**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**BÁO CÁO ĐỀ TÀI**

**XÂY DỰNG TEXT EDITOR CHO LẬP TRÌNH VIÊN**

**Môn học : Lập trình trực quan**

**Giảng viên lý thuyết : Hoàng Văn Hà**

**Giảng viên lý thực hành : Trần Khánh Nguyên**

**Sinh viên thực hiện : Lê Hoàng Thịnh - 19520285**

***TP. Hồ Chí Minh, tháng 1 năm 2021***

**LỜI NÓI ĐẦU**

Text editor là một loại phần mềm đa dụng giúp chúng ta làm việc với văn bản thuần tuý trong các tập tin một cách dễ dàng hơn trong quá trình lập trình và phát triển sản phẩm.

Dựa vào các tính năng hữu ích như syntax highlight cho từng ngôn ngữ, auto brace match hay mở các external terminal, các text editor được sử dụng rộng rãi bởi các lập trình viên. Một số text editor nổi tiếng như Sublime Text, Notepad++ hay VSCode.

Vì thế nhằm củng cố, phát triển và hoàn thiện kiến thức về môn học, em quyết định sẽ xây dựng và phát triển một text editor hoàn chỉnh.

# **LỜI CẢM ƠN**

Trân trọng gửi lời cảm ơn đến thầy Hoàng Văn Hà vì đã tạo điều kiện và cơ hội giúp đỡ em trong quá trình phát triển và hoàn thiện đồ án. Nhờ sự hướng dẫn từ thầy, em đã có được những kiến thức quan trọng và nhất định để hoàn chỉnh chương trình

Trân trọng gửi lời đến thầy Trần Khánh Nguyên đã hỗ trợ và đóng góp ý kiến cho em trong quá trình tìm hiểu và giải quyết những vấn đề trong thời gian hoàn thiện sản phẩm.

Mục lục

[***LỜI NÓI ĐẦU*** 2](#_Toc61088432)

[***LỜI CẢM ƠN*** 3](#_Toc61088433)

[**I.** **TỔNG QUAN** 6](#_Toc61088434)

[1. DẪN NHẬP 6](#_Toc61088435)

[2. CÁC VẤN ĐỀ CHÍNH TRONG ĐỀ TÀI 6](#_Toc61088436)

[a. BÀI TOÁN 6](#_Toc61088437)

[b. GIỚI HẠN/PHẠM VI 6](#_Toc61088438)

[c. CÁC NỘI DUNG NGHIÊN CỨU VÀ THỰC HIỆN 6](#_Toc61088439)

[d. KẾT QUẢ HƯỚNG TỚI 7](#_Toc61088440)

[**II.** **CÁC KIẾN THỨC VÀ CÔNG NGHỆ NỀN TẢNG** 7](#_Toc61088441)

[1. LẬP TRÌNH NGÔN NGỮ C# 7](#_Toc61088442)

[2. CÔNG NGHỆ WPF 7](#_Toc61088443)

[a. ĐỊNH NGHĨA 7](#_Toc61088444)

[b. MỤC TIÊU CƠ BẢN CỦA VIỆC SỬ DỤNG WPF TRONG ỨNG DỤNG 8](#_Toc61088445)

[3. COMMAND VÀ COMMAND BINDING TRONG LẬP TRÌNH WPF 8](#_Toc61088446)

[**III.** **XÂY DỰNG ỨNG DỤNG** 9](#_Toc61088447)

[1. TỔNG QUAN – KHẢO SÁT CÁC ỨNG DỤNG CÓ CÙNG TÍNH NĂNG 9](#_Toc61088448)

[2. PHÂN TÍCH, THIẾT KẾ, CÀI ĐẶT PHẦN MỀM DỰA TRÊN CÁC KIẾN THỨC, CÔNG NGHỆ ĐÃ NGHIÊN CỨU. 11](#_Toc61088449)

[a. Xây dựng giao diện người dùng 11](#_Toc61088450)

[b. Xây phần dựng xử lý logic 13](#_Toc61088451)

[c. Xây dựng cơ sở dữ liệu 14](#_Toc61088452)

[d. Các thành phần, lớp trong ứng dụng 16](#_Toc61088453)

[IV. THỬ NGHIỆM VÀ ĐÁNH GIÁ 21](#_Toc61088454)

[1. MÔI TRƯỜNG THỬ NGHIỆM 21](#_Toc61088455)

[2. KẾT QUẢ THỬ NGHIỆM 21](#_Toc61088456)

[3. KẾT QUẢ ĐÁNH GIÁ SẢN PHẨM 21](#_Toc61088457)

[V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 22](#_Toc61088458)

[1. CÁC KẾT QUẢ ĐẠT ĐƯỢC 22](#_Toc61088459)

[2. HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN 23](#_Toc61088460)

[a. Thuận lợi 23](#_Toc61088461)

[b. Hạn chế 23](#_Toc61088462)

[c. Hướng phát triển 23](#_Toc61088463)

[References 24](#_Toc61088464)

# **TỔNG QUAN**

## DẪN NHẬP

Text editor là một trình soạn thảo văn bản chuyên dụng dành cho các lập trình viên trong quá trình xây dựng, hoàn thiện và phát triển sản phẩm bởi vì các chức năng hữu ích của nó. Cụ thể hơn, text editor hỗ trợ chỉnh sửa các mã nguồn của sản phẩm dưới dạng văn bản thuần túy và hỗ trợ một số tính năng tiện ích mạnh mẽ như highlight các từ khóa cho từng ngôn ngữ lập trình hay tự động đóng mở ngoặc. Đối với một lập trình viên, phần mềm hỗ trợ văn bản là thành phần không thể thiếu trong quá trình làm việc bởi tính đặc thù của nó. Vì thế, nhằm hiểu rõ phương hướng xử lý với các dữ liệu kiểu chuỗi trong văn bản, với các tập tin và phím tắt cũng như có thể phát triển và tái sử dụng lại ứng dụng trong tương lai, trình soạn thảo văn bản nâng cao đã được em chọn làm đề tài cho môn học này.

## CÁC VẤN ĐỀ CHÍNH TRONG ĐỀ TÀI

### BÀI TOÁN

Một text editor hoàn chỉnh phải đạt các chức năng tối thiểu để hỗ trợ các lập trình viên trong quá trình thao tác với mã nguồn. Bên cạnh đó, text editor phải nhanh, giao diễn dễ sử dụng và dễ dàng thao tác với văn bản để các lập trình viên có thể soạn thảo một cách tiện lợi và nhanh nhất.

### GIỚI HẠN/PHẠM VI

Giới hạn ở mức ứng dụng có thể sử dụng tốt với các tính năng hỗ trợ, chưa cần xử lý với các thuật toán tối ưu. Các ngôn ngữ lập trình hỗ trợ là C, C++, Java, và C#.

### CÁC NỘI DUNG NGHIÊN CỨU VÀ THỰC HIỆN

1. Nghiên cứu tổng quan về lập trình WPF và ứng dụng trong quá trình xây dựng sản phẩm
2. Nghiên cứu về Rich Text Area, một text area chính để soạn thảo văn bản
3. Nghiên cứu về Kiến trúc thư mục và ứng dụng vào Folder Container
4. Nghiên cứu về Regular Expression để sử lý chuỗi văn bản

### KẾT QUẢ HƯỚNG TỚI

1. Sử dụng lập trình WPF để cải thiện giao diện đa chức năng
2. Rich Text Area hỗ trợ highlight cho các loại ngôn ngữ tùy chọn
3. Hoàn thiện cây thư mục và hiển thị đường dẫn cho người dùng
4. Ứng dụng Regular Expression vào Tìm kiếm, highlight từ khóa
5. Intellisense cho từng loại ngôn ngữ riêng.

# **CÁC KIẾN THỨC VÀ CÔNG NGHỆ NỀN TẢNG**

## LẬP TRÌNH NGÔN NGỮ C#

C# (hay CSharph) là ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft vào năm 2000. CSharph loại bỏ một vài sự phức tạp và rối rắm của những ngôn ngữ nó kế thừa như Java và c++, bao gồm việc loại bỏ những macro, những template, đa kế thừa, và lớp cơ sở ảo (virtual base class). Ngôn ngữ này làm việc dựa trên nền tảng .NET framework và có khả năng tạo ra các ứng dụng mạnh mẽ cho hệ điều hành Windows.

## CÔNG NGHỆ WPF

### ĐỊNH NGHĨA

* 1. **Windows Presentation Foundation (viết tắt là WPF)** do Microsoft phát triển, là công nghệ kế tiếp Windows Form dùng để xây dựng các ứng dụng dành cho máy trạm chạy hệ điều hành Windows. WPF được giới thiệu từ năm 2006 trong.NET Framework 3.0 (dưới tên gọi Avalon).
  2. WPF sử dụng 2 thư viện lõi là PresentationCore và PresentationFramework để xử lý các điều hướng, ràng buộc dữ liệu, sự kiện và quản lý giao diện. WPF dựa trên nền tảng đồ họa là DirectX, xử lý vector, hỗ trợ gam màu rộng, cho phép tùy biến giá trị opacity hay tạo gradient một cách dễ dàng, cho phép tạo ảnh không gian 2 chiều hoặc 3 chiều. Thư viện thực thi của WPF tự động tính toán và tận dụng tài nguyên của hệ thống một cách tối ưu để giảm tải cho CPU.
  3. Được xem như thế hệ kế tiếp của WinForms, WPF tăng cường khả năng lập trình giao diện bằng cách cung cấp các API cho phép tận dụng những lợi thế về đa phương tiện hiện đại.
  4. Các control trong WPF có thể được lồng ghép linh động để tạo ra giao diện do được viết bằng XAML. Một ứng dụng WPF có thể được xây dựng để chạy độc lập dưới dạng mở rộng EXE hoặc đóng gói với phần mở rộng là XBAP để có thể tích hợp lên website.

### MỤC TIÊU CƠ BẢN CỦA VIỆC SỬ DỤNG WPF TRONG ỨNG DỤNG

* 1. Cung cấp một nền tảng thống nhất để xây dựng giao diện người dùng.
  2. Cho phép người lập trình và người thiết kế giao diện làm việc cùng nhau một cách dễ dàng
  3. Tách biệt XAML và giao diện người dùng logic cho phép tách biệt rõ ràng vai trò của nhà thiết kế và nhà phát triển.
  4. Xây dựng giao diện rõ ràng, mạch lạc và dễ tiếp cận hơn cho người dùng.

## COMMAND VÀ COMMAND BINDING TRONG LẬP TRÌNH WPF

* 1. ĐỊNH NGHĨA

Trong WPF, command(lệnh) là:

* Lệnh là hành động được thực hiện.
* Nguồn lệnh là đối tượng phát động lệnh.
* Đích lệnh là đối tượng mà lệnh tác động lên.
* Liên kết lệnh là đối tượng ánh xạ logic thực hiện lệnh với lệnh.

Lệnh trong WPF đơn thuần chỉ thông báo cho các phần tử UI biết rằng có một lệnh có ngữ nghĩa như thế đang được phát động - Bản thân phần tử UI phải triển khai/hiện thực hoá logic hành động phản ứng lại.

* 1. MỤC TIÊU CƠ BẢN KHI SỬ DỤNG COMMAND VÀ COMMAND BINDING
     + Làm cho ứng dụng trực quan hơn với người sử dụng bằng cách biểu hiện ràng khi nào tính năng được sử dụng và khi nào bị khóa.
     + Xử lý các logic dễ dàng hơn khi có thể chia xử lý logic thành nhiều lớp so với việc gộp các logic vào một class.
     + Có thể dễ dàng bảo trì và phát triển mã nguồn trong tương lai.

# **XÂY DỰNG ỨNG DỤNG**

## TỔNG QUAN – KHẢO SÁT CÁC ỨNG DỤNG CÓ CÙNG TÍNH NĂNG

* 1. Notepad++

Notepad++ là 1 phần mềm soạn thảo tiện ích của mã nguồn mở được hỗ trợ cho nhiều ngôn ngữ lập trình như HTML, PHP, ASP, C#, Python, Java… và hoạt động trong môi trường Microsoft Windows. Notepad++ là một trong những công cụ chỉnh sửa mã nguồn lâu đời nhất và vẫn được sử dụng và hỗ trong thời điểm hiện tại. Các tính năng mà Notepad++ hỗ trợ bao gồm nhưng không phải tất cả:

* Syntax Highlight cho một số loại ngôn ngữ
* Kiểm tra mở và đóng ngoặc
* Encoding mạnh mẽ
* Multi Documents Editor

Notepad++ Interface


Figure 1: Giao diện của ứng dụng Notepad++

* 1. Sublime Text 3

Sublime Text 3 là một phần mềm lập trình với đầy đủ tính năng để chỉnh sửa các tệp cục bộ hoặc đoạn code. Nó bao gồm các tính năng khác nhau để chỉnh sửa code giúp các nhà lập trình theo dõi các thay đổi này. Các tính năng mà sublime text hỗ trợ bao gồm những không phải tất cả:

* Syntax Highlight (Tô màu từ khóa)
* Auto Indentation (Tự động thụt lề)
* File Type Reconigtion (Nhận diện loại tệp)
* Sidebar with files directory (Hiển thị cấu trúc thư mục tập tin)
* Các plug in package được hỗ trợ bởi cộng đồng
* Built-in Terminal
* Command Pallete

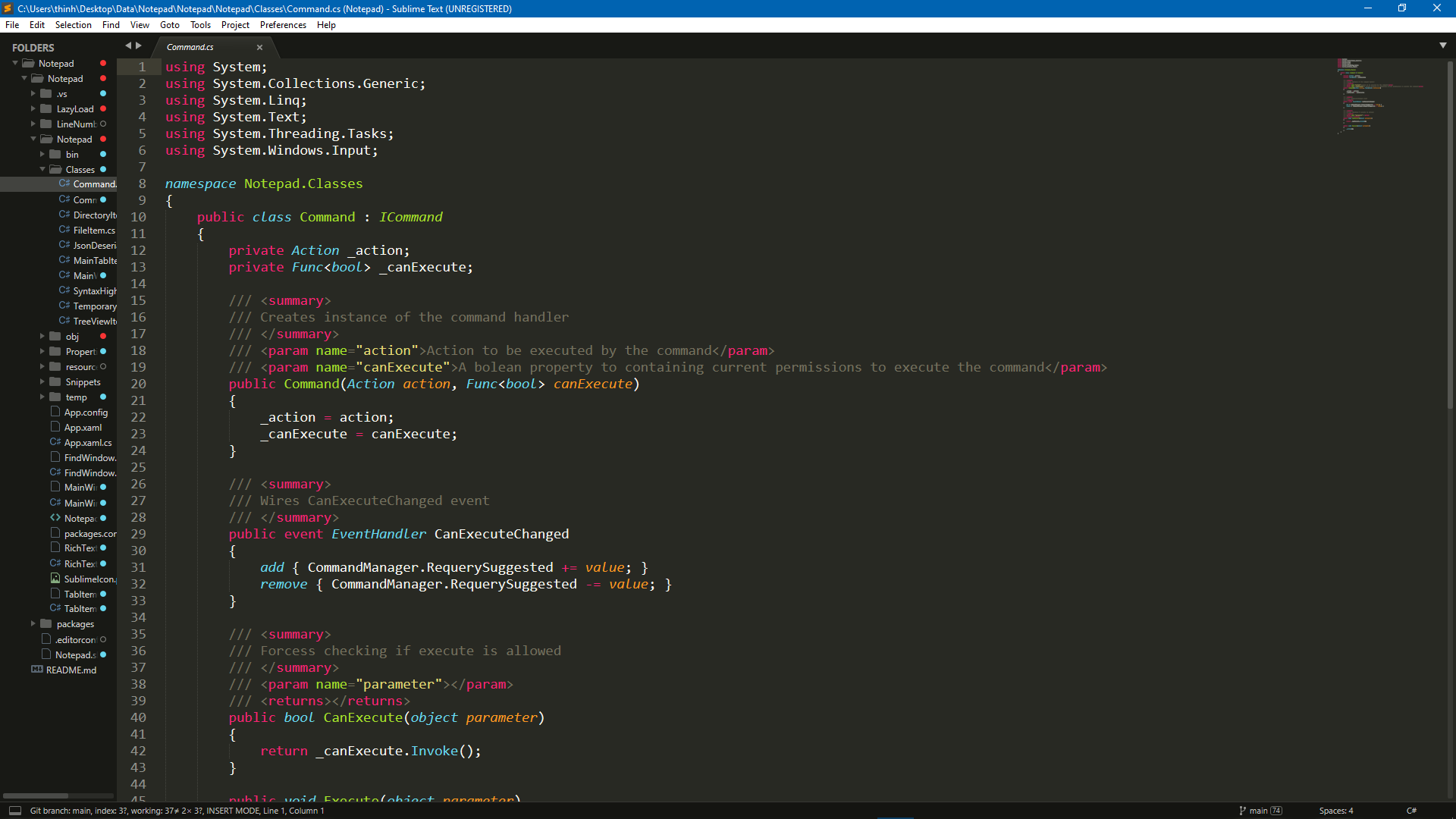


Figure 2: Giao diện ứng dụng Sublime Text 3

## PHÂN TÍCH, THIẾT KẾ, CÀI ĐẶT PHẦN MỀM DỰA TRÊN CÁC KIẾN THỨC, CÔNG NGHỆ ĐÃ NGHIÊN CỨU.

*Khi xây dựng phần mềm hỗ trợ soạn thảo và chỉnh sửa mã nguồn, có ba thành phần chính cần được quan tâm phân tích và thiết kế:*

### Xây dựng giao diện người dùng

Giao diện người dùng phải trực quan, dễ sử dụng, thao tác nhanh để đáp ứng với nhu cầu của lập trình viên. Bên cạnh đó giao diện không được nên quá phức táp tránh gây khó khăn cho người dùng. Để tạo tính trực quan của giao diện, phòng nền sử dụng hai màu tương phản và một số màu bổ trợ tạo nên sự trong sáng và rành mạch của giao diện.

Sử dụng XAML để làm giao diện giúp cho giao diện dễ nhìn và tạo thiện cảm hơn với người dùng

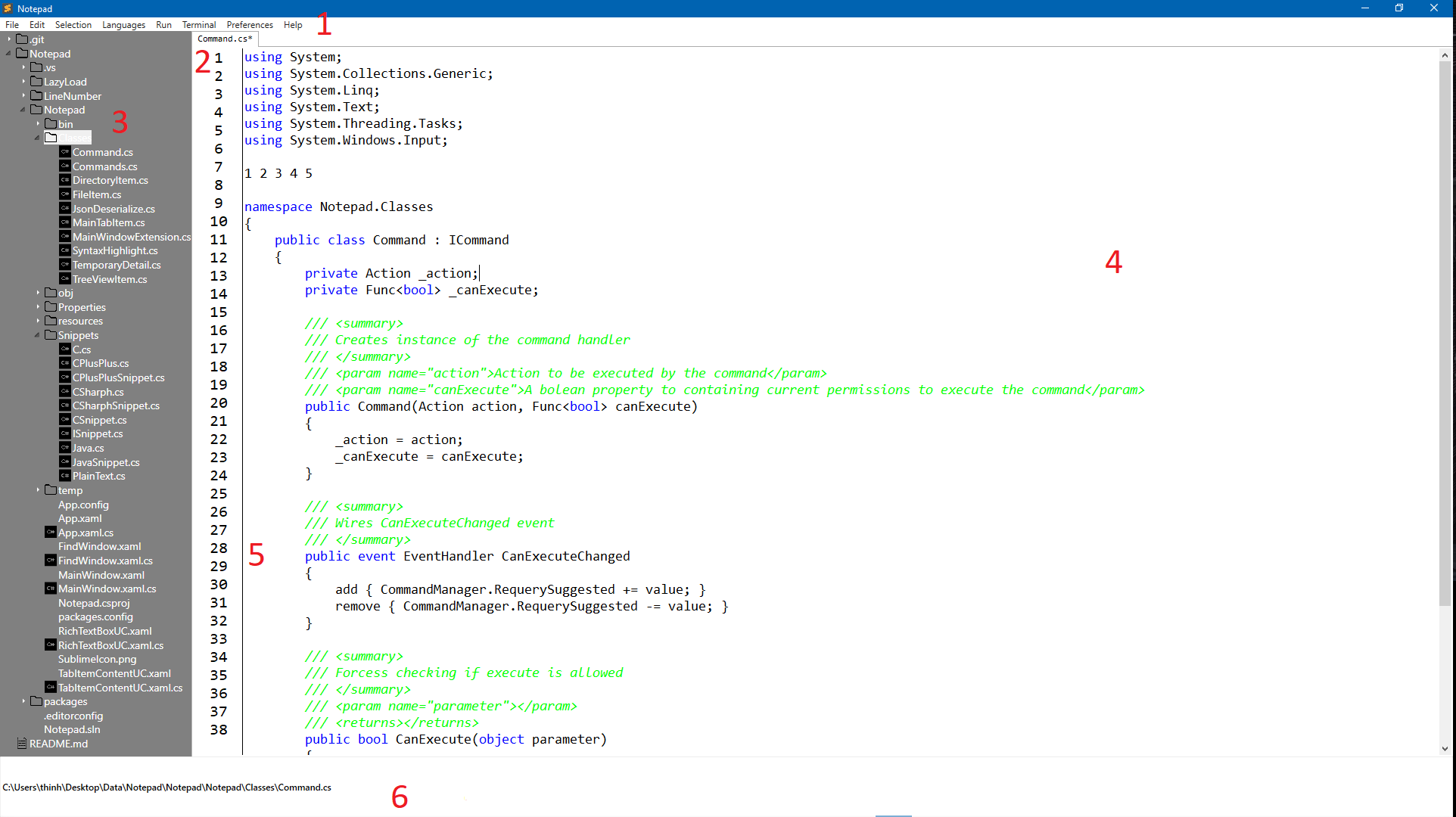


Figure 3:: Giao diện ứng dụng của nhóm

|  |  |  |
| --- | --- | --- |
| **Số thứ tự** | **Tên gọi** | **Chức năng** |
| 1 | Thanh công cụ | Cung cấp các thao tác cho người dùng |
| 2 | Thanh Tab Header | Quản lý các trang được mở |
| 3 | Cây thư mục | Truy cập và xem các tệp trong thư mục chỉ định |
| 4 | Nơi soạn thảo văn bản | Soạn thảo văn bản |
| 5 | Thanh hiển thị số dòng | Hiển chị chỉ số từng dòng |
| 6 | Thanh thông tin | Hiển thị trạng thái văn bản |

* Giao diện chính của ứng dụng là một MainWindow đóng vai trò là một UI container chứa các controls nhỏ như thanh công cụ, Tab Control, thanh thông tin và cây thư mục (nếu có).
* Với mỗi TabItem trong TabControl, nội dung của Tab bao gồm một 2 controls nhỏ chính là TextBox kế thừa từ Window Forms đóng vai trò hiển thị số dòng và RichTextBoxUC đóng vai trong là container của text area.
* RichTextBoxUC User Control chứa một Rich Text Box kế thừa từ Window Forms, là nơi người dùng có thể thao tác với văn bản nguồn.
* Các trang nhỏ đi kèm với chức năng như Find và Replace

### Xây phần dựng xử lý logic

*Để tiện cho việc quản lý, bảo trì và nâng cấp ứng dụng, phần xử lý logic được chia nhỏ và sử dụng các Command với các chức năng chuyên dụng.*

*Với các chức năng nhỏ, các class được phân chia hợp lý và hoàn thành chức năng riêng của nó.*

#### Xây dựng xử lý logic cho các thanh công cụ

* Sử dụng phương thức Command Binding để xử lý những sự kiện từ người dùng.
* Sử dụng Key Binding để nhận các sự kiện có phím tắt.
* Đưa các xử lý logic vào một class để tiện quản lý

#### Xây dựng xử lý logic cho cây thư mục khi được mở

* Xây dựng 3 lớp chính:

|  |  |
| --- | --- |
| **Lớp** | **Chức Năng** |
| FileItem | Lưu loại ngôn ngữ và định dạng icon cho tệp |
| TreeViewItem | Lưu tên và đường dẫn tệp |
| DirecttoryItem | Lưu tên và đường dẫn thư mục |

* Khi người dùng chọn vào một tệp bất kỳ sẽ mở tệp đó lên ở TabItem, nếu tiếp tục ấn thì tệp đó sẽ tắt và mở tệp khác.

#### Xây dựng xử lý logic cho thanh thông tin

* Mỗi khi một chỉ số của Tab Control thay đổi, thanh thông tin sẽ được chỉnh sửa lại bao gồm cập nhật đường dẫn (nếu có).

#### Xây dựng xử lý logic cho từng Tab

* 1. Xử lý với Tab Item
  + Tạo một lớp mới MainTabItem kế thừa từ TabItem nhưng thêm các thuộc tính chính:
    - FilePath: lưu đường dẫn của tệp
    - IsSaved: lưu trạng thái lưu của file
    - IsPinned: lưu trạng thái được ghim của file
    - Data: lưu giá trị của đoạn văn bản đang được chỉnh sửa
  1. Xử lý với nội dung của TabItem
     + Nội dung sẽ gồm 2 controls chính được kế thừa từ Window Forms và ứng dụng thông qua nhúng bằng Window Form Host Integration:
       - Text Box: Hiển thị số dòng của văn bản
       - Rich Text Box: Đóng vai trò là vùng soạn thảo văn bản chỉnh bao gồm các phương thức chỉnh sửa, truy cập và tính năng khác

### Xây dựng cơ sở dữ liệu

*Vì là phần mềm thao văn bản nên phần cơ sở dữ liệu cho ứng dụng không quá phức tạp*. *Thao tác với cơ sở dụng liệu sử dụng ngôn ngữ JSON làm ngôn ngữ cho cở sở dữ liệu.*

#### Đối với những văn bản chưa được lưu trữ trong tệp tại máy người sử dụng

Thuộc tính text của Rich Text Box cũng là nơi lưu trữ thông tin đoạn văn bản người dùng đang thao tác.

#### Đối với những văn bản đã được lưu trong file người sử dụng

Cơ sở dữ liệu chính là máy của người sử dụng và được truy cập thông qua thuộc tính FilePath của TabItem đã được kế thừa.

#### Đối với việc lưu trữ những bản nháp trong quá trình soạn thảo.

Các bản nháp trước khi ứng dụng bị tắt sẽ được lưu trữ vào một tệp tạm dưới dạng JSON và sẽ được tái truy cập khi ứng dụng được mở.

Các thông tin được lưu lại bao gồm:

* Đường dẫn của đoạn văn bản đang chỉnh sửa
* Nội dung của đoạn văn bản đang chỉnh sửa
* Trạng thái lưu và chưa lưu của đoạn văn bản
* Ngôn ngữ mà đoạn văn bản đang sử dụng

#### Đối với việc lưu trữ các luật tô màu văn bản của các loại ngôn ngữ

Các luật tô màu, từ khóa, và mã màu sẽ được lưu trữ dưới dạng JSON trong thư mục nguồn khi cài đặt. Người dùng có thể tùy biến chỉnh sửa mã màu cho từng từ ngữ và đặt thêm các từ khóa mình mong muốn. Khi một ngôn ngữ được chọn, ứng dụng sẽ mã hóa JSON dưới dạng các class và lưu trữ các thông tin sau:

* Từ khóa
* Màu sắc
* Cách thức tô màu văn bản
* Loại từ khóa (chuỗi, bình luận, preprocessor…)

### Các thành phần, lớp trong ứng dụng

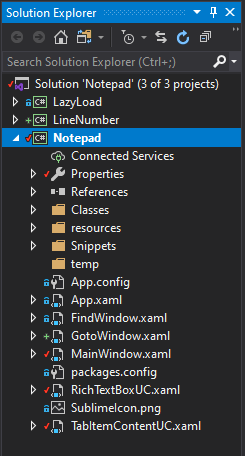


Figure 4: Cấu trúc thư mục

#### Thư mục cài đặt (thư mục gốc)

* Gồm các window được sử dụng trong ứng dụng.
* Gồm các thư mục con chứa các lớp nhỏ.

#### Thư mục Classes

Chứa các lớp con với các chức năng chuyên dụng.

#### Thư mục resources

Chứa các tài nguyên để sử dụng trong ứng dụng bao gồm cái JSON snippet, icon…

#### Thư mục temp

Chứa tệp JSON dùng để lưu lại các bản nháp của lần sử dụng trước

#### Cách thức sử dụng các tính năng trong ứng dụng

|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| New File | Mở một Tab mới |
| Open File | Mở tệp từ máy tính người dùng |
| Open Folder | Mở thư mục (Folder Tree View) |
| Save | Lưu văn bản hiện tại |
| Save As | Lưu văn bản đang soạn thảo ở thư mục chỉ định |
| Save All | Lưu tất cả các văn bản |
| Close File | Đóng Tab hiện tại |
| Close All Files | Đóng tất cả Tab |
| New Window | Cửa sổ mới |
| Exit | Thoát ứng dụng |

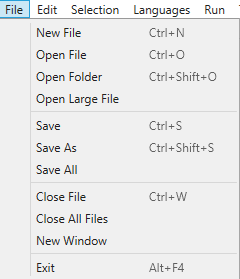


Figure 5: Giao diện file

|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Copy | Sao chép |
| Cut | Cắt |
| Paste | Dán |
| Undo | Quay về trước |
| Redo | Quay về sau |
| Open Containing Folder | Mở thư mục cha trong File Explorer |
| Copy File Path | Sao chép đường dẫn thư mục |

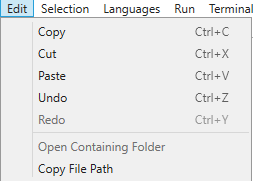
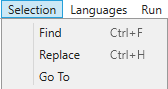


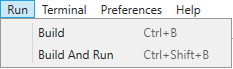
Figure 6: Giao diện edit

|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Find | Tìm kiếm từ khóa |
| Replace | Thay thế từ khóa |
| Go To | Trỏ đến dòng |

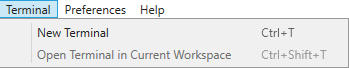
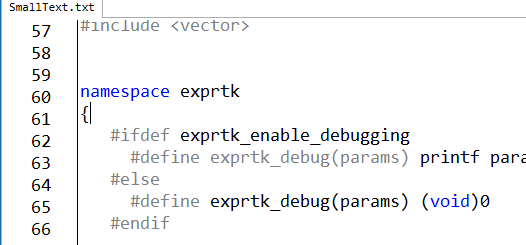
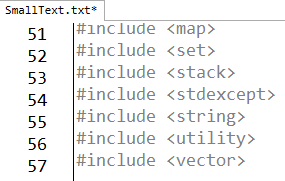


|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| New Termianl | Mở command line trong máy |
| Open Terminal in Current Workspace | Mở command tại thư mục chứa tệp đang chỉnh sửa |

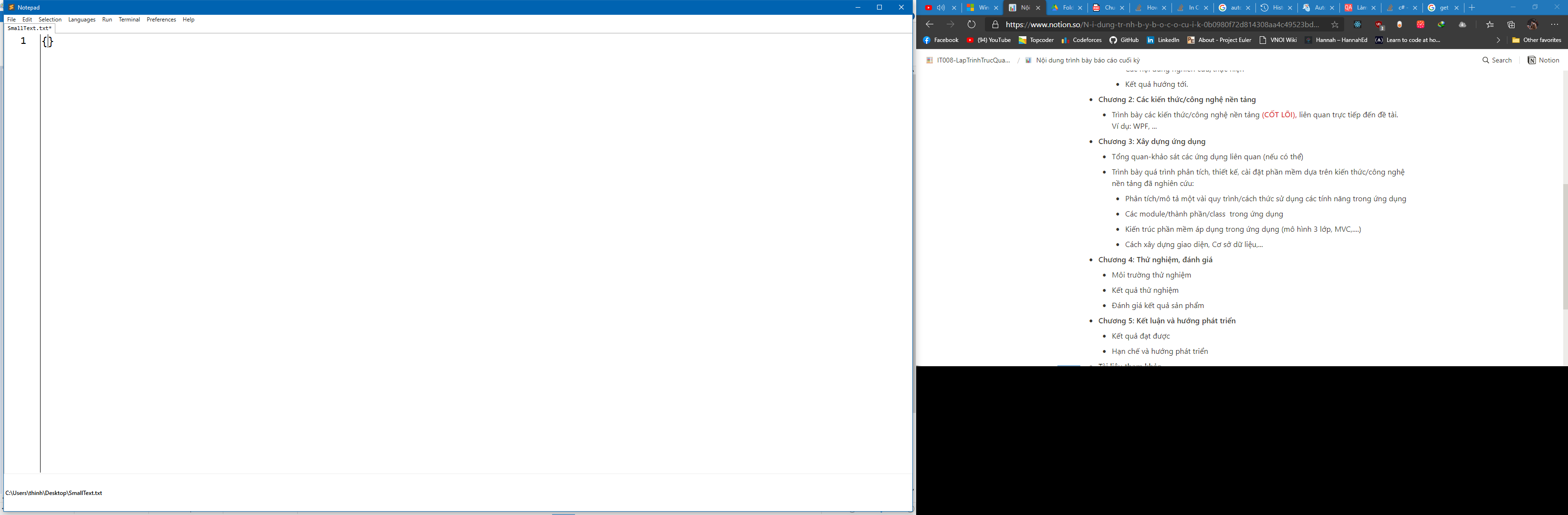
|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Build | Compile chương trình |
| Build and Run | Compile và chạy chương trình |



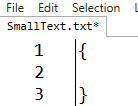
|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Hiển thị dấu \* khi thư mục | Thông báo cho người dùng văn bản đã được chỉnh sửa nhưng chưa lưu |



|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Auto Bracket Matching | Tự động đóng dấu ngoặc khi người dùng nhập văn bản |

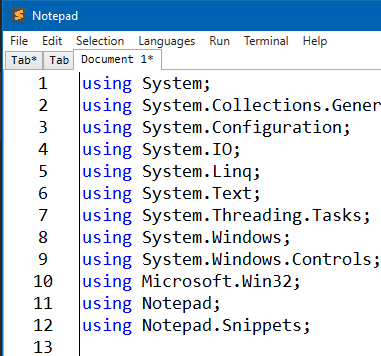
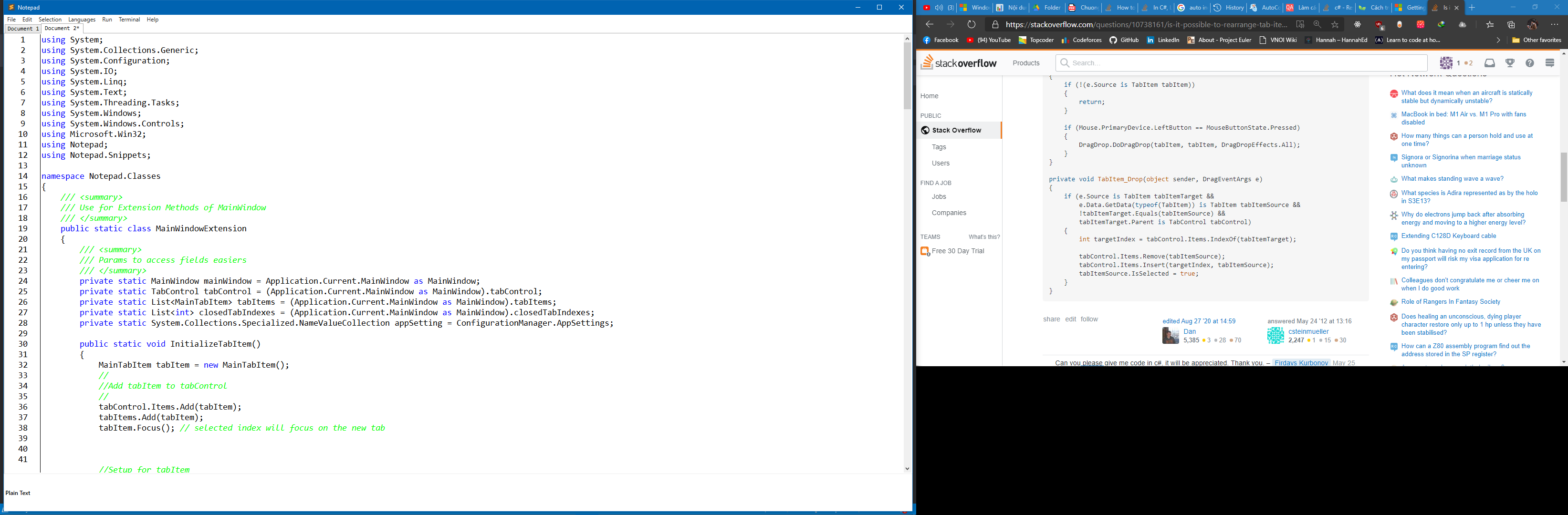


|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Auto Indent Bracket | Tự động cách một hàng và thụt lề để người dùng có thể nhập |



|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Cây thư mực | Xem thư mục được chọn |

Folder Tree View 
Cây thư mục



|  |  |
| --- | --- |
| **Tính năng** | **Mục đích** |
| Pin Tab | Để giao diện trực quan và tránh gây rồi khi lập trình với nhiều Tab soạn thảo văn bản |

*Cụ thể hơn khi người dùng nhấp chuột phải vào một tab header và chọn pin, tab đó sẽ được thu gọn và chuyển về đầu thanh giúp quản lý và ghim các tab dễ dàng hơn.*

# THỬ NGHIỆM VÀ ĐÁNH GIÁ

## MÔI TRƯỜNG THỬ NGHIỆM

* Môi trường thử nghiệm của ứng dụng bao gồm một laptop và một máy tính bàn chạy hệ điều hành Window 10 và phiên bản .NET 4.8.
* Trên các môi trường cùng phiên bản hệ điều hành và .NET từ 4.5 trở lên có sử dụng phần mềm Visual Studio đều có thể sử dụng tốt ứng dụng.
* Mục đích của việc sử dụng phần mềm trên các máy tính khác nhau để thu thập và tìm hiểu những lỗi phát sinh trong quá trình sử dụng phần mềm.

## KẾT QUẢ THỬ NGHIỆM

* Phần mềm hoạt động tốt và không có lỗi phát sinh lớn trên các dòng máy tính khác nhau,
* Các tính năng hoạt động đúng như mong muốn của người dùng, trải nghiệm tốt về tốc độ tô màu văn bản.
* Phản hồi của những người dùng về sản phẩm khá tốt, thu được nhiều góp ý tích cực cho sự bảo trì và phát triển của phần mềm trong tương lai.

## KẾT QUẢ ĐÁNH GIÁ SẢN PHẨM

* Sản phẩm có tính khả dụng cao trong thực tế.
* Các chức năng được cung cấp đều hoạt động tốt, đạt được những yêu cầu tối thiểu, không có tính năng bị dư thừa.
* Phần mềm phù hợp với người dùng với phân khúc định hướng, hỗ trợ tối đa cho những lập trình viên và có thể mở rộng để tiếp cận với đa dạng người dùng hơn trong tương lai.
* Tuy nhiên số lượng chức năng còn hạn chế và đôi khi hơi xử lý văn bản lớn chưa được linh hoạt.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## CÁC KẾT QUẢ ĐẠT ĐƯỢC

* Giao diện cơ bản và trực quan, phù hợp với người dùng, có tính rõ ràng, rảnh mạch trong sáng bằng phương pháp sử dụng bảng màu đối lập.
* Hỗ trợ được các loại ngôn ngữ lớn:
  + C++
  + C#
  + C
  + Java
* Syntax Highlighting
  + Ưu việt hơn các chương trình bình thường khi cho người dùng chỉnh sửa các các luật Syntax bằng cách thêm từ khóa, đổi màu từ khóa và các dạng tô màu.
  + Có thể thêm được các ngôn ngữ khác ở phiên bản cập nhật tiếp theo.
* Brace Matching hiệu quả giúp cho thời gian soạn thảo văn bản với các ký tự ngoặc được giảm đi đáng kể.
* Folder Tree View
  + Hỗ trợ xem tất cả các thư mục và tập tin con
  + Hình ảnh trực quan biểu thị cho từng loại ngôn ngữ giúp người dùng có thể truy suất và sửa đổi dễ dàng.
  + Xem các tệp đã lưu và không bật Tab mới giúp cho chương trình chạy nhanh hơn và tăng hiệu suất làm việc.
* Multi document
  + Cho phép mở thư mục trong nhiều tab giúp cho người sử dụng có thể thao tác với nhiều mã nguồn một cách dễ dàng hơn.
* Pin Tab
  + Tính năng cho phép thu gọn header của tab giúp cho quá trình làm việc dễ dàng hơn, tránh gây răc rối rườm rà cho người sử dụng.
  + Di chuyển tab lên đầu để dễ truy cập.
* Save Icon
  + Hiển thị trạng thái đã và chưa lưu của văn bản đang soạn thảo để người dùng có cái nhìn trực quan hơn về các tệp.
* Tính năng Compile và Run với ngôn ngữ C++
  + Hỗ trợ tính năng Compile và Run chạy bằng command line tại máy tính người dùng.

## HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN

### Thuận lợi

* Sử dụng lập trình WPF nên tùy biến giao diện dễ dàng.
* Việc sử dụng RichTextBox của winform giúp tăng tốc độ xử lý văn bản

### Hạn chế

* Một số thuật toán chưa được tối ưu dẫn đến việc load các đoạn văn bản lớn sẽ làm cho chương trình hơi lâu và các tính năng chưa được xử lý nhanh.
* Hiện tại chưa thêm được các loại ngôn ngữ mới vào chương trình.
* Cần có thêm auto ident để tăng tốc độ soạn thảo văn bản của người dùng.
* Cần thêm những tính năng nâng cao hơn ví dụ như Command Pallete hay built-in command giống Sublime Text

### Hướng phát triển

* Sử dụng thuật toán tối ưu hơn với các chức năng. Từ đó giúp tăng tốc độ xử lý và cải thiện tương tác với người dùng
* Tự động thêm từ khóa khi người dùng thêm giao diện mới.
* Hỗ trợ biển dịch với nhiều ngôn ngữ hơn.
* Hỗ trợ Document Map giúp xem trực quan văn bản dễ dàng hơn.
* Hỗ trợ và phát triển nhiều ngôn ngữ hơn trong tương lai.

# References

Adi Lester, 2012. *RichTextBox syntax highlighting.* [Online]   
Available at: https://stackoverflow.com/questions/11940744/richtextbox-syntax-highlighting  
[Accessed 08 13].

Microsoft, C., 2018. *Get started (WPF).* [Online]   
Available at: https://docs.microsoft.com/en-us/dotnet/desktop/wpf/getting-started/introduction-to-wpf-in-vs?view=netframeworkdesktop-4.8

Newtonsoft, 2006. *Json.NET Documentation.* [Online]   
Available at: https://www.newtonsoft.com/json/help/html/Introduction.htm

Selvam, A. R. U., 2020. *Visualize File Structure with Our New WPF TreeView Control.* [Online]   
Available at: https://www.syncfusion.com/blogs/post/visualize-file-structure-with-our-new-wpf-treeview-control.aspx