**ĐẠI HỌC QUỐC GIA TP HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**BÁO CÁO**

**MÔN: ĐỒ ÁN 1**

Đề tài: **Phát triển ứng dụng Betterpad (text editor)**

**GVHD**: Đỗ Thị Thanh Tuyền

**Lớp** : SE121.K21.PMCL

**Thành viên**:

* Huỳnh Quang Minh 17520749
* Lê Anh Vũ 17521269

**TP. HỒ CHÍ MINH, 2020**

**DANH MỤC HÌNH ẢNH**

Hình 1.1 Các chức năng tiêu biểu của Betterpad………………………………

Hình 2.1 Giao diện chính…………………………………………………….

Hình 2.2 Giao diện Format Tool Strip Menu……………………………….

Hình 2.3 Giao diện khi click vào Text Color/ Background Color………………

Hình 2.4 Mục Others dùng để hiển thị hình ảnh, lưu văn bản và hình ảnh vào file và mở file đã lưu để đọc lại ……………………………………………..

Hình 2.5 Mục Code Editor với các chức năng đơn giản ………………………..

Hình 2.6 Giao diện của mục Set list of keyword ………………………………

**MỤC LỤC**

[**NHẬN XÉT CỦA GVHD** 4](#_Toc45914914)

[**MỞ ĐẦU** 5](#_Toc45914915)

[**CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI** 6](#_Toc45914916)

[**1.1. Giới thiệu đề tài :** 6](#_Toc45914917)

[**1.2. Các chức năng chính** 6](#_Toc45914918)

[**1.3. Mục tiêu của đề tài :** 7](#_Toc45914919)

[**CHƯƠNG 2. GIAO DIỆN NGƯỜI DÙNG (UI)** 8](#_Toc45914920)

[**CHƯƠNG 3. CHI TIẾT VỀ CODE** 13](#_Toc45914921)

[**3.1. Chức năng Text Color ( Đổi màu cụm chữ đã chọn ) :** 13](#_Toc45914922)

[**3.2. Chức năng Background Color ( Đổi màu nền của cụm chữ đã chọn ):** 13](#_Toc45914923)

[**3.3. Chức năng Open Image ( Mở hình ảnh từ thư mục hoặc paste từ clipboard ):** 14](#_Toc45914924)

[**3.4. Chức năng Save/Load RTF files :** 14](#_Toc45914925)

[**3.5. Chức năng Set list of keyword (Đổi màu cho các keyword người dùng nhập ):** 16](#_Toc45914926)

[**3.6. Chức năng đổi màu khác cho keyword và từ thường :** 18](#_Toc45914927)

[**3.7. Chức năng Auto fill brackets ( Tự động fill các dấu thiếu cho “(),[],{},””,\” )** 18](#_Toc45914928)

[**CHƯƠNG 4.HƯỚNG DẪN CÀI ĐẶT** 20](#_Toc45914929)

[**4.1. Hướng dẫn cài đặt :** 20](#_Toc45914930)

[**KẾT LUẬN** 21](#_Toc45914931)

[**Nhận xét :** 21](#_Toc45914932)

[**Những chức năng có thể phát triển trong tương lai :** 21](#_Toc45914933)

[**DANH SÁCH TÀI LIỆU THAM KHẢO** 22](#_Toc45914934)

[**PHỤ LỤC** 23](#_Toc45914935)

[**Chart tỷ lệ công việc của từng thành viên** 23](#_Toc45914936)

[**Link source code :** 23](#_Toc45914937)

# **NHẬN XÉT CỦA GVHD**

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

…………………………………………………………………………………………………………………………………………………….....

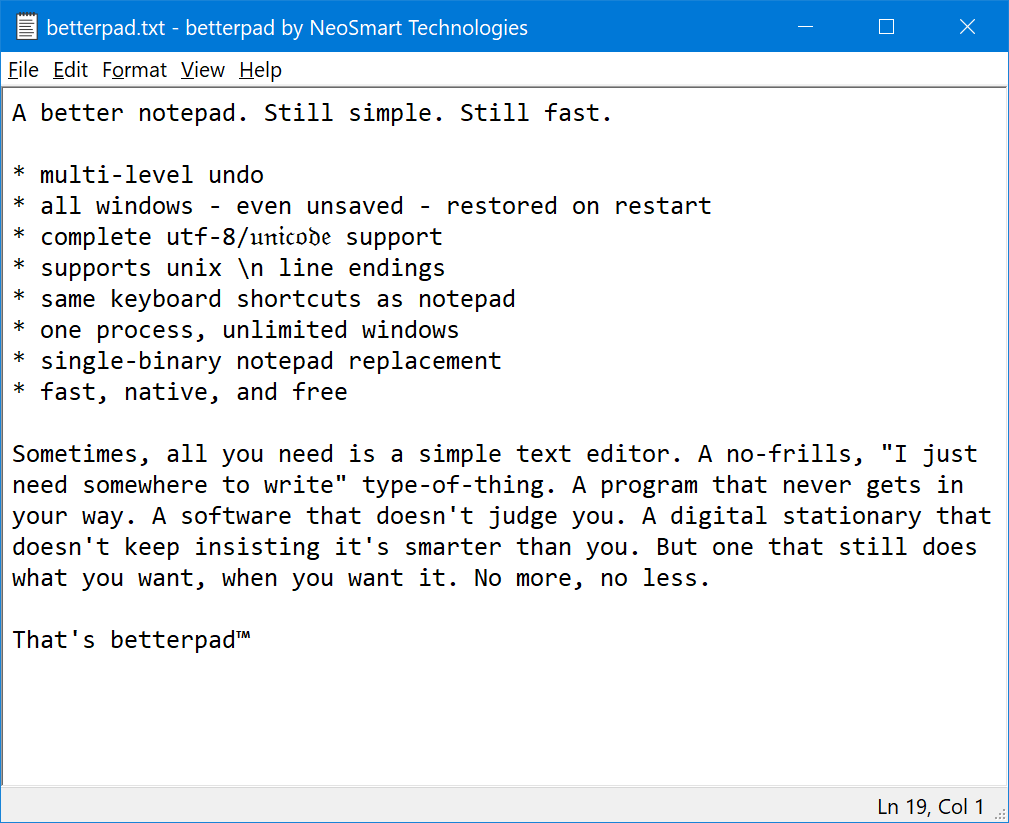
# **MỞ ĐẦU**

Đầu tiên, nhóm chúng em xin gởi lời cảm ơn chân thành đến tới cô Đỗ Thị Thanh Tuyền giáo viên hướng dẫn môn Đồ án 1 – SE121.K21.PMCL thuộc Trường Đại học Công nghệ thông tin – Đại học Quốc gia TP.HCM và quý thầy cô khoa Công Nghệ Phần Mềm đã góp phần giúp cho nhóm chúng em có những kiến thức cơ bản làm nền tảng để thực hiện đề tài này.

# **CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI**

## **1.1. Giới thiệu đề tài :**

BetterPad là một phần mềm soạn thảo mã nguồn mở đơn giản cho phép soạn thảo lưu trữ và đọc dữ liệu từ file text (\*.txt). Betterpad có thể được gọi là một phiên bản nhẹ hơn của notepad ( một phần mềm soạn thảo khá nổi tiếng ) với phương châm “Still simple, Still fast” – đơn giản và nhanh gọn.



*Hình 1.1. Các chức năng tiêu biểu của betterpad*

## **1.2. Các chức năng chính**

Bởi vì Betterpad hiện tại chỉ là một phần mềm soạn thảo cực kì đơn giản nên chúng em quyết định phát triển nó theo hai hướng chính gồm các chức năng như sau :

**1) Phát triển thành Code Editor đơn giản:**

* Cho nhập một danh sách các từ khóa, những từ nào trùng với từ khóa sẽ đổi màu khác.
* Kiểm tra các cặp dấu " ", ( ), { }. Nếu người dùng nhập một dấu thì sẽ tự động thêm vào dấu còn lại để thành cặp. VD : “(“ 🡪 “( )”
* Cho phép chọn màu chữ của từ thường và từ khóa.

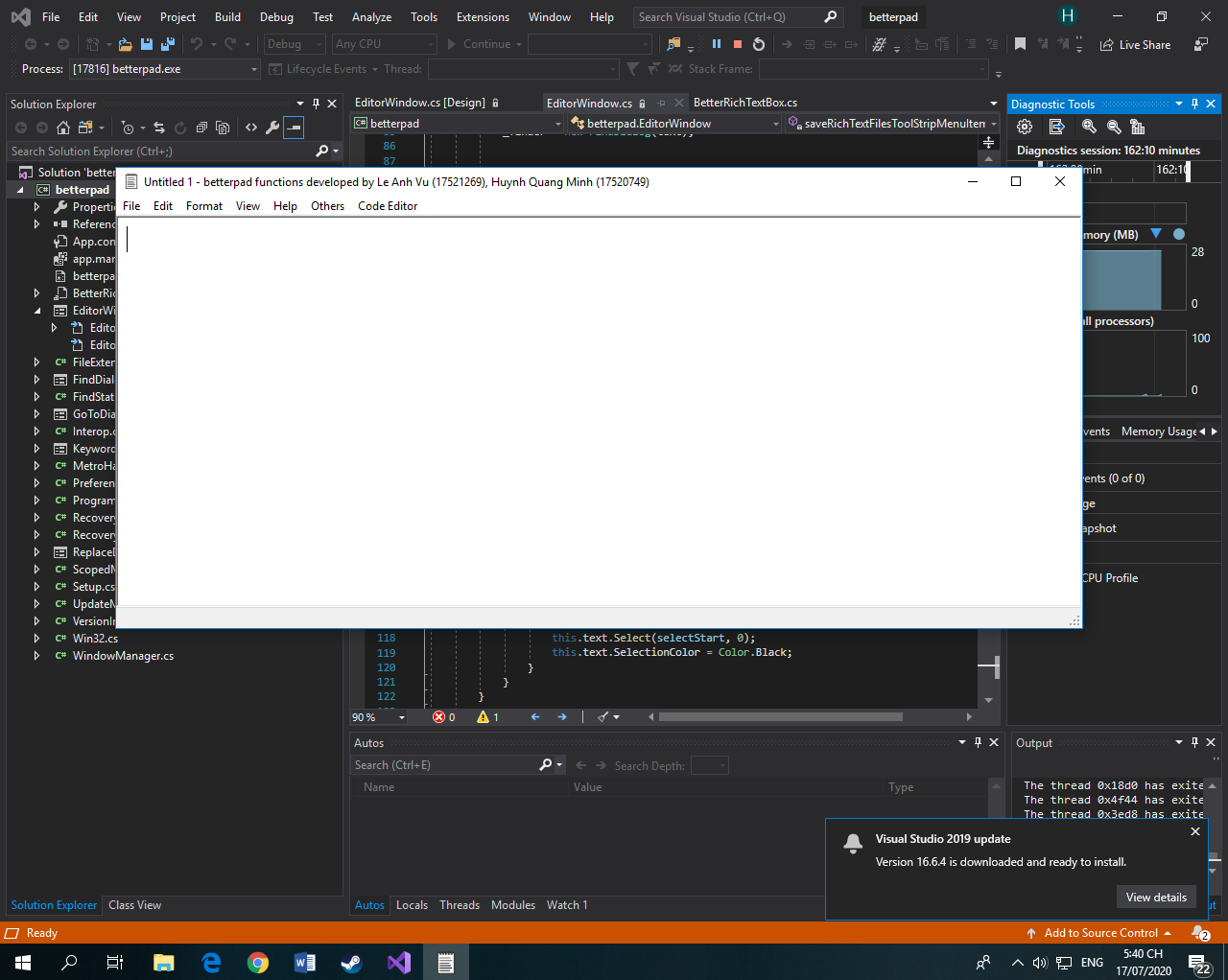
**2) Phát triển thành một Document Editor đơn giản:**

* Cho phép chọn font/màu nền/màu chữ cho từng cụm chữ riêng.
* Cho phép hiển thị hình ảnh (chưa canh lề). Có thể chụp màn hình dùng Screen Snip hoặc key “prt sc” để lưu hình chụp màn hình vào clipboard và paste vào chương trình.
* Lưu nội dung văn bản và hình ảnh vào file và mở file đã lưu để đọc lại. File này nhóm em quyết định chọn định dạng Rich Text Files (\*rtf).

## **1.3. Mục tiêu của đề tài :**

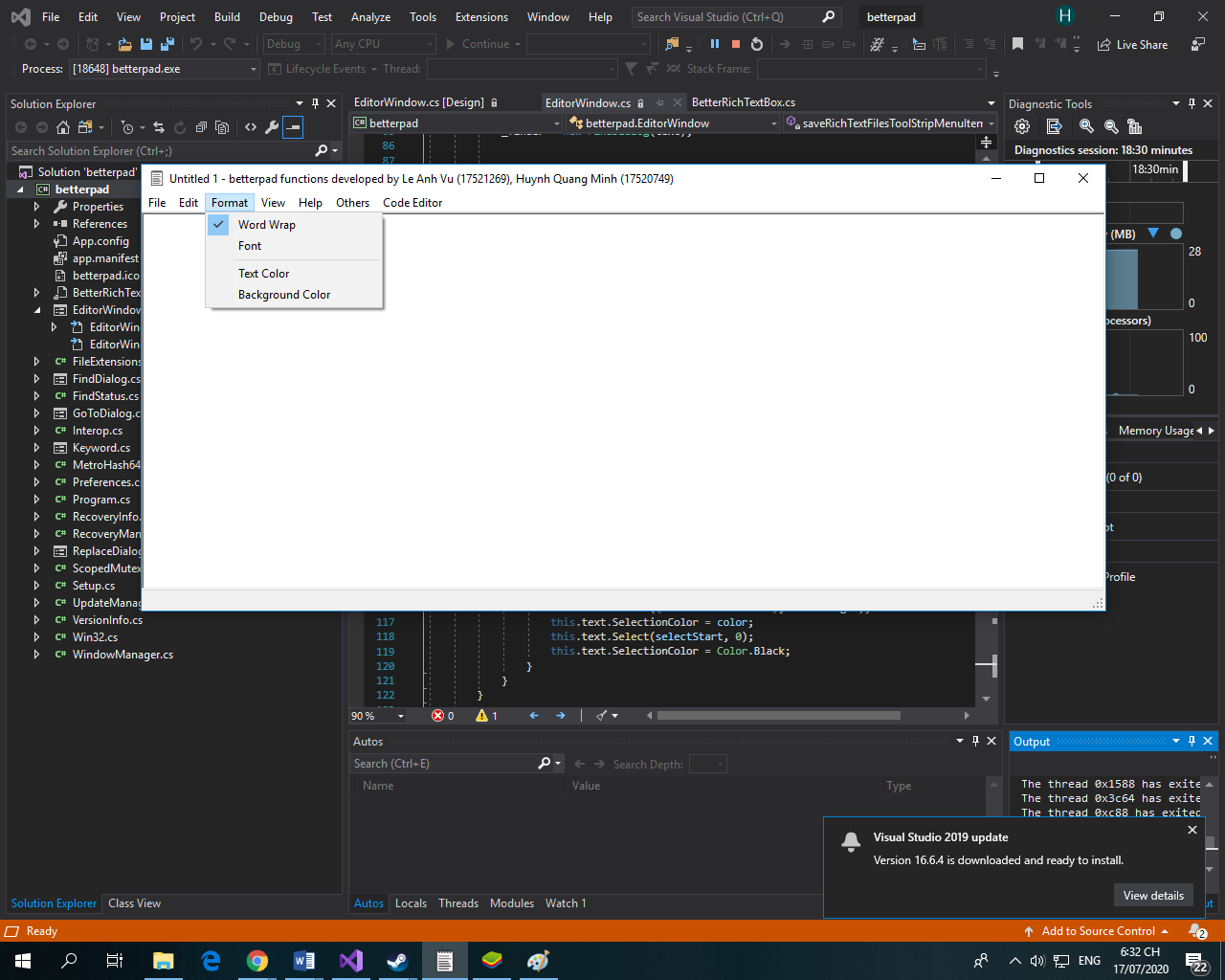
* Tạo ra các chức năng mới phù hợp với mục tiêu của môn học
* Hoàn thành tất cả các chức năng đã đưa ra
* Đưa ra các chức năng mới để cả thiện hiệu suất làm việc của người dùng

# **CHƯƠNG 2. GIAO DIỆN NGƯỜI DÙNG (UI)**

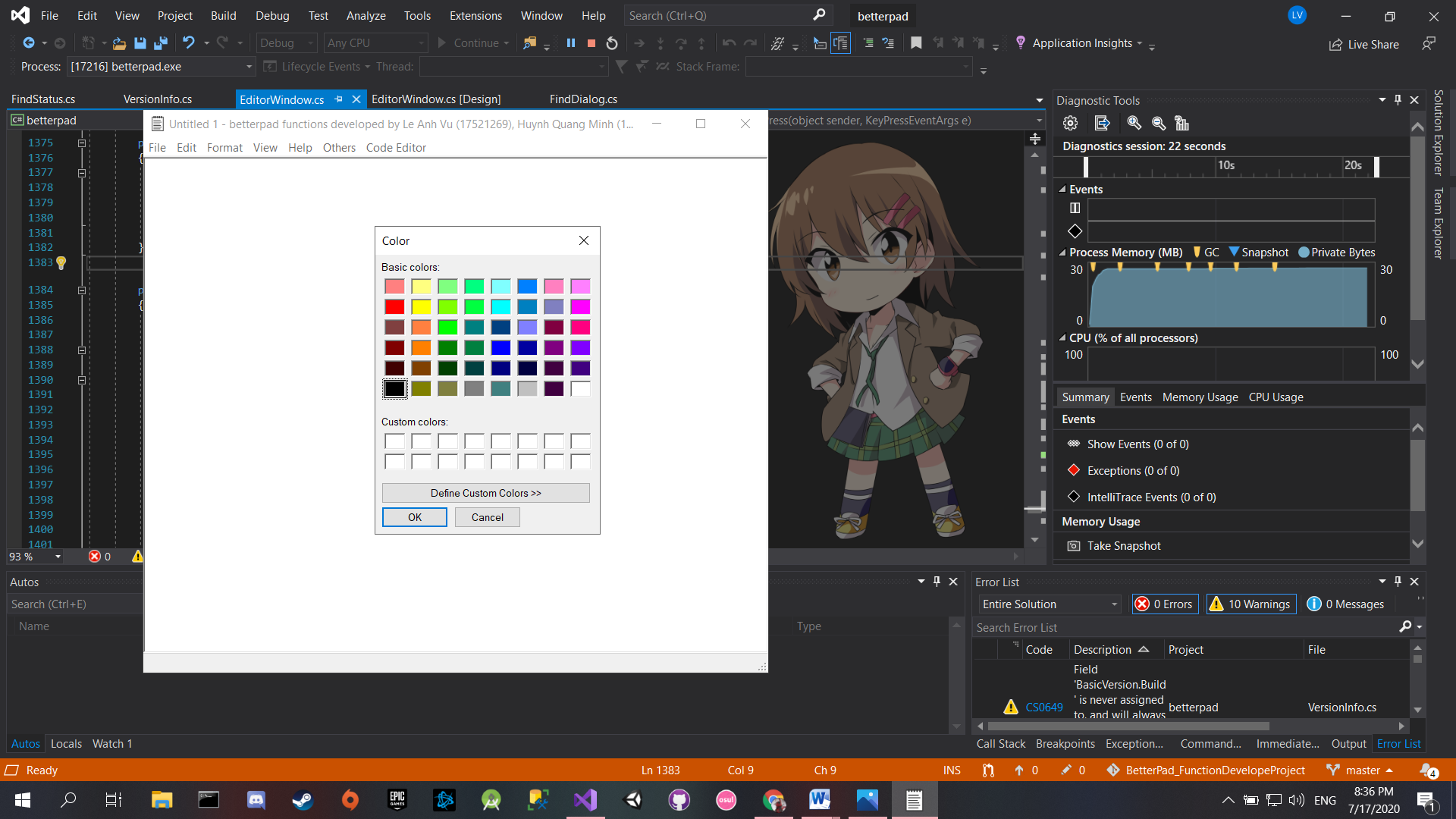


*Hình 2.1 : Giao diện chính*

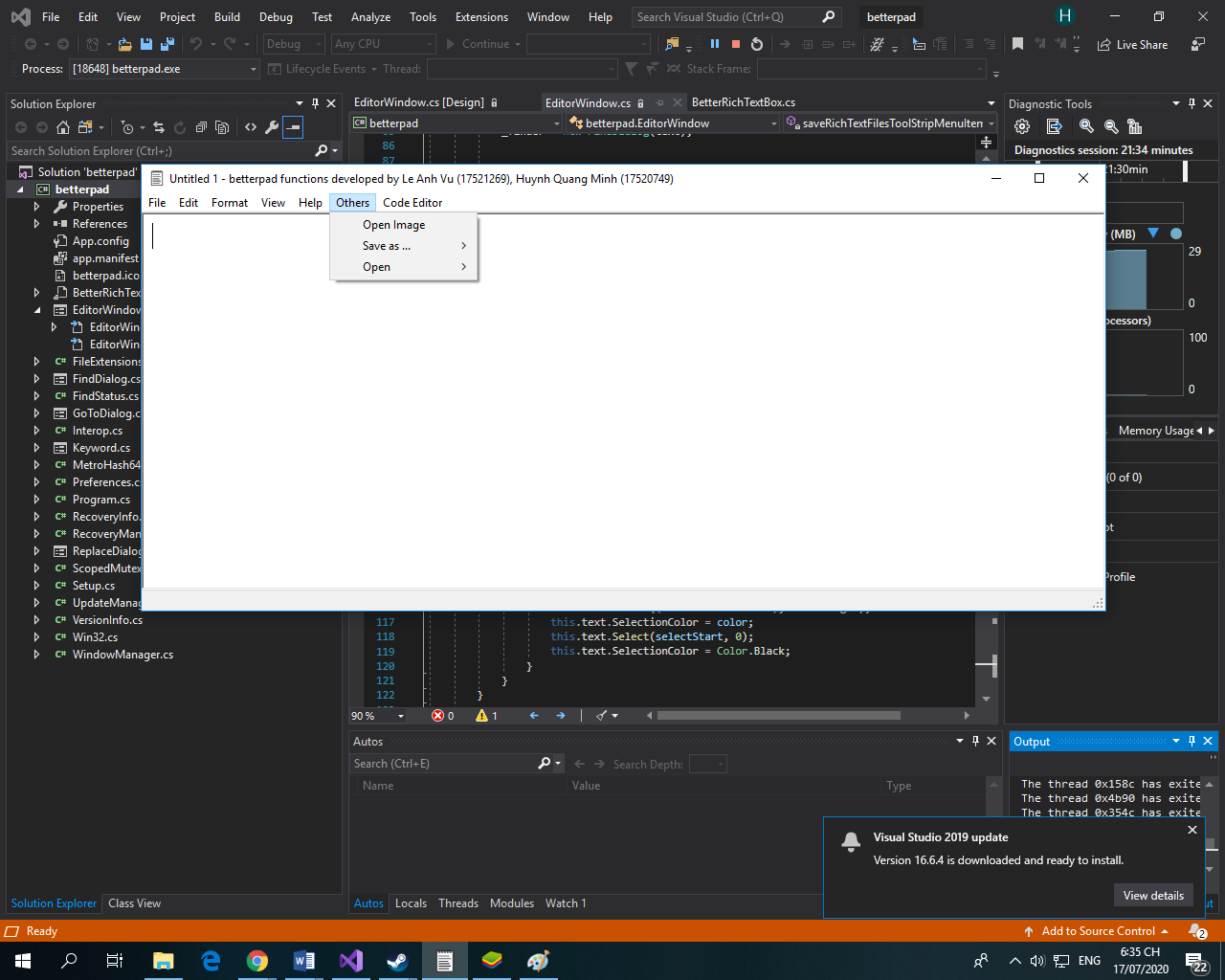
* Giao diện khi khởi động chương trình.



*Hình 2.2. Giao diện khi trỏ vào Format Tool Strip Menu*

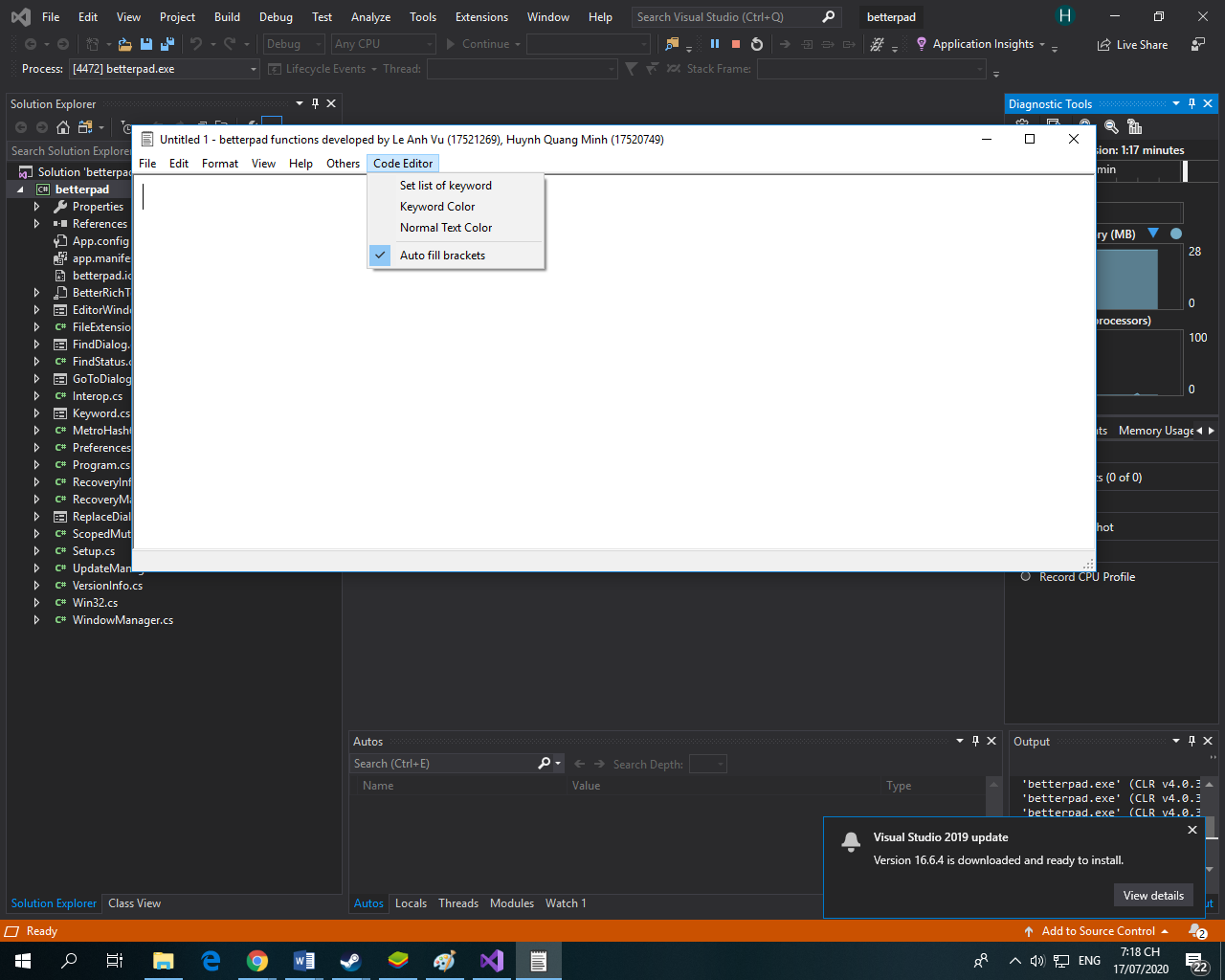


*Hình 2.3 Giao diện khi click vào Text Color/ Background Color*



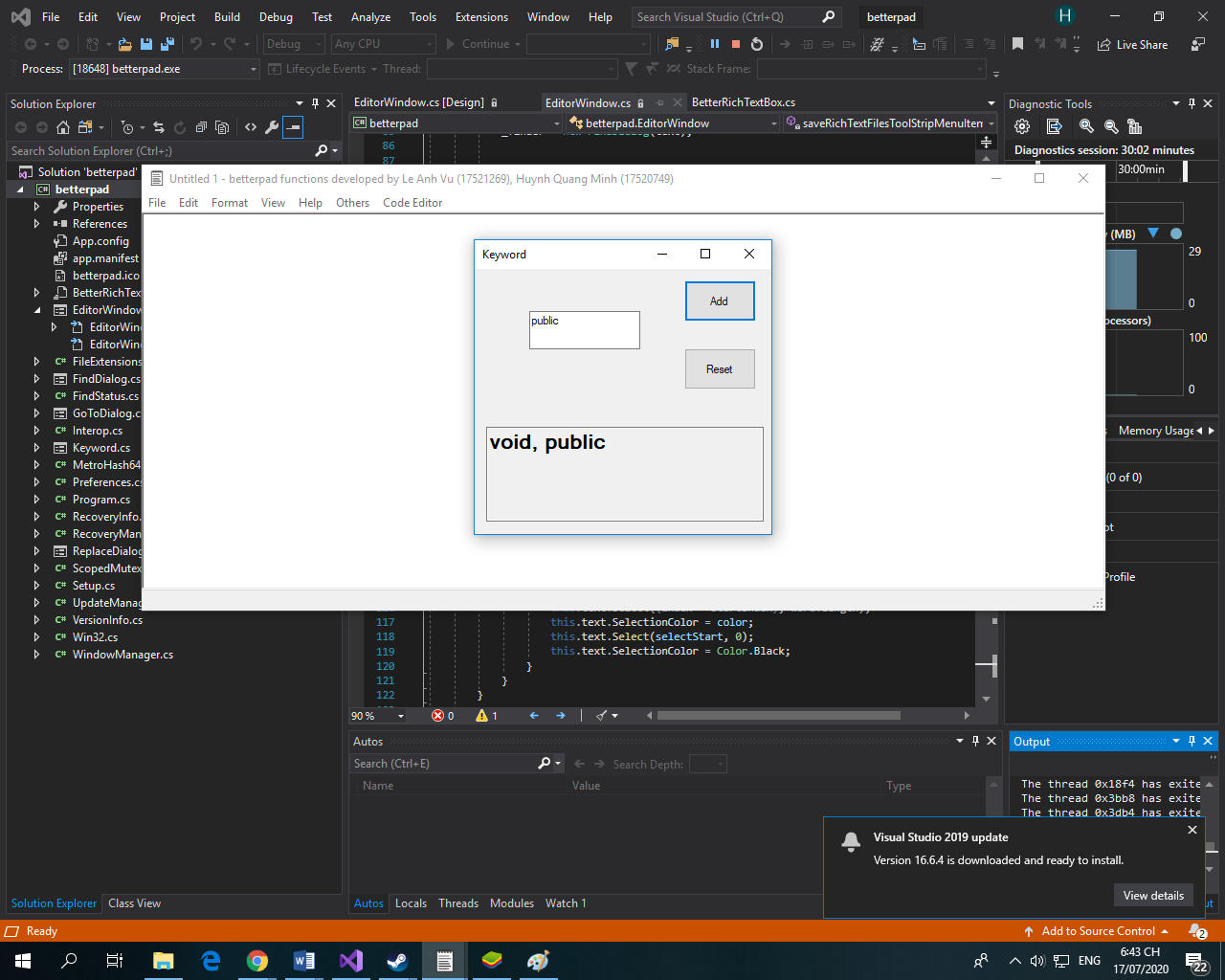
*Hình 2.4 : Mục Others dùng để hiển thị hình ảnh, lưu văn bản và hình ảnh vào file và mở file đã lưu để đọc lại*

* Mục Open Image dùng để hiển thị hình ảnh ( lấy từ ỗ đĩa cứng/ clipboard).
* Mục Save as dùng để lưu văn bản và hình ảnh vào file ( ở đây là file định dạng RTF).
* Mục Open dùng để mở file định dạng RTF đã lưu gồm văn bản và hình ảnh.



*Hình 2.5 : Mục Code Editor với các chức năng đơn giản*

* Mục Set list of keyword dùng để nhập danh sách các từ khóa để hiển thị màu riêng so với từ thường.
* Mục Keyword Color dùng để chọn màu của từ khóa ( Hiện Color Dialog như Text Color và Background Color ).
* Mục Normal Text Color dùng để chọn màu của từ thường( Hiện Color Dialog như Text Color và Background Color ).
* Mục Auto fill Brackets dùng để tự tạo các cặp ngoặc nhọn, ngoặc đơn,…, bấm lần nữa để tắt chức năng này hoặc bật lại



*Hình 2.6 : Giao diện của mục Set list of keyword*

* Gõ keyword người dùng mong muốn vào text box và nhấn nút Add để thêm vào danh sách từ khóa đặc biệt.
* Bấm nút Reset để xóa toàn bộ danh sách từ khóa đặc biệt.
* Textbox dưới cùng dùng để hiển thị những từ khóa mà người dùng đã Add vào.

# **CHƯƠNG 3. CHI TIẾT VỀ CODE**

Do Betterpad là một ứng dụng đơn giản nên không có tài liệu phân tích thiết kế mà phải xem từ source code nên chúng em sẽ thể hiện code của các chức năng. Có hai lớp quan trọng của ứng dụng là EditorWindow và BetterRichTextBox:

* EditorWindow là giao diện gồm cửa sổ và các menu
* BetterRichTextBox sẽ hiển thị nội dung của tài liệu

## **3.1. Chức năng Text Color ( Đổi màu cụm chữ đã chọn ) :**

* **EditorWindow.cs :** line 884 🡪 894

private void textColorToolStripMenuItem\_Click(object sender, EventArgs e)

{

ColorDialog colorD = new ColorDialog();

colorD.Color = text.SelectionColor;

if (colorD.ShowDialog() == System.Windows.Forms.DialogResult.OK && colorD.Color != text.SelectionColor)

{

text.SelectionColor = colorD.Color;

}

}

## **3.2. Chức năng Background Color ( Đổi màu nền của cụm chữ đã chọn ):**

* **EditorWindow.cs :** line 896 🡪 906

private void backgroundColorToolStripMenuItem\_Click(object sender, EventArgs e)

{

ColorDialog bColor = new ColorDialog();

bColor.Color = text.SelectionBackColor;

if (bColor.ShowDialog() == System.Windows.Forms.DialogResult.OK && bColor.Color != text.SelectionBackColor)

{

text.SelectionBackColor = bColor.Color;

}

}

## **3.3. Chức năng Open Image ( Mở hình ảnh từ thư mục hoặc paste từ clipboard ):**

* **EditorWindow.cs :** line 1323 🡪 1332

private void openImageToolStripMenuItem\_Click(object sender, EventArgs e)

{

OpenFileDialog openimg = new OpenFileDialog();

openimg.Filter = "Image files(\*.jpeg; \*.jpg; \*.gif; \*.png; \*.peg) | \*.jpeg; \*.jpg; \*.gif; \*.png; \*.peg";

if (openimg.ShowDialog() == DialogResult.OK)

{

createImage(openimg.FileName);

}

}

* **Chỉnh sửa hàm Paste để dán hình từ clipboard : EditorWindow :** line 576 🡪580

private void Paste()

{

text.Paste(DataFormats.GetFormat(DataFormats.Bitmap));

text.Paste(DataFormats.GetFormat(DataFormats.UnicodeText));

}

## **3.4. Chức năng Save/Load RTF files :**

* **EditorWindow.cs :** line 808 🡪 835

public void SaveAsRTF()

{

SaveFileDialog saveFile1 = new SaveFileDialog();

saveFile1.DefaultExt = "\*.rtf";

saveFile1.Filter = "RTF Files|\*.rtf";

if (saveFile1.ShowDialog() == System.Windows.Forms.DialogResult.OK &&

saveFile1.FileName.Length > 0)

{

text.SaveFile(saveFile1.FileName, RichTextBoxStreamType.RichText);

SetTitle(saveFile1.FileName);

}

}

public void LoadRTF()

{

OpenFileDialog openFile1 = new OpenFileDialog();

openFile1.DefaultExt = "\*.rtf";

openFile1.Filter = "RTF Files|\*.rtf";

if (openFile1.ShowDialog() == System.Windows.Forms.DialogResult.OK &&

openFile1.FileName.Length > 0)

{

text.LoadFile(openFile1.FileName);

SetTitle(openFile1.FileName);

}

}

* **EditorWindow.cs :** line 1339 🡪 1347

private void saveRichTextFilesToolStripMenuItem\_Click(object sender, EventArgs e)

{

SaveAsRTF();

}

private void openRichTextFilesToolStripMenuItem\_Click(object sender, EventArgs e)

{

LoadRTF();

}

## **3.5. Chức năng Set list of keyword (Đổi màu cho các keyword người dùng nhập ):**

* **Keyword.cs :**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace betterpad

{

public partial class Keyword : Form

{

public Keyword()

{

InitializeComponent();

InitializeLayout();

}

private void InitializeLayout()

{

TextBox2\_TextChanged(null, null);

}

private void Button1\_Click(object sender, EventArgs e)

{

string abc = textBox1.Text;

EditorWindow.aaa.Add(abc);

InitializeLayout();

}

private void Button2\_Click(object sender, EventArgs e)

{

EditorWindow.aaa.Clear();

InitializeLayout();

}

private void TextBox2\_TextChanged(object sender, EventArgs e)

{

string abc = String.Join(", ", EditorWindow.aaa.ToArray());

textBox2.Text = abc;

}

}

}

* **EditorWindow.cs :** line 837 🡪 863

private void text\_TextChanged(Object sender, EventArgs e)

{

redoToolStripMenuItem.Enabled = text.CanRedo;

undoToolStripMenuItem.Enabled = text.CanUndo;

for (int i = 0; i < aaa.Count; i++)

{

this.CheckKeyword(aaa[i], kwcolor, 0);

}

}

private void CheckKeyword(string word, Color color, int startIndex)

{

if (this.text.Text.Contains(word))

{

int index = -1;

int selectStart = this.text.SelectionStart;

while ((index = this.text.Text.IndexOf(word, (index + 1)))!= -1)

{

this.text.Select((index + startIndex), word.Length);

this.text.SelectionColor = color;

this.text.Select(selectStart, 0);

this.text.SelectionColor = Color.Black;

}

}

}

* **EditorWindow.cs :** line 1352 🡪 1359

public static IList<string> aaa = new List<string>();

private void setListOfKeywordToolStripMenuItem\_Click(object sender, EventArgs e)

{

Keyword a = new Keyword();

a.ShowDialog();

}

public static Color kwcolor = Color.Blue;

## **3.6. Chức năng đổi màu khác cho keyword và từ thường :**

* **EditorWindow.cs :** line 1360 🡪 1372

private void keywordColorToolStripMenuItem\_Click(object sender, EventArgs e)

{

ColorDialog colorD = new ColorDialog();

colorD.ShowDialog();

kwcolor = colorD.Color;

}

private void normalTextColorToolStripMenuItem\_Click(object sender, EventArgs e)

{

ColorDialog colorD = new ColorDialog();

colorD.ShowDialog();

text.ForeColor = colorD.Color;

}

## **3.7. Chức năng Auto fill brackets ( Tự động fill các dấu thiếu cho “(),[],{},””,\\” )**

* **EditorWindow.cs :** line 1374 🡪 1446

public static Boolean isCurslyBracesKeyPressed = false;

private void autoCorrectBracketsToolStripMenuItem\_Click(object sender, EventArgs e)

{

if (autoCorrectBracketsToolStripMenuItem.Checked == true)

{

autoCorrectBracketsToolStripMenuItem.Checked = false;

}

else autoCorrectBracketsToolStripMenuItem.Checked = true;

}

private void text\_KeyPress(object sender, KeyPressEventArgs e)

{

String s = e.KeyChar.ToString();

int sel = text.SelectionStart;

if (autoCorrectBracketsToolStripMenuItem.Checked == true)

{

switch (s)

{

case "(":

text.Text = text.Text.Insert(sel, "()");

e.Handled = true;

text.SelectionStart = sel + 1;

break;

case "{":

String t = "{}";

text.Text = text.Text.Insert(sel, t);

e.Handled = true;

text.SelectionStart = sel + t.Length - 1;

isCurslyBracesKeyPressed = true;

break;

case "[":

text.Text = text.Text.Insert(sel, "[]");

e.Handled = true;

text.SelectionStart = sel + 1;

break;

case "<":

text.Text = text.Text.Insert(sel, "<>");

e.Handled = true;

text.SelectionStart = sel + 1;

break;

case "\"":

text.Text = text.Text.Insert(sel, "\"\"");

e.Handled = true;

text.SelectionStart = sel + 1;

break;

case "'":

text.Text = text.Text.Insert(sel, "''");

e.Handled = true;

text.SelectionStart = sel + 1;

break;

}

}

}

private void text\_KeyDown(object sender, KeyEventArgs e)

{

int sel = text.SelectionStart;

if (e.KeyCode == Keys.Enter)

{

if (isCurslyBracesKeyPressed == true)

{

text.Text = text.Text.Insert(sel, "\n \n");

e.Handled = true;

text.SelectionStart = sel + " ".Length;

isCurslyBracesKeyPressed = false;

}

}

# **CHƯƠNG 4. HƯỚNG DẪN CÀI ĐẶT**

## **4.1. Hướng dẫn cài đặt :**

1. Vào github theo link : <https://github.com/zz2zz22/BetterPad_FunctionDevelopProject>
2. Pull về toàn bộ repo.
3. Chạy file betterpad.sln bằng IDE sau khi giải nén.
4. Ctrl + B để build toàn bộ project (Phải build mới có thể check ).
5. Nhấn “F5” hoặc debug

# **KẾT LUẬN**

Do BetterPad là một ứng dụng đơn giản nên không có tài liệu phân tích thiết kế chi tiết mà phải tự xem source code nên file báo cáo còn nhiều thiếu sót nhưng chúng em đã hoàn thành việc phát triển các chức năng dựa trên sự hỗ trợ nhiệt tình của giảng viên hướng dẫn. Hoàn thành đủ số các chức năng mà giảng viên hướng dẫn yêu cầu.

## **Nhận xét :**

* + Hoàn thành đúng với mục tiêu đã đề ra
  + Phần mềm còn nhiều điều cần cải thiện.

## **Những chức năng có thể phát triển trong tương lai :**

* Kiểm tra các dấu () , {} , “” . Nếu không đủ cặp thì đổi màu cho nó.
* Lưu thành các dạng file như HTML files, Binary files,…
* Kiểm tra các dấu ngoặc có đặc đúng theo quy tắc không.

Em xin chân thành cảm ơn cô Đỗ Thị Thanh Tuyền đã hỗ trợ hết mình để chúng em có thể hoàn thành đề tài này.

# **DANH SÁCH TÀI LIỆU THAM KHẢO**

1. Tham khảo lỗi với designer mode:

<https://docs.microsoft.com/en-us/dotnet/framework/winforms/controls/design-time-errors-in-the-windows-forms-designer?f1url=https%3A%2F%2Fmsdn.microsoft.com%2Fquery%2Fdev16.query%3FappId%3DDev16IDEF1%26l%3DEN-US%26k%3Dk(DTELErrorList)%3Bk(TargetFrameworkMoniker-.NETFramework%2CVersion%3Dv4.7.2)%26rd%3Dtrue>

1. <https://www.c-sharpcorner.com/article/syntax-highlighting-in-rich-textbox-control-part-1/?fbclid=IwAR13kaVBcxbSEGTw7HhGhh0yZYJ51EyNK1NiLcch0BjNpqo1gDFF9tjWnWw>
2. Tài liệu sửa lỗi được tìm chủ yếu trên stackoverflow : <https://stackoverflow.com/>
3. Trang tham khảo tài liệu số : <https://www.howkteam.vn/>
4. <https://www.codeproject.com/Articles/27431/Writing-Your-Own-RTF-Converter>
5. <https://docs.microsoft.com/en-us/dotnet/api/system.windows.forms.richtextbox.savefile?view=netcore-3.1>

# **PHỤ LỤC**

## **Chart tỷ lệ công việc của từng thành viên**

## **Link source code :**

* <https://github.com/zz2zz22/BetterPad_FunctionDevelopProject>
* <https://sourceforge.net/projects/betterpadfunctiondevelop/>