**Sử dụng Next.js với .NET Framework**

**Giới Thiệu**

Next.js là gì? Next.js là một React framework giúp bạn tạo ra ứng dụng web (Web Application) một cách nhanh chóng và thuận tiện vì Next.js hỗ trợ rất nhiều công cụ hữu ích trong mà các bạn cần trong quá trình làm ra một Web App. Nó có thể tạo ra một ứng dụng web react full-stack với những công cụ mà nó hỗ trợ! Từ Client và Server Rendering, Dynamic Routes đến API Routes, mọi thứ dường như rất dễ dàng hơn rất nhiều so với ứng dụng React thuần.

**Tài liệu liên quan**

- [React Docs](https://reactjs.org/docs/getting-started.html)

- [Next.js Docs](https://nextjs.org/docs/getting-started)

**Điều kiện**

Trước khi bắt đầu, có một số điều kiện cần thiết để cài đặt Next.js như là phần Views (hoặc Frontend) với .NET Framework:

- Visual Studio Community 2022 (Tài liệu được viết và thử nghiệm trên VS Community 2022 version 17.5.0)

- Solution trên Visual Studio đã bao gồm project .NET Web Application 4.7.2 với template Web API (để sử dụng làm Models, Controller)

- [Node.js 14.6.0](https://nodejs.org) hoặc mới hơn

**Thực hiện**

1. Tạo 1 project Next.js theo hướng dẫn trên [tài liệu liên quan](#Docs).

- [Tự động (có hướng dẫn)](https://nextjs.org/docs/getting-started#automatic-setup)

- [Thủ công](https://nextjs.org/docs/getting-started#manual-setup)

\*Lưu ý: Hãy tạo thẳng project Next.js trong folder chứa Solution, nó sẽ giúp việc Liên kết dễ dàng hơn. Nếu bạn lỡ tạo project Next.js ở folder nào khác, hãy cut/di chuyển cả folder đó vào folder chứa Solution.

2. Thêm dự án Next.js vào Solution đang chứa Project Web API

2.1. Mở Solution

2.2. Để ý cửa sổ Solution Explorer. Click chuột phải vào Solution -> Add -> New Project…

Graphical user interface, text

Description automatically generated

2.3. Tìm chọn template From Existing Node.js code -> Next

A screenshot of a computer

Description automatically generated with medium confidence

2.4. Cửa sổ Configure your new project hiện ra, ở bước này bạn không cần quan tâm đến field Project name, chỉ cần kiểm tra xem Location có đúng đang ở folder chứa Solution cần làm hiện tại là được. Nếu xong nhấn Create

Text

Description automatically generated

2.5. Cửa sổ Create New Project from Existing Node.js Code hiện ra. Ở field Enter or browse to the folder containing your Node.js code, bạn hãy dẫn đường dẫn đến folder project Next.js bạn vừa tạo ở bước 1. Chọn Next.

Text

Description automatically generated

Figure : Trước khi dẫn đường dẫn chứa project Next.js

Graphical user interface, text

Description automatically generated

Figure : Sau khi dẫn đến folder của project Next.js

\*Lưu ý: Nếu bạn sử dụng TypeScript cho project Next.js. Hãy chọn TypeScript.

2.6. Chọn Next

A screenshot of a computer

Description automatically generated with medium confidence

2.7. Đến đây, hãy nhập vào field này tên của project Next.js, như project Next.js mình tạo tên là “demo-next” nên mình cũng sẽ điền vào như vậy.

Graphical user interface, text, application

Description automatically generated

Thông thường, project sẽ có trùng tên với folder chứ project, để chắc chắn, bạn hãy vào file package.json của project Next.js để xem chính xác tên project Next.js của bạn là gì. Tên chính xác của project nằm ở field “name” trong file package.json

Text

Description automatically generated

Figure : Giá trị của field "name" chính là tên của project

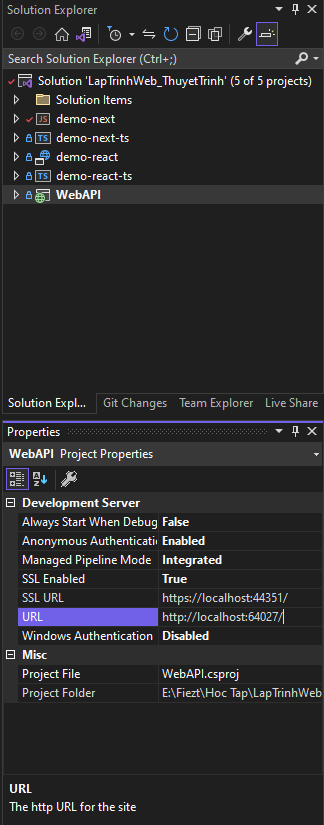
Nếu đã hoàn tất, nhấn Finish.

Graphical user interface, text, application, chat or text message

Description automatically generated2.8. Kiểm tra cửa sổ Solution Explorer, bạn sẽ thấy project Next.js của bạn đã được thêm vào.

3. Kiểm tra URL của Web API bằng cách nhấn vào Project Web API trong Solution Explorer. Tại cửa sổ Properties, bạn hãy tìm field URL (không phải SSL URL) và copy giá trị của field đó chuẩn bị cho bước 4.

\*Lưu ý: Nếu không thấy cửa sổ Properties, nhấn F4 hoặc View -> Properties Window trước khi thực hiện



4. Sổ project Next.js bạn vừa thêm vào Solution (double click hoặc nhấn mũi tên bên cạnh project), tìm file next.config.js và thêm 1 hàm rewrites vào Object nextConfig có sẵn. ([Tham khảo](https://nextjs.org/docs/api-reference/next.config.js/rewrites" \l "rewriting-to-an-external-url), [ví dụ](https://github.com/vercel/next.js/blob/canary/examples/rewrites/next.config.js))



Với {URL} hãy thay bằng đoạn url bạn đã copy ở bước 3, như ví dụ ở đây sẽ là http://localhost:64027

A screenshot of a computer

Description automatically generated with medium confidence

Figure : Trước khi thêm hàm rewrites

Graphical user interface, text, application

Description automatically generated

Figure : Sau khi thêm hàm rewrites

Vậy là đã hoàn tất, hãy kiểm thử xem bạn có thật sự hoàn tất bằng cách khởi động cả 2 project.

Với Web API, chạy bằng IIS Express trong chính Visual Studio.

Graphical user interface, text, application, chat or text message

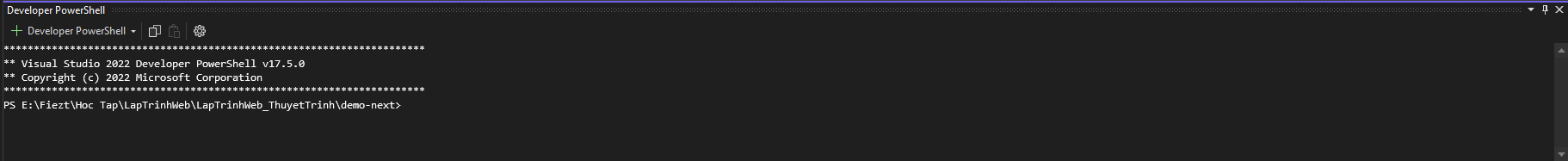
Description automatically generated

Để khởi động Next.js, bạn hãy sử dụng lệnh “npm run dev” trong folder project Next.js bằng cách chuột phải vào project -> chọn Open in Terminal.

A screenshot of a computer

Description automatically generated with medium confidence

Để ý bên dưới của giao diện, bạn sẽ thấy cửa sổ Developer PowerShell hiện ra. Gõ “npm run dev” vào đó rồi nhấn enter



Khi đã thấy dòng ready – started server…, url: <http://localhost:3000>. Bạn hãy mở trình duyệt và truy cập vào url <http://localhost:3000> để xem Next.js có chạy chưa.

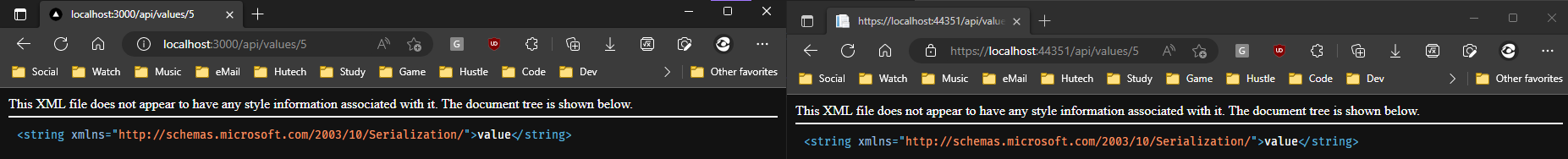
Text

Description automatically generated

Nếu thấy có nội dung hiện ra thì app đã chạy, tiếp theo bạn hãy truy cập tiếp vào route api và so sánh với api của Web API xem đã nội dung trả về đã giống nhau chưa. Nếu đã giống thì xin chúc mừng bạn đã hoàn thành rồi!

Kết quả test controller Values với phương thức GET:  
Graphical user interface, text, application

Description automatically generated



Bạn cũng có thể test phương thức khác của api bằng các công cụ khác nhau như Postman, Reqbin v.v

**Tổng Kết**

Vậy cách làm trên có ý nghĩa gì? Nó sẽ không chỉ giúp các bạn làm đẹp đường dẫn api mà còn bảo mật được phần Back-end (Controller và Models) khi không phải truy vấn trực tiếp url của Backend mà thông qua url của Front-end (Views) khi bạn sử dụng Client-side rendering nói chung và React nói riêng.