Đề tài:

Tìm hiểu về .NET và React, xây dựng trang web tìm kiếm thông tin việc làm IT cho khoa CNTT.

Những năm hiện nay, các ngành liên quan tới công nghệ thông tin đang là xu hướng, em tạo ra trang web này nhằm mục đích giúp các các bạn sinh viên tìm được môi trường làm việc phù hợp, các nhà tuyển dụng có thể tìm được nhân viên mới cho doanh nghiệp.

3 Quyền chính:

* Nhà tuyển dụng: Đăng bài, xem cv ứng viên, xem thông tin ứng viên, tìm kiếm ứng viên, Quản lý các tài khoản phụ, quản lý các cv đã nộp, quản lý thông tin doanh nghiệp
* Ứng viên: Đăng CV, tìm kiếm công việc dựa trên hagtag, tìm kiếm công ty, nộp cv cho nhà tuyển dụng, xem kết quả sau khi ứng tuyển (tiến độ - progress), quản lý chỉnh sửa thông tin cá nhân.
* Khoa cntt: vai trò quản trị, thống kê số lượng, thực trạng, thông tin của SV tại từng DN",…

Giúp ích:

* Hỗ trợ sinh viên tìm được công việc mong muốn
* Hỗ trợ nhà tuyển dụng tìm được nhân viên như mong muốn
* Hỗ trợ sinh viên khoa công nghệ thông tin có việc làm sau khi tốt nghiệp (năm 3, năm 4)

Các chức năng chính

* Quản lý doanh nghiệp: thêm xóa sửa, tìm kiếm, doanh nghiệp đăng ký tài khoản, cập nhật thông tin doanh nghiệp: tên, địa chỉ, lĩnh vực, mô tả,…
* Quản lý ứng viên: thêm xóa sửa tìm kiếm ứng viên, chỉnh sửa thông tin ứng viên: tên, cv, năm sinh, năm kinh nghiệm, chuyên môn. Kinh nghiệm làm việc, hình ảnh, github, linkedin, sdt, email, trường, trung tâm đào tạo, chứng chỉ
* Quản lý công việc: tiêu đề, mô tả, phúc lợi, yêu cầu, số lương tuyển, tiền lương
* Tìm kiếm ứng viên: tìm theo năm kn, theo công nghệ, theo ngôn ngữ, framework, theo tên, địa điểm,…
* Tìm kiếm công việc: tìm theo level, kn, công nghệ, ngôn ngữ, framewỏk, theo tên, địa điểm, vị trí,…
* Ứng viên tạo cv,…
* Thống kê: thực trạng, công nghệ nổi bật, các bài đăng,…

Phi chức năng:

* Dùng React 18.2 , .NET core 6
* Xác thực bằng JWT, refresh token || OAuth 2
* Database mysql hoặc sql server
* Nếu được thì làm
* Sử dụng fulltextsearch || elasticsearch

Sinh viên viên đăng nhập bằng gmail sinh viên

Nhà tuyển dụng: gửi form trên trang xong khoa duyệt mới có tài khoản

Khoa: Duyệt tài khoản cho nhà tuyển dụng

Chức năng:

Sinh viên:

* Đăng nhập = email,
* chỉnh sửa thông tin cá nhân,
* đăng CV tìm việc,
* tìm kiếm việc làm,
* nộp cv vào công ti,
* xem tiến độ nộp cv(đã nộp, đã xem, phỏng vấn,…)

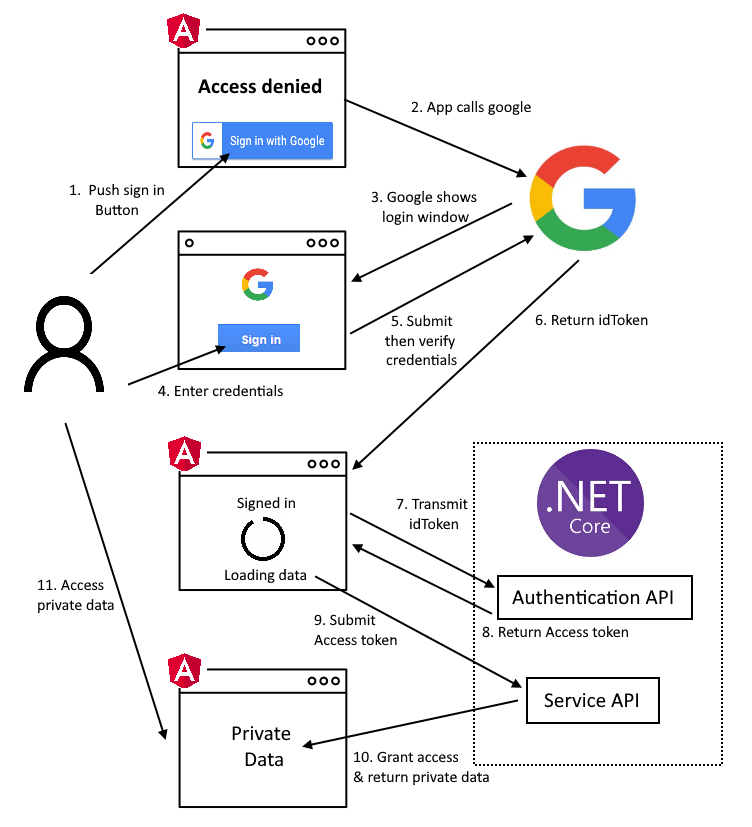
Nhà tuyển dụng:

* Đăng ký, đăng nhập,
* chỉnh sửa thông tin doanh nghiệp,
* đăng tin tuyển dụng,
* tìm kiếm sinh viên thông qua các sinh viên có nhu cầu tìm việc(bật tìm kiếm việc làm),
* xem thông thông tin sinh viên,
* duyệt CV sinh viên đã nộp

Khoa:

* Quản lý tài khoản,
* thống kê việc làm nhà tuyển dụng đăng (theo ngôn ngữ, framework),
* thống kê công việc sinh viên tìm kiếm(theo ngôn ngữ, framework),
* Quản lý bài viết,
* quản lý tin tuyển dụng,
* Quản lý trang thông tin

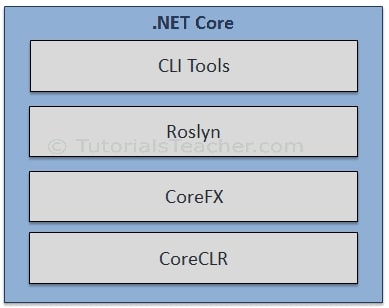
# Sinh viên đăng nhập vào hệ thống



.NET Core

Là phiên bản mới của .NET framwork được hỗ trợ miền phí, duy trì bởi Microsoft. .NET core hỗ trợ đa nền tảng trên Windows, Mac OS và LINUX được dùng để xây dựng các ứng dụng trên nhiều thiết bị khác nhau như mobile, desktop, web, cloud, game, … Với thế mạnh như nhanh, nhẹ và đa nền tảng.

* Đặc điểm:
* Nhất quán giữa các kiến trúc: có thể chạy mã nguồn trên nhiều kiến ​​trúc hệ thống, bao gồm x64, x86 và ARM.
* Hỗ trợ nhiều ngôn ngữ: Có thể sử dụng ngôn ngữ lập trình C#, F# và Visual Basic để phát triển ứng dụng .NET Core.
* Triển khai linh hoạt: Các ứng dụng .NET Core có thể được triển khai song song (cài đặt toàn bộ người dùng hoặc toàn hệ thống).
* Khả năng tương thíc: .NET Core tương thích với .NET Framework và Mono API (thông qua .NET Standard).
* Hỗ trợ bởi Microsoft: Các cập nhật, hỗ trợ được update thường xuyên giúp người sử dụng có thể giải quyết vấn đề nhanh chóng.
* Công cụ CLI: .NET core bao gồm các công cụ CLI (Command – line –interface: Giao tiếp thông qua dòng lệnh) để phát triển và tích hợp liên tục.
* Kiến trúc mô-đun: .NET core hỗ trợ cách tiếp cận kiến trúc mô-đun bằng cách sử dụng các gói NuGet. Có các gói NuGet khác nhau cho các tính năng khác nhau có thể được thêm vào dự án .NET core nếu cần. Nhờ đó bạn sẽ ít tiêu tốn dung lượng bộ nhớ, tăng hiệu suất và dễ bảo trì ứng dụng hơn.
* Các thành phần:
* CLI Tool: Công cụ CLI(Command – line –interface: Giao tiếp thông qua dòng lệnh) là bộ công cụ để phát triển và triển khai .NET core
* Roslyn: Trình biên dịch ngôn ngữ cho C# và Visual Basic
* CoreFX: Tập hợp các thư viện khung (framework library)
* CoreCLR (Command Language Runtime): Môi trường thực thi của .NET core, CoreCLR sử dụng trình biên dịch trung gian dựa trên JIT. Nhờ đó nó có thể dịch mã IL sang mã máy của những nền tảng mà nó hỗ trợ.



* Ưu điểm:
* Đa nền tảng.NET core có thể được sử dụng cho nhiều nền tảng khác nhau như WIndows, Linux hay Mac OS giúp bạn phát triển ứng dụng trên các nền tảng khác nhau một cách dễ dàng.
* .NET core kết hợp với Container Docker hoặc Azure Kubernates Service sẽ giúp bạn tiết kiệm tài nguyên đáng kể.
* Với kiến trúc Mô đun, .NET Core ít tiêu tốn dung lượng bộ nhớ, tăng hiệu suất và dễ bảo trì ứng dụng hơn.
* Tốc độ thực thi nhanh và khả năng mở rộng ứng dụng cũng là một trong những ưu điểm lớn của .NET core.

ReactJS

ReactJS là một opensource được phát triển bởi Facebook từ năm 2013. Đây là một thư viện Javascript được dùng để xây dựng các tương tác với các thành phần trên web. Ưu điểm lớn nhất là việc render dữ liệu được ở cả client và server.

ReactJS giúp xâu dựng giao diện người dùng (UI) và xử lý các tương tác trên giao diện đó. ReactJS giúp chúng ta có khả năng tái sử dụng code bằng JSX và Virtual DOM.

* JSX (JavaScript Syntax Extention): Giúp thay đổi DOM bằng cách HTML và style đơn giản, cho phép viết HTML động.
* Virtual DOM: Bản copy của DOM thật trên trang đó, React sử dụng bản copy này để tìm đúng phần trong DOM cần cập nhật khi bất kì sự kiện nào xảy ra mà không phải load lại trang.

Ưu điểm của React:

* Phù hợp với đa dạng thể loại website
* Tái sử dụng các Component
* Có thể sử dụng cho cả Mobile application
* Thân thiện với SEO
* Debug dễ dàng: Chrome extension dùng trong việc debug trong quá trình phát triển ứng dụng.
* Công cụ phát triển web hot nhất hiện nay

Các thành phần nổi bật của React JS:

* Component: Components giúp phân chia các giao diện người dùng thành các phân nhỏ để dễ dàng quản lý và tái sử dụng. Bao gồm 2 loại chính là Function Component và Class Component.
* Props: là một object được truyền vào trong một components, mỗi components sẽ nhận vào props và trả về react element. Props cho phép chúng ta giao tiếp giữa các components với nhau bằng cách truyền tham số qua lại giữa các components. Khi một components cha truyền cho component con một props thì components con chỉ có thể đọc và không có quyền chỉnh sửa nó bên phía components cha. Cách truyền một props cũng giống như cách mà bạn thêm một attributes cho một element HTML.
* State: là một object có thể được sử dụng để chứa dữ liệu hoặc thông tin về components. State có thể được thay đổi bất cứ khi nào mong muốn. Khác với props bạn có thể truyền props sang các components khác nhau thì state chỉ tồn tại trong phạm vi của components chứa nó, mỗi khi state thay đổi thì components đó sẽ được render lại.
* React: giúp ta có thể để tham chiếu một element trong DOM hoặc từ một class component con đến component cha. Điều này cho phép chúng ta đọc và chỉnh sửa các element đó.
* Ngoài ra việc xử lý các sự kiện, xử lý form gần như tương tự như khi ta viết bằng HTML thuần thông qua các thành phần phía trên để xử lý chúng.

Redux là gì?

* Là một thư viện JS dùng để quản lý và cập nhật state của ứng dụng (state: dữ liệu hay trạng thái trong một ứng dụng về frontend). Hoạt động giống như một kho lưu trữ tập trung cho các state được dùng trong nhiều component hay nhiều nơi trong ứng dụng.
* Là một pattern

Vì sao nên sử dụng redux – redux toolkit

* Redux
* Quản lý Global state.
  + Các component tại mọi nơi trong ứng dụng có thể truy xuất và cập nhật
  + Giải quyết vấn đề truyền dữ liệu vào các cấp component mà không phải truyền tuần tự qua các cấp tránh việc truyền dữ liệu phức tạp.
* Dễ dàng debug, rút ngắn code và đơn giản chúng, có tổ chức.
* Xử lý caching dữ liệu từ Server
* Redux toolkit: Là thư việc Js bọc bên ngoài của redux chưa các package, function cần thiết khi sử dụng redux.
* Giảm thiểu việc cấu hình (config) Redux phức tạp
* Redux yêu cầu rất nhiều boilerplate code (code lặp đi lặp lại)

Khi nào nên xử dụng redux:

* Dự án có số lượng lớn state và các state được sử dụng ở nhiều nơi
* State được cập nhật thường xuyên
* Logic code cập nhật state phức tạp
* Ứng dụng có số lượng code trung bình hoặc lớn và có nhiều làm chung
* Cần debug và muốn xem các state được cập nhật tại bất kì khoảng thời gian nào.

Kiến trúc và các khái niệm của Redux:

* State Management: Quản lý dữ liệu (trạng thái)
* Immutability (bất biến)

Action: là một object do chúng ta quy định gửi từ ứng dụng đến Store bao gồm 2 phần là type (kiểu mô tả action và giá trị tham số truyền vào)

Action creator fuction tạo ra action giúp trả về một object mà không phải viết các object lặp đi lặp lại

Reducer là một function cơ bản nhận 2 tham số state và action xử lý và trả về state mới không làm thay đổi state cũ.

Store là 1 object lưu trữ tất cả state của ứng dụng, cho phép truy cập state thông qua getsate(), update chúng bằng dispash(action), đăng ký listener qua subscribe(listener).

Dispash nhận vào một action và dispash tiếp tục sẽ được đẩy xuống Reducer và trả về 1 state mới giúp tạo nên tính predictable (cùng một state và 1 action thì luôn luôn cho ra một state giống nhau)

Redux toolkit

Là một thư viện giúp việc Redux dễ hơn và thuận tiện hơn.

- Cấu hình Redux quá phức tạp.

- Tôi phải thêm rất nhiều gói để Redux làm bất cứ điều gì mình cần.

- Redux yêu cầu quá nhiều mã viết sẵn.

Bao gồm các phương thức:

* configureStore(): có sẵn Redux Devtools, redux-thunk dùng cho các async actions
* createReducer():
* createActions()
* createSlice()
* createSelector()
* createAsyncThunk()
* createEntityAdapter()
* Thư viện IMMER: Giúp việc thao tác trên dữ liệu, cập nhật dữ liệu imutation dưới dạng code mutiontion.

Tài liệu tham khảo

.NET Core:

<https://ironhackvietnam.edu.vn/net-core-la-gi-phan-biet-net-core-net-framework-net-standard-va-mono/>

https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet-core

ReactJS:

- https://reactjs.org/

- https://quantrimang.com/cong-nghe/reactjs-la-gi-184174

- https://wiki.tino.org/cach-thuc-hoat-dong-cua-reactjs/

https://viblo.asia/p/props-va-state-trong-reactjs-naQZRYvqKvx

Redux:

- https://redux.js.org

- https://topdev.vn/blog/redux-la-gi/

Redux toolkit:

- https://redux-toolkit.js.org/

- https://itzone.com.vn/vi/article/react-gioi-thieu-tong-quat-ve-redux-toolkit/

- https://redux-toolkit.js.org/usage/immer-reducers