào Mừng bạn ghé thăm website</h1>  
  <h2>Hôm nay là <%= kindOfDay %></h2>  
</main>
```



Thực hiện code bên trong ejs template

Dùng cú pháp `<% code js %>`

If...else

views > <> home.ejs >  html

```
<main>
  <h1>Chào Mừng bạn ghé thăm website</h1>
  <% if(kindOfDay=='Chủ nhật' || kindOfDay=='Thứ bảy'){ %>
    <h2>Hôm nay là cuối tuần</h2>
  <% } else { %>
    <h2>Hôm nay là <%= kindOfDay %></h2>
  <% } %>
</main>
```

## Vòng lặp for

```
<main>
  <div class="grid">
    <% for (let product of products) { %>
      <article class="card product-item">
        <header class="card__header">
          <h1 class="product__title"><%= product.title %></h1>
        </header>
        <div class="card__image">
          
        </div>
        <div class="card__content">
          <h2 class="product__price">$<%= product.price %></h2>
          <p class="product__description"><%=product.description %></p>
        </div>
        <div class="card__actions">
          <button class="btn">Add to Cart</button>
        </div>
      </article>
    <% } %>
  </div>
</main>
```

## Sử dụng CSS

```

  ✓ Lab02
    > node_modules
  ✓ public
    ✓ css
      # style.css
    > images
  
```

```

JS server.js > ...
const express=require('express');
var app=express();
const port=3000;

//khai bao sử dụng template ejs
app.set("view engine", "ejs");
app.set("views", "./views");
app.use(express.static("public"));
  
```

Chỉ định public là thư mục gốc cho view truy cập

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet" href="css/style.css">
  <title>Danh mục sản phẩm</title>
</head>
  
```

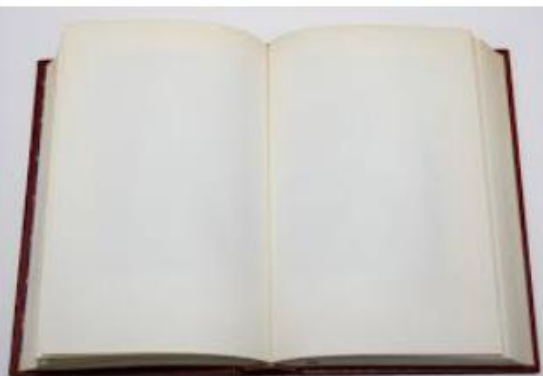
## Sản phẩm Demo

Trang chủ

Sản phẩm

Thêm sản phẩm

### Apple Book

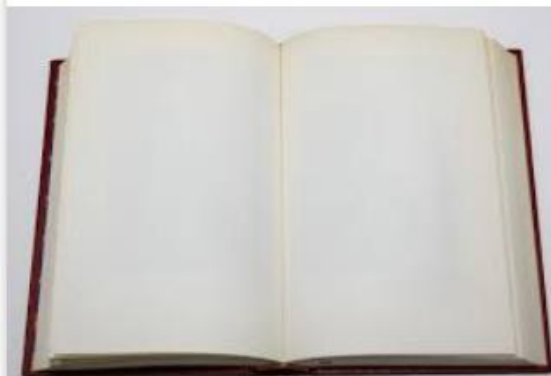


**\$2000**

A very interesting book about so many even more interesting things!

Add to Cart

### Microsoft Book

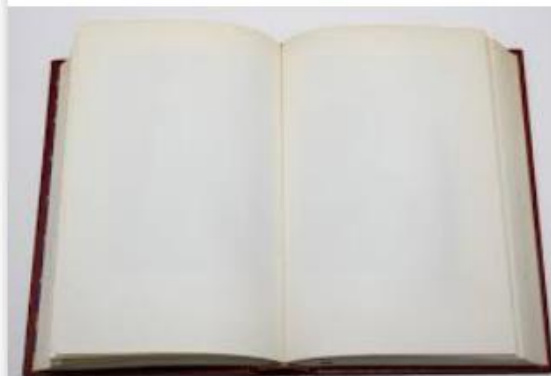


**\$2000**

A very interesting book about so many even more interesting things!

Add to Cart

### VFast Book



**\$1000**

A very interesting book about so many even more interesting things!

Add to Cart

demo



## PHẦN 2: FORM VỚI TEMPLATE EJS

## ❑ Cài đặt và sử dụng module body-parser

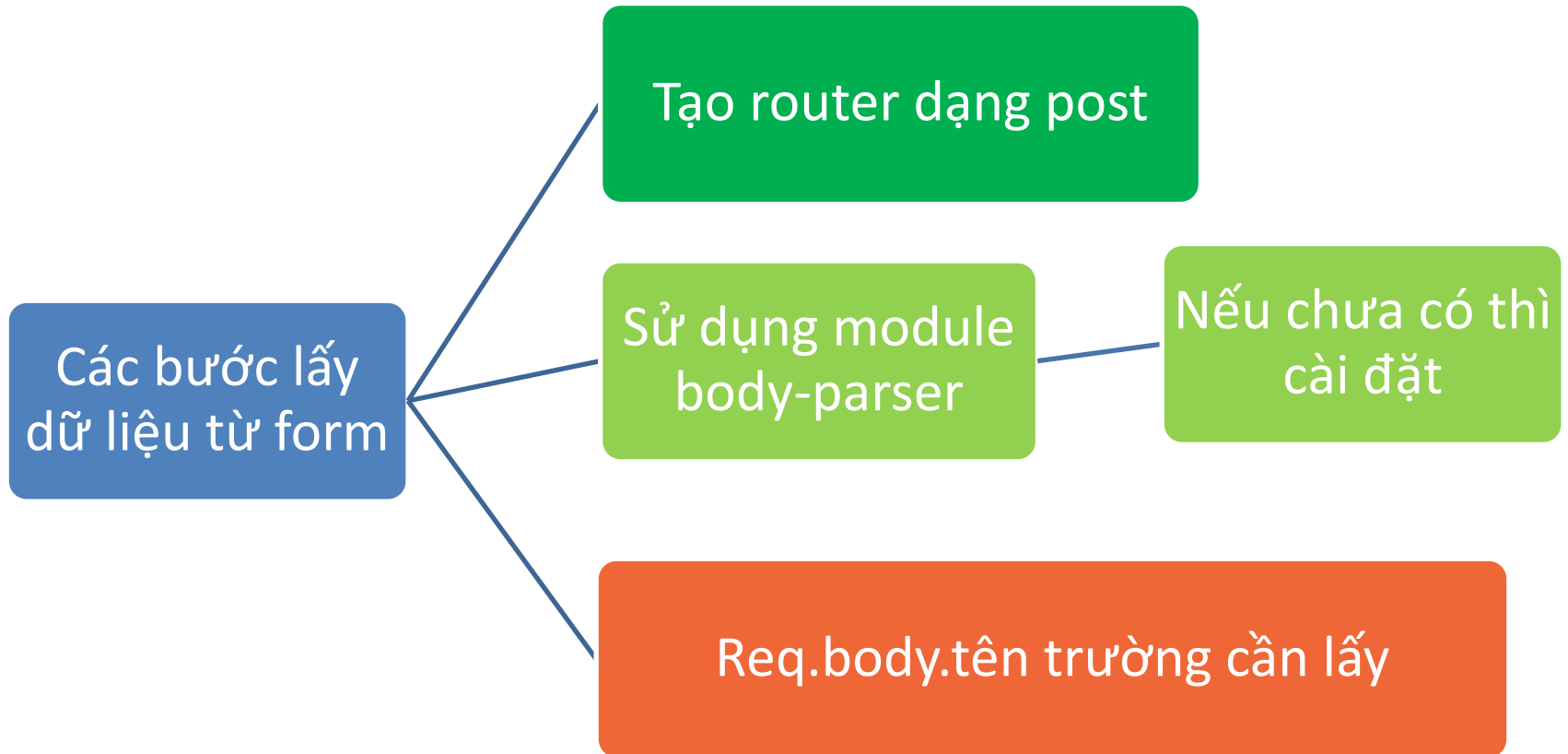
Npm install body-parser

```
JS server.js > ...  
const express=require('express');  
var bodyParser = require('body-parser');  
const app=express();  
const port=3000;  
app.use(bodyParser.urlencoded());
```



## ❑ Thiết kế form

```
<main>  
  <form class="product-form" action="/addnew" method="POST">  
    <div class="form-control">  
      <label for="title">Tên sản phẩm</label>  
      <input type="text" name="productName" id="productName">  
    </div>  
    <button type="submit">Thêm sản phẩm</button>  
  </form>  
</main>
```



## ❑ Nhận dữ liệu từ form

```
app.post('/addnew', (req, res) => {  
  let title=req.body.productName;  
  listProduct.push({ ...  
    res.redirect('/shop');  
});
```

Lấy dữ liệu của  
textbox productName  
trên form

- ❑ Với yêu cầu có dạng Get, tại server khai báo tham số để nhận
- ❑ Dùng thuộc tính params

```
// GET https://localhost:3000/product/123  
  
app.get('product/:id', (req, res) => {  
  console.log(req.params.id) // "123"  
})
```

## □ Dùng query

<http://localhost:3000/product?id=123&name=Tao My>

```
app.get('/product', (req, res, next) => {  
  console.log(req.query);  
});
```

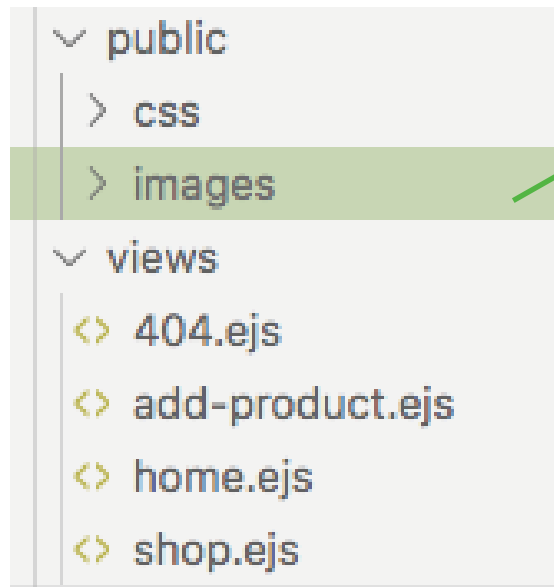
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

2: node

```
> node server.js
```

```
body-parser deprecated undefined extended: provide extended option server.js:6:20  
ứng dụng đang chạy với port: 3000  
{ id: '123', name: 'Tao My' }
```

- ❑ Cài đặt multer => **npm install multer**
- ❑ Thêm thuộc tính cho form là  
***enctype="multipart/form-data"***
- ❑ Dùng hàm **require('multer')** để sử dụng module



Tạo sẵn folder  
chứa hình ảnh

## ❑ Sử dụng multer

```
//khai báo sử dụng multer
const storage=multer.diskStorage({
  destination:function(req,file,cb){
    cb('null','./public/images');
  },
  filename:function(req,file,cb){
    cb('null','${Date.now()}-${file.originalname}');
  }
});
var upload=multer({storage:storage})
```

Thư mục  
chứa hình

Tên hình ảnh

- ✓ public
  - > css
  - > images
- ✓ views
  - <> 404.ejs
  - <> add-product.ejs
  - <> home.ejs
  - <> shop.ejs

## ❑ Sử dụng hàm single để upload

```
app.post('/addnew', upload.single('productImage'), (req, res) => {  
  const file = req.file  
  let title=req.body.productName;  
  let nameImage=file.filename;  
  listProduct.push({...  
    res.redirect('/shop');  
  });  
});
```

```
<form class="product-form" action="/addnew" method="POST" enctype="multipart/form-data">  
  <div class="form-control">  
    <label for="title">Tên sản phẩm</label>  
    <input type="text" name="productName" id="productName">  
  </div>  
  <div class="form-control">  
    <label for="img">Hình ảnh</label>  
    <input type="file" name="productImage" id="productImage" accept="image/*">  
  </div>  
  <button type="submit">Thêm sản phẩm</button>  
</form>
```



## Viết hàm chỉ cho phép upload ảnh

```

var express = require('express');
var router = express.Router();
var multer = require('multer');

var storage = multer.diskStorage({
  destination: function (req, file, cb) {
    cb(null, './public/uploads'); // cần tạo thư mục uploads trong public
  },
  filename: function (req, file, cb) {
    cb(null, Date.now() + '-' + file.originalname)
  }
});

function checkFileUpload(req, file, cb) {
  if (!file.originalname.match(/\.(jpg|png|gif|jpeg)$/)) {
    cb(new Error('Bạn chỉ được upload file ảnh !'));
  }
  else {
    cb(null, true);
  }
}

var upload = multer({ storage:storage, fileFilter:checkFileUpload })
  
```

**Nơi lưu trữ**

**Kiểm tra trước khi upload**

Có phân biệt chữ hoa/thường

## Vấn đề thảo luận

Để upload nhiều ảnh cùng lúc ta thực hiện như thế nào?

## Sử dụng module fs

```
var fs=require('fs');
```

## Dùng hàm unlink để thực hiện xóa

```
app.get("/delete",(req,res)=>{  
  fs.unlink('link tới ảnh cần xóa', function (err) {  
    if (err) throw err;  
    // if no error, file has been deleted successfully  
    console.log('File deleted!');  
  });  
})
```

demo

- ☑ Template là gì? Tại sao cần template
- ☑ Tổ chức và sử dụng template ESJ engine
- ☑ Form với template EJS



thank  
you!