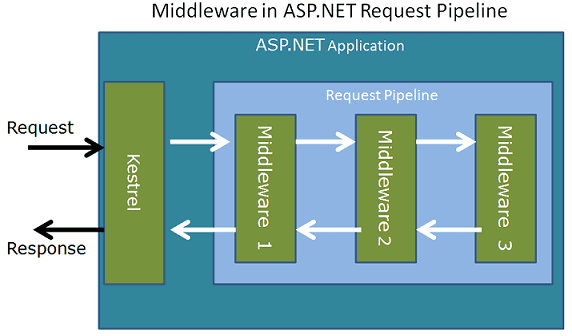
Giới thiệu về công nghệ

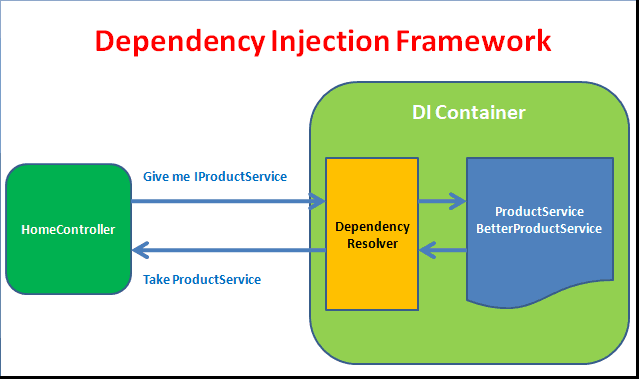
Backend

Giới thiệu về ASP.NET Core

* ASP.NET Core là một mã nguồn mở, có thể tạo ứng dụng bằng .Net core cho các nền tảng như Windows, macOS và Linux x64, x86, ARM32 và ARM64. Frameworks và APIs cung cấp cloud, IoT, client UI và machine learning.
* Một số điểm mới trong .Net core như:
  + .Net core giao diện dòng lệnh (command-line interface) là chuỗi công cụ chạy đa nền tảng hỗ trợ cho việc phát triển (developing), xây dựng dự án (building), chạy chương trình (running) và xuất bản dự án (publishing)
  + Kestrel web server: Là web server mã nguồn mở, đa nền tảng, hướng sự kiện và bất đồng bộ. Được phát triển để chạy trên bất cứ nền tảng nào. Được thêm vào mặc định trong ứng dụng .Net core
  + Middleware và Request Pipeline trong ASP.NET Core:
    - Request Pipeline: là cơ chế bắt đầu khi các request bắt đầu được xử lý với một đối tượng Request đầu vào và kết thúc với đầu ra là một response. Pipeline chỉ ra cách mà ứng dụng phản hồi với HTTP Request. Request đến từ trình duyệt đi qua pipeline và quay trở lại khi xử lý xong để trả về client. Các thành phần đơn lẻ tạo nên pipeline này được gọi là middleware.
    - Middleware: là thành phần của phần mềm đóng vai trò tác động vào request pipeline (luồng request) để xử lý chúng và tạo ra response phản hồi lại client. Mỗi một tiến trình middleware thao tác với các request nhận được từ middleware trước nó. Nó cũng có thể quyết định gọi middleware tiếp theo trong pipeline hoặc trả về response cho middleware ngay trước nó. (ngắt pipeline).

Hình 1.3: Hình ảnh Request Pipeline kết hợp Middleware

* + Cơ chế Dependency Injection trong ASP.NET Core: là một phần của ASP.NET Core. Tất cả các service của framework đều được inject khi chúng ta cần. Lợi ích Dependency Injection:
    - Dependency injection giúp thực hiện kiến trúc lỏng lẻo (loose coupling) trong phần mềm.
    - Code sẽ sạch và dễ đọc hơn
    - Tăng khả năng có thể kiểm thử và bảo trì
    - Cho phép bạn thay đổi triển khai mà không phải thay đổi quá nhiều code.

Hình 1.4: Hình ảnh mô tả cách ProductService được Inject vào HomeController

* + Vòng đời của Dependency Injection: Transient, Singleton, Scoped. Mỗi service có vòng đời khác nhau như:
    - Transient: Một thể hiện mới luôn được tạo, mỗi khi được yêu cầu.
    - Scoped: Tạo một thể hiện mới cho tất cả các scope (Tức mỗi request là một scope). Trong scope thì service được dùng lại.
    - Singleton: Service được tạo chỉ một lần duy nhất.

Ưu điểm của ASP.NET Core

* ASP.NET Core có một số thay đổi kiến trúc dẫn đến module framework nhỏ hơn.
* ASP.NET Core không còn dựa trên System.Web.dll. Nó dựa trên một tập hợp nhiều yếu tố của Nuget packages.
* Điều này cho phép tối ưu ứng dụng của mình chỉ cần những NuGet packages cần thiết.
* Lợi ích của diện tích bề mặt ứng dụng nhỏ hơn thì bảo mật chặt chẽ hơn, giảm dịch vụ, cải thiện hiệu suất và giảm chi phí.

Lí do sử dụng công nghệ:

Front-end:

Mustache Template:

Quá trình phát triển ứng dụng, chúng ta thường viết các nhiều đoạn mã HTML trong mã Javascript dưới dạng kiểu chuỗi. Gây khó khăn cho quá trình phát triển và bảo trì mã nguồn. Vì vậy thư viện MustacheJS là một khuôn mẫu HTML. Việc gắn chặt giao diện người dùng và logic khiến cho ứng dụng khó khăn hơn để phát triển và bảo trì lại sau này.

Thư viện Mustache ra đời giải quyết vấn đề này bằng cách tách các định nghĩa giao diện người dùng ra khỏi dữ liệu, Mustache giúp kết xuất giao diện nhanh hơn sau mỗi lần kết xuất sau, do cơ chế lưu cache.

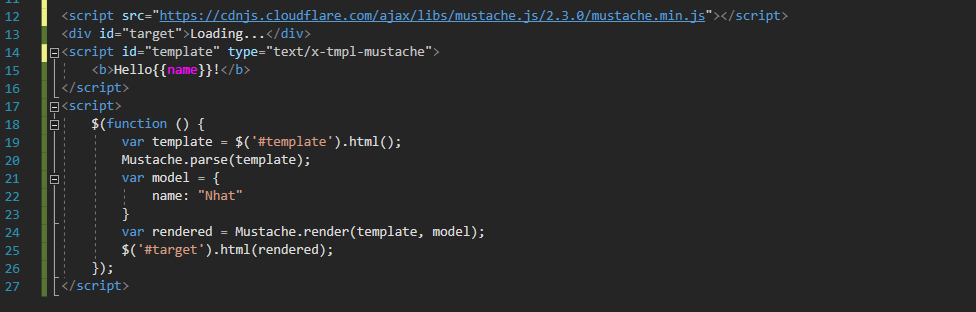
Cài đặt Mustache.js

* Cài đặt thông qua npm

Ta dùng lệnh console: npm install mustache –save.

Cách sử dụng

Một số ví dụ đơn giản

Bước 1: Việc đầu tiên cần làm là include mustache.js trong file HTML:

Bước 2: Khai báo Mustache Template sử dụng thẻ script:

* + Đặt id cho script và khai báo **type="x-tmpl-mustache"** để thông báo đây là a
  + Bên trong đoạn HTML dùng là template. Với đặc điểm quan trọng là mỗi tham số được đặt trong cặp dấu **{{variable}}.** Qua đó, Mustache sẽ biết và thay nó với đối số tương ứng khi truyền vào.
  + Bước 3: Sử dụng hàm html trong Jquery để xuất HTML ra giao diện

<https://github.com/janl/mustache.js>

<https://techblog.vn/tim-hieu-ve-x-template-trong-jquery>

<https://completejavascript.com/mustache-template-voi-jquery>