



TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN
MÔN: **ĐỒ HỌA MÁY TÍNH**
LỚP: 16CNTN

ĐỒ ÁN 1

ĐỒ HỌA 2D

1612829 Nguyễn Quốc Vương

1612842 Lê Thành Công

GVLT: Trần Thái Sơn

GVTH: Võ Hoài Việt

Mục lục

I. Tổng quan	3
II. Nội dung	4
1. Yêu cầu 1	4
2. Yêu cầu 2	5
3. Yêu cầu 3	5
4. Yêu cầu 4	6
5. Yêu cầu 5	6
III. Giao diện chương trình	7
IV. Hướng dẫn sử dụng	10
V. Nguồn tham khảo	12

I. Tổng quan

Thông tin thành viên

STT	Mã số sinh viên	Họ và tên
1	1612829	Nguyễn Quốc Vương
2	1612842	Lê Thành Công

Phân công công việc

STT	Công việc	Yêu cầu	Phân công	Mức độ hoàn thành
1	Vẽ đường thẳng, hình chữ nhật, hình bình hành, đa giác, đường gấp khúc	1	Công	100%
2	Vẽ đường tròn, cung tròn, ellipse, cung ellipse, parabol, hyperbol	1	Công	100%
3	Ký tự, đường cong Bezier, xóa một phần bản vẽ	1	Vương	100%
4	Lưu trữ/đọc bản vẽ xuống đĩa bằng Serialize/Deserialize XML	2	Công	100%
5	Chọn đối tượng	3	Vương	100%
6	Thay đổi màu, nét vẽ, kiểu tô, font chữ, độ dày nét vẽ	3	Vương	100%
7	Phép quay	4	Công	100%
8	Tịnh tiến	4	Công	100%
9	Tỉ lệ	4	Công	100%
10	Thay đổi thứ tự trên dưới	5	Vương	100%
11	Chức năng copy các đối tượng (...)	5	Vương	100%
12	Thao tác trên các file ảnh bitmap, jpg,...	5	Công	100%
13	Chức năng Undo/Redo	5	Vương	100%
14	Tô màu đối tượng với mẫu tô bất kỳ	5	Vương	100%
15	Cho phép resize lại kích thước bản vẽ	5	Vương	100%
16	Report	-	Công	100%

Mức độ hoàn thành của đồ án

STT	Yêu cầu đồ án		Mức độ hoàn thành
1	Yêu cầu 1		100%
2	Yêu cầu 2		100%
3	Yêu cầu 3		100%
4	Yêu cầu 4		100%
5	Yêu cầu 5	Thay đổi thứ tự trên dưới	100%
		Chức năng copy các đối tượng (...)	100%
		Thao tác trên các file ảnh bitmap, jpg,...	100%
		Cho phép vẽ các đường biên có độ dày >1	100%
		Chức năng Undo/Redo	100%

		Tô màu đối tượng với màu tô bất kỳ	100%
		Cho phép resize lại kích thước bản vẽ	100%

II. Nội dung

1. Yêu cầu 1

- Vẽ đường thẳng chủ yếu sử dụng hàm DrawLine() từ thư viện Graphics điểm đầu lúc MouseDown và điểm cuối lúc MouseUp. Tham số truyền vào gồm điểm đầu, điểm cuối và thuộc tính Pen p, qui định màu sắc, kích thước, loại nét vẽ.
- Hình chữ nhật cũng sử dụng điểm đầu, điểm cuối làm góc, dùng hàm DrawRectangle để vẽ, tuy nhiên hình chữ nhật có thêm thuộc tính Brush fill tùy chỉnh việc tô màu, màu sắc, kiểu tô
- Hình bình hành sử dụng 4 điểm tạo thành các đường thẳng song song, sử dụng hàm DrawPolygon để vẽ từ 4 điểm đó.
- Đa giác và đường gấp khúc vẽ gần giống nhau, về cơ bản lưu lại các điểm đã chọn trên màn hình sau dùng hàm DrawPolygon để vẽ đối với đa giác, DrawLines để vẽ đối với đường gấp khúc.
- Đường tròn và cung tròn, đường ellipse, cung ellipse cần quy định điểm bắt đầu vẽ từ góc trên bên trái, kích thước dài và rộng của hình chữ sau đó sử dụng hàm DrawEllipse, đối với đường tròn thì kích thước dài rộng bằng nhau. Đối với cung tròn hay cung ellipse có thêm thuộc tính góc bắt đầu và góc quét. Góc quét ở đây được định nghĩa bằng cách quy đổi số pixel tính bằng khoảng cách hoành độ giữa điểm đầu và điểm cuối với góc của cung.
- Parabol và Hyperbol được vẽ bằng cách giới hạn phạm vi vẽ trong hình chữ nhật qui định bởi điểm đầu và điểm cuối. Dùng phương trình Parabol: $y = \frac{-5}{d}(x - x_0)^2 + y_0$. d là khoảng cách giữa điểm cuối và đỉnh của Parabol, càng xa đỉnh thì hệ số 5/d càng nhỏ, Parabol càng mở rộng. Dấu – mục đích lật ngược Parabol lại do hệ trục trong bản vẽ có y hướng xuống là chiều +. Tính được x, y theo phương trình, ta dùng hàm DrawLines để hoàn thành Parabol. Hyperbol được vẽ tương tự, nhưng chia ra 2 nhánh: left và right. Dùng công thức $\frac{(x-x_0)^2}{a^2} - \frac{(y-y_0)^2}{b^2} = 1$. Sau đó biểu diễn x theo y và cho y chạy từ y của điểm Start đến y của End, tính x theo y. Tìm được tập hợp các điểm đó dùng DrawLines để hoàn thành Hyperbol.
- Ký tự: dùng hàm DrawString truyền vào nội dung, điểm bắt đầu vẽ, Font chữ, màu sắc (brush). Click chọn 1 điểm trên màn hình và nhập nội dung, sau đó text được vẽ với màu sắc và Font đã chọn.
- Đường cong Bezier: chọn lần lượt 4 điểm để vẽ, dùng hàm DrawBezier của thư viện Graphics để vẽ đường cong từ 4 điểm này
- Xóa một đối tượng của bản vẽ được thực hiện bằng cách xóa phần tử đã chọn từ List<Shape> đã lưu khi vẽ hình trước đó.
- Xóa một phần bản vẽ bất kì, đưa tất cả hình ảnh đã vẽ về file bitmap sau đó dùng brush xóa trên ảnh bitmap đó, set ảnh bitmap đó về background. Như vậy ta có thể xóa chi tiết bất kì trên bản vẽ mà sau đó vẽ tiếp được trên nền background đó

2. Yêu cầu 2

- Sử dụng kỹ thuật Serialize XML và Deserialize XML
- Trước hết, ta đã lưu trữ các đối tượng đã vẽ bằng List<Shape> với Shape là abstract class, đại diện cho các đối tượng con: đường thẳng, hình chữ nhật, đường tròn, ellipse,...
- Xây dựng class XMLShape: lưu trữ hình background, kích thước khung vẽ và danh sách các hình

```
public class XMLShapes
{
    public bool isBackGroundImage;
    public ImageFile imageFile;

    public int w;
    public int h;
    public IColor bgColor;

    [XmlElement(typeof(MyArcCircle))]
    [XmlElement(typeof(MyArcEllipse))]
    [XmlElement(typeof(MyBezier))]
    [XmlElement(typeof(MyCircle))]
    [XmlElement(typeof(MyEllipse))]
    [XmlElement(typeof(MyHyperbol))]
    [XmlElement(typeof(MyParabol))]
    [XmlElement(typeof(MyParallelogram))]
    [XmlElement(typeof(MyPolygon))]
    [XmlElement(typeof(MyPolyline))]
    [XmlElement(typeof(MyRectangle))]
    [XmlElement(typeof(MyText))]
    [XmlElement(typeof(StraightLine))]
    public List<Shape> listShape;
}
```

- Với cấu trúc đã được định sẵn, thực hiện Serialize XML để tiến hành đưa dữ liệu đang có thành định dạng XML:

```
XmlSerializer s = new XmlSerializer(typeof(XMLShapes));
s.Serialize(w, saveShape); //w kiểu StreamWriter, saveShape kiểu XMLShapes
```

- Tiếp theo để đọc lại dữ liệu đó dùng Deserialize
saveShape = (XMLShapes)xs.Deserialize(xr); //xs kiểu XMLSerializer, xr kiểu XMLTextReader
- Lúc này saveShape đã có toàn bộ dữ liệu được nạp từ file XML

3. Yêu cầu 3

- Màu sắc sử dụng ColorDialog có sẵn để chọn màu, truyền vào khi khởi tạo nét vẽ Pen
- Loại nét vẽ sử dụng DashStyle load lên comboBox, khi vẽ đối tượng ta chỉ cần thay đổi giá trị DashStyle của nét vẽ Pen
- Kiểu tô màu: SolidBrush: SolidColor và Transparent, HatchBrush cho phép tô theo mẫu có sẵn và TextureBrush tô màu theo ảnh bitmap bất kỳ
- Font chữ được tùy chỉnh từ hộp thoại FontDialog có sẵn
- Ta lưu trữ đối tượng bằng List<Shape>, thuộc tính prevShape xác định chỉ số đối tượng trong list đang được chọn.

- Để chọn được đối tượng, ta xét vị trí chuột hiện tại, tìm trọng tâm của các đối tượng còn lại, tìm đối tượng có khoảng cách từ con trỏ chuột đến trọng tâm của nó là ngắn nhất. Khi đó gán prevShape = chỉ số của đối tượng trong listShape. Khi chọn được đối tượng, ta dễ dàng thay đổi các thuộc tính trên

4. Yêu cầu 4

- Phép tịnh tiến:
 - o Đối với đối tượng trên: ta lưu trữ tập điểm của đối tượng vì vậy khi tịnh tiến ta tịnh tiến các điểm theo công thức:

$$x' = x + \text{delta}X$$

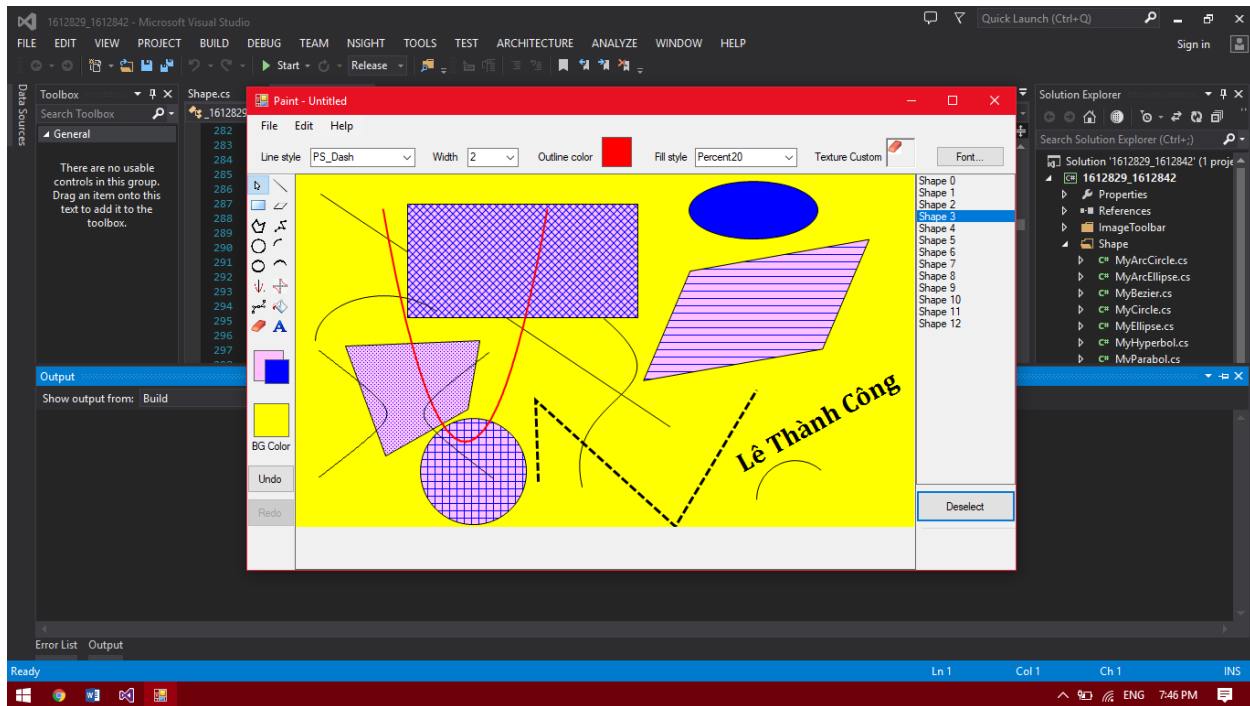
$$y' = y + \text{delta}Y$$
- Phép quay:
 - o Sử dụng công thức $\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x_r & y_r \end{bmatrix} + \begin{bmatrix} x - x_r & y - y_r \end{bmatrix} + \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}$
 Với (x_r, y_r) là tâm quay
- Phép tỉ lệ:
 - o Công thức $\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} + \begin{bmatrix} x_f & y_f \end{bmatrix} \begin{bmatrix} 1 - S_x & 0 \\ 0 & 1 - S_y \end{bmatrix}$
 Với (x_f, y_f) là điểm cố định trong phép tỉ lệ, S_x là tỉ lệ phương Ox, S_y là tỉ lệ theo phương Oy

5. Yêu cầu 5

- Thay đổi thứ tự trên dưới:
 - o Trước đó, ta đã lưu trữ danh sách các đối tượng thông qua List<Shape>, vì vậy chỉ cần hoán đổi vị trí của phần tử được chọn với các vị trí còn lại
 - o Bring to front: đối tượng được chọn chuyển về cuối list
 - o Bring forward: đối tượng được chọn tăng lên 1 chỉ số
 - o Bring to back: đối tượng được chọn chuyển về đầu list
 - o Bring backward: đối tượng được chọn giảm 1 chỉ số
- Chức năng copy các đối tượng:
 - o Copy: Lưu lại đối tượng vào biến tạm
 - o Cut: Lưu lại đối tượng đồng thời xóa đối tượng đó
 - o Paste: Thêm lại đối tượng đã lưu vào list và vẽ lại lên bản vẽ
 - o Copy to clipboard: chọn đối tượng, thực hiện vẽ đối tượng vào file bitmap và dùng hàm Clipboard.SetImage(bitmap) để lưu đối tượng vào clipboard
- Thao tác trên các file ảnh bitmap, jpg,...: dùng các hàm có sẵn của Bitmap để save/open file Bitmap. Khi load bitmap, ta lưu làm background nên có thể vẽ đè lên file ảnh.
- Chức năng Undo/Redo: một cách đơn giản, ta lưu lại trạng thái bản vẽ vào file xml trước khi thực hiện thao tác khác, sau đó muốn quay lại chỉ cần load lại file xml đó
- Tô màu đối tượng với màu tô bất kỳ: khởi tạo TextureBrush truyền vào file bitmap, sau đó tô màu các hình chỉ cần fill brush vừa tạo.
- Cho phép resize lại kích thước bản vẽ: truyền vào kích thước, sau đó chỉnh lại pictureBox.ClientSize cùng với size của file bitmap đang vẽ.

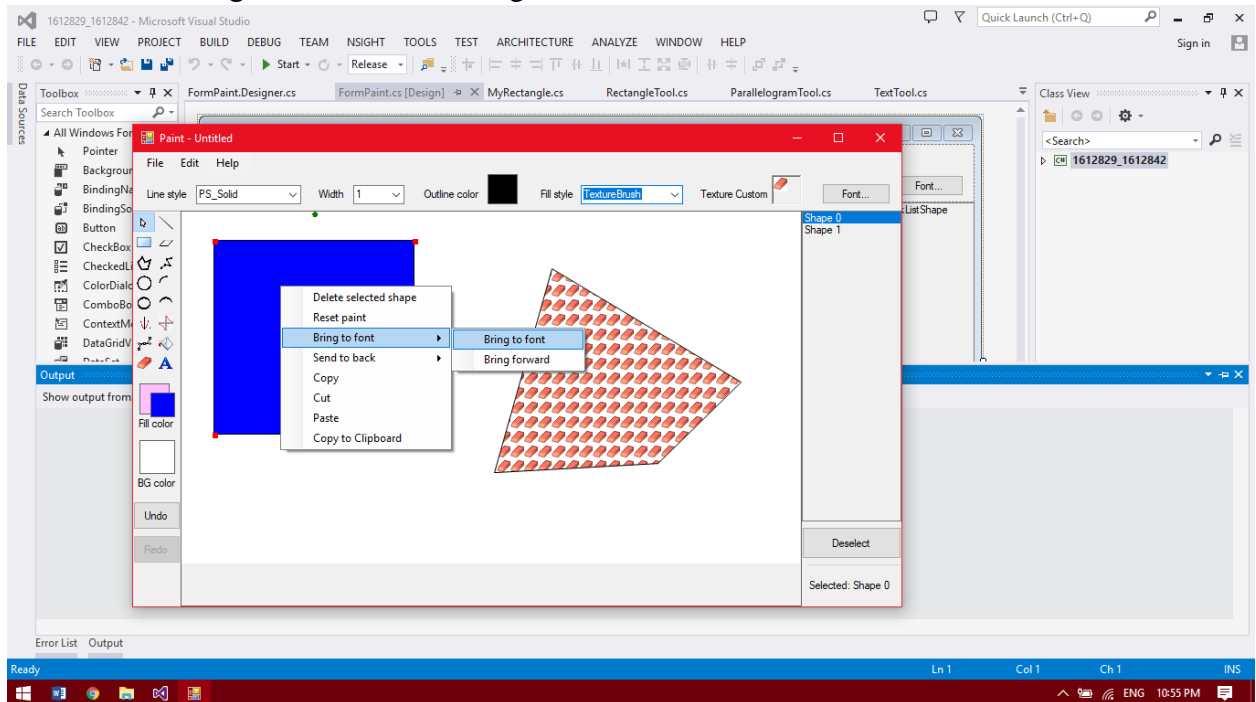
III. Giao diện chương trình

- Giao diện chương trình ở hình dưới đây:

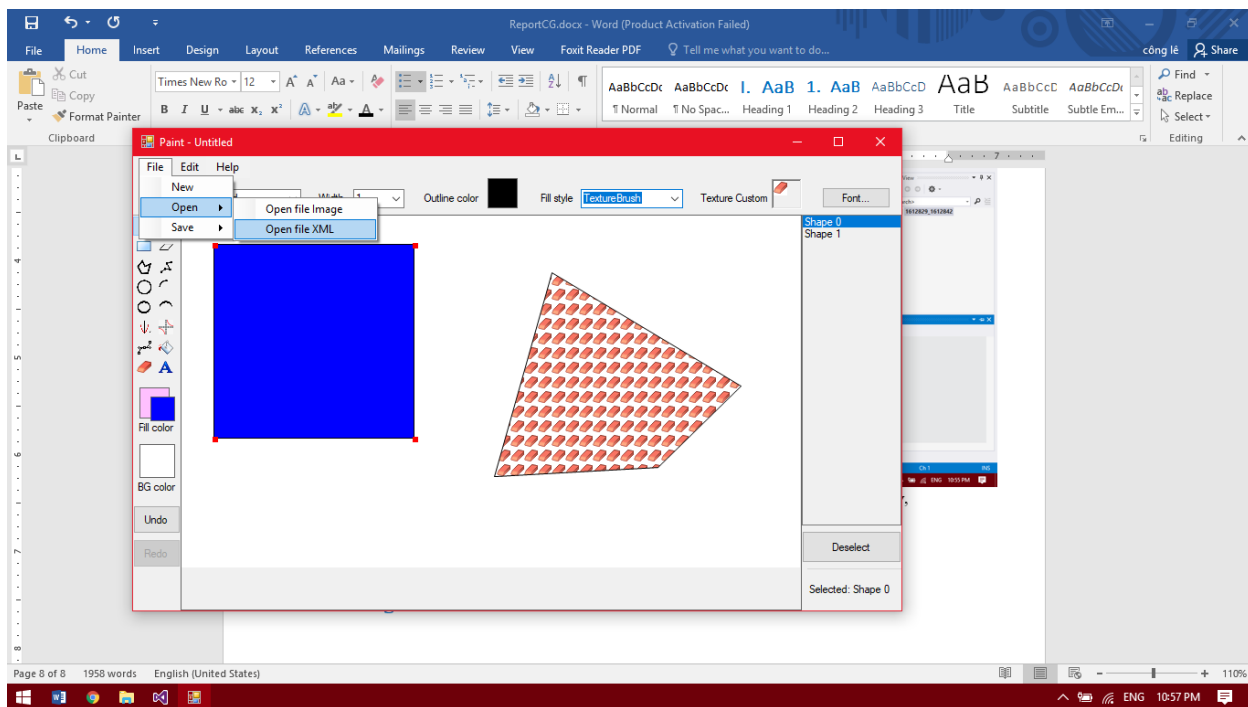


- Hộp thoại ComboBox LineStyle: chọn loại nét vẽ: PS_Solid, PS_Dash, PS_DashDot,..
- Width: chỉnh kích thước nét vẽ
- Outline color: chọn màu nét vẽ
- Fill style: kiểu tô màu gồm có lần lượt là: Solid, Transparent, TextureImage, Các mẫu tô có sẵn. Để TextureImage có hiệu lực cần nhấn chọn vào hình vuông bên cạnh để chọn hình bitmap trước để có mẫu tô
- Font là nút để thay đổi font sau khi chọn đối tượng text đã vẽ lên màn hình
- 16 nút trên thanh công cụ bên trái phần mềm:
 - o Select tool: dùng để chọn đối tượng, sau khi chọn xong, có thể di chuyển, resize, xoay đối tượng
 - o Vẽ đường thẳng
 - o Vẽ hình chữ nhật
 - o Vẽ hình bình hành
 - o Đa giác
 - o Vẽ đường gấp khúc
 - o Hình tròn
 - o Cung tròn
 - o Ellipse
 - o Cung ellipse
 - o Parabol
 - o Hyperbol
 - o Bezier

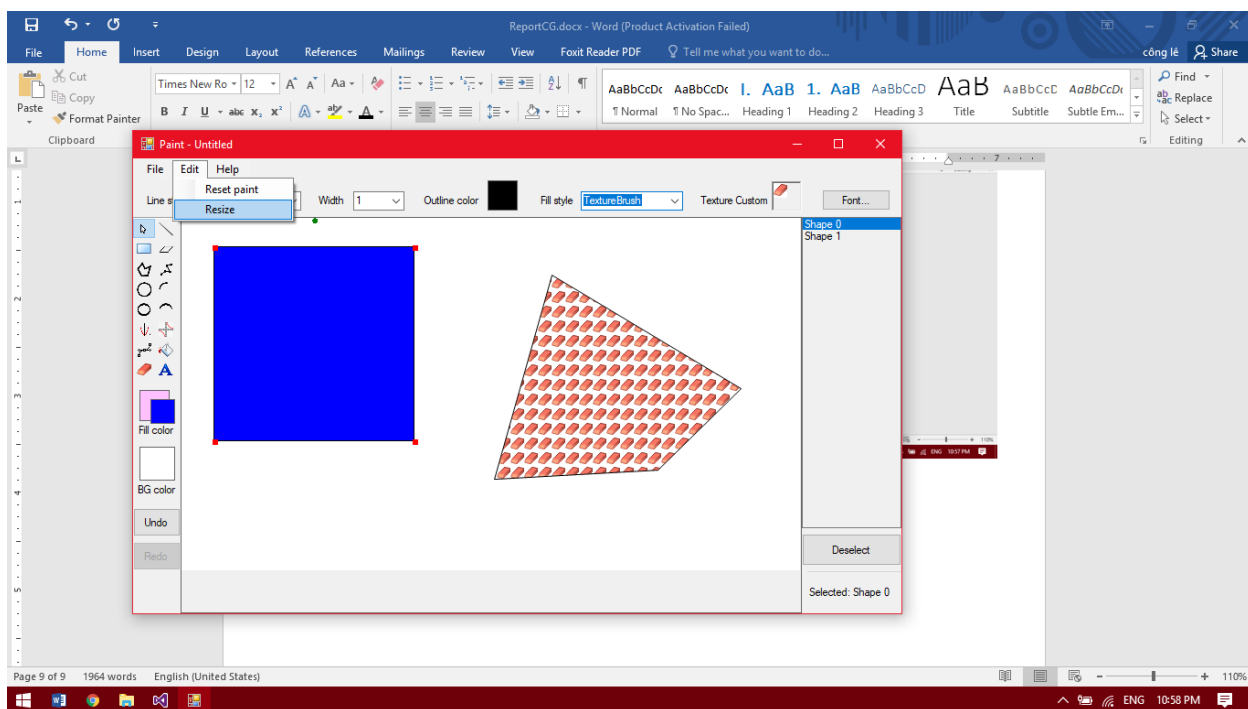
- Fill tool
- Eraser tool
- Text
- Tiếp theo là Fill color, gồm màu primary là màu chính là hình vuông nằm trên, secondary là màu phụ nằm dưới. 2 loại màu này kết hợp để tô hatchbrush. Tô màu Solid bình thường chủ yếu dùng primary
- BG Color: màu background
- Danh sách các đối tượng đã vẽ nằm phía bên phải, danh sách đối tượng được vẽ theo thứ tự trước – sau.
- Nút Deselect dùng để bỏ chọn đối tượng.



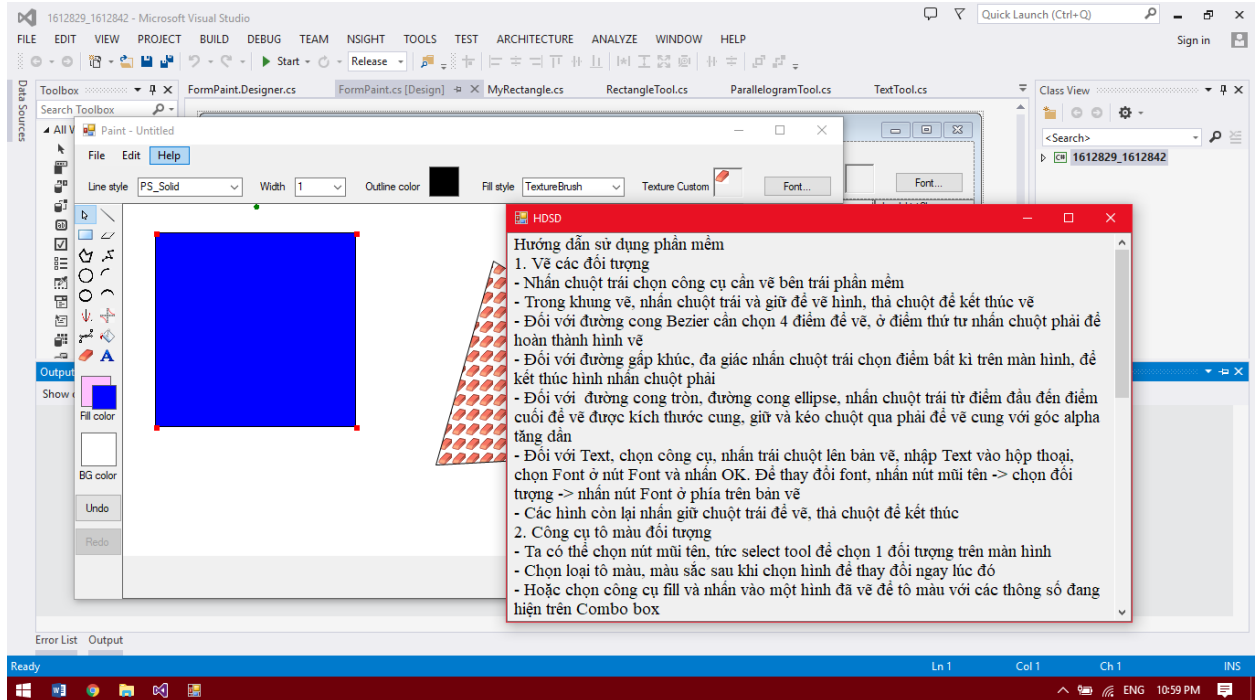
- Sau khi chọn đối tượng, nhấn chuột phải để dùng các chức năng như: Delete, Copy, Paste, Copy to Clipboard...



Các chức năng ở nút File



Các chức năng Reset và Resize ở nút Edit



Chức năng hướng dẫn sử dụng ở nút Help

IV. Hướng dẫn sử dụng

1. Vẽ các đối tượng

- Nhấn chuột trái chọn công cụ cần vẽ bên trái phần mềm
- Trong khung vẽ, nhấn chuột trái và giữ để vẽ hình, thả chuột để kết thúc vẽ
- Đối với đường cong Bezier cần chọn 4 điểm để vẽ, ở điểm thứ tư nhấn chuột phải để hoàn thành hình vẽ
- Đối với đường gấp khúc, đa giác nhấn chuột trái chọn điểm bất kì trên màn hình, để kết thúc hình nhấn chuột phải
- Đối với đường cong tròn, đường cong ellipse, nhấn chuột trái từ điểm đầu đến điểm cuối để vẽ được kích thước cung, giữ và kéo chuột qua phải để vẽ cung với góc alpha tăng dần
- Đối với Text, chọn công cụ, nhấn trái chuột lên bản vẽ, nhập Text vào hộp thoại, chọn Font ở nút Font và nhấn OK. Để thay đổi font, nhấn nút mũi tên -> chọn đối tượng -> nhấn nút Font ở phía trên bản vẽ
- Các hình còn lại nhấn giữ chuột trái để vẽ, thả chuột để kết thúc

2. Công cụ tô màu đối tượng

- Ta có thể chọn nút mũi tên, tức select tool để chọn 1 đối tượng trên màn hình
- Chọn loại tô màu, màu sắc sau khi chọn hình để thay đổi ngay lúc đó
- Hoặc chọn công cụ fill và nhấn vào một hình đã vẽ để tô màu với các thông số đang hiện trên Combo box

3. Di chuyển, chỉnh kích thước, xoay đối tượng

- Chọn công cụ mũi tên Select tool
- Chọn hình bằng chuột trái

- Đưa chuột vào vị trí trung tâm hình để xuất hiện con trỏ mũi tên 4 hướng sau đó nhấn giữ và kéo đối tượng di chuyển theo chuột
- Đưa chuột vào các điểm màu đỏ để xuất hiện con trỏ hình dấu cộng và nhấn giữ để kéo giãn kích thước
- Đưa chuột vào điểm màu xanh lá và kéo để xoay đối tượng

4. Lưu trữ bản vẽ

- Lưu trữ và mở file dưới dạng Bitmap: File -> Save/Open -> Bitmap
- Lưu trữ và mở file dưới dạng XML -> có thể chỉnh sửa được đối tượng

5. Xóa bất kì

- Tính năng cục tẩy trên thanh công cụ giúp xóa 1 chi tiết bất kì nhưng sẽ ko lưu lại đối tượng nữa

6. Danh sách đối tượng

- Các Shape 0, Shape 1 hiện bên hộp thoại bên phải phần mềm có tác dụng hiển thị tên và thứ tự đối tượng được vẽ trên màn hình

7. Cut, Copy, Delete

- Sau khi chọn đối tượng, nhấn chuột phải để copy, cut, delete đối tượng

8. Bring to font, send to back

- Sau khi chọn đối tượng, nhấn chuột phải -> chọn Bring to font hoặc send to back

9. Undo, Redo

- Nút Undo, Redo có tác dụng lấy lại trạng thái đang vẽ trước đó hoặc sau đó

10. Chức năng chỉnh lại kích thước bản vẽ

- File -> New để tạo mới bản vẽ
- Edit -> Resize để chỉnh lại kích thước bản vẽ mà không reset lại hình vẽ
- Reset paint: Xóa hết các đối tượng đã vẽ
- Nút Deselect: Bỏ chọn đối tượng

Lưu ý: Để chọn được đối tượng phải nhấn nút mũi tên Select tool ở thanh công cụ bên trái

V. Nguồn tham khảo

<http://csharpHelper.com/blog/2014/11/draw-a-conic-section-from-its-polynomial-equation-in-c/>

<https://docs.microsoft.com/en-us/dotnet/framework/winforms/advanced/how-to-fill-a-shape-with-a-hatch-pattern>

<https://docs.microsoft.com/en-us/dotnet/framework/winforms/advanced/how-to-fill-a-shape-with-an-image-texture>

<https://github.com/mitkonikov/Rotating-Plane/blob/master/RotatingPlane2D/RotatingPlane/Form1.cs>

<http://csharpHelper.com/blog/2018/02/copy-files-to-the-clipboard-in-c/>

<https://github.com/malharthi/Paint>

<https://docs.microsoft.com/en-us/dotnet/standard/serialization/examples-of-xml-serialization>

<https://www.mastercode.vn/blog/web-development/bai-9-serialization-nen-tang-lap-trinh-c.57>

Và Slide bài giảng lý thuyết.