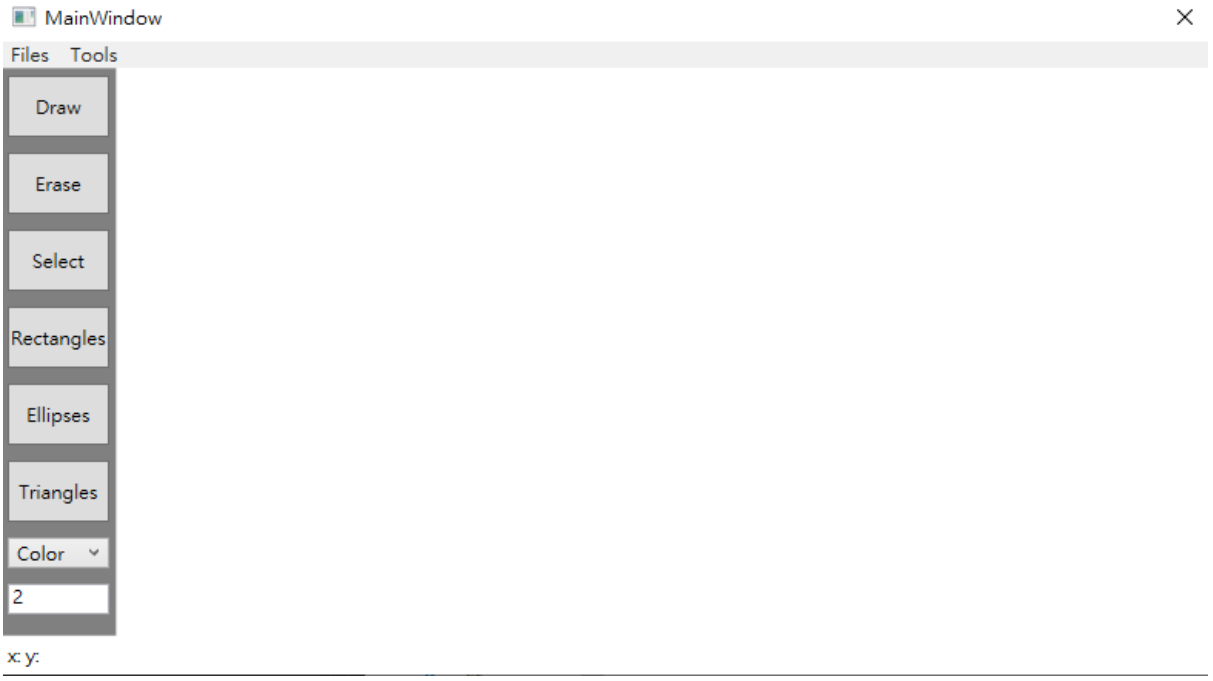
1. Purpose :

This application is a basic drawing application in WPF. It allows users to draw rectangles, ellipses, and triangles on a canvas. Users can also erase shapes and select them for movement. The application lets users load and save PNG images.

2. UI :

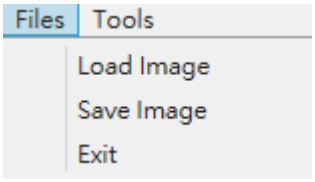
Users are able to select features in the upper menu and tool buttons on the left. And draw shapes in canvas on the right white side.



3. Features :

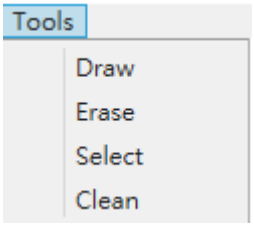
Menu includes two tabs, Files and Tools.

3.1. Files :



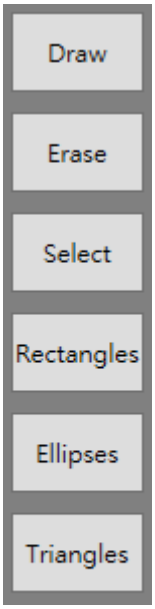
Function name	Descriptions
Load Image	Load image PNG file to canvas.
Save Image	Save canvas to PNG file.
Exit	Exit application.

3.2. Tools :



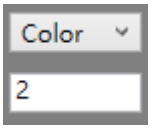
Function name	Descriptions
Draw	It is the same as the draw button.
Erase	It is the same as the erase button.
Select	It is the same as the select button.
Clean	Clean contexts in canvas.

3.3. Buttons :



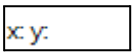
Function name	Descriptions
Draw	Draw with shape buttons selected.
Erase	Delete selected shapes.
Select	Drag and drop selected shapes to target position.
Rectangles	Draw rectangle. Only available with Draw mode.
Ellipses	Draw ellipse. Only available with Draw mode.
Triangles	Draw triangle. Only available with Draw mode.

3.4. Color and thickness



Function name	Descriptions
Color	Provide black, red and green for use.
Thickness	Brush thickness. Only 1 to 10 accepted.

3.5. Coordinate bar :



Function name	Descriptions
x: y:	Display mouse press and release position when drawing shapes.

4. Design architecture :

This application is designed with Winform for the first draft. The purpose is verification of drawing and other features are workable. And the next is modified to fulfill MVVM architecture.

4.1. Main Components:

View	The graphical user interface (GUI) is defined in XAML (Extensible Application Markup Language) and is responsible for presenting the user interface elements, including menus, buttons, and the canvas.
ViewModel	The ViewModel serves as an intermediary between the View and the Model. It encapsulates the application's logic and exposes properties and commands for data binding.
Model	The Model represents the core data and functionality of the application. In this context, it includes shapes, their properties (e.g., color, thickness), and the canvas itself.

4.2. ViewModel:

- The ViewModel, represented by the ViewModel class, implements the INotifyPropertyChanged interface to facilitate data binding between the View and ViewModel.
- It contains properties such as Color and Thickness, which control the drawing properties of shapes on the canvas.
- The DrawCommand is an implementation of the ICommand interface, enabling the execution of drawing actions.

4.3. CanvasShape Class:

The CanvasShape class represents a shape that can be drawn on the canvas. It encapsulates a WPF Shape object.

4.4. Main Window (View):

- The main window, defined in XAML, represents the application's user interface.
- It includes menus for file operations (load and save), tools (draw, erase, select, clean), and shape selection (rectangles, ellipses, triangles).
- The canvas area (DrawAreaCanvas) is where shapes are drawn and manipulated.

4.5. Drawing Logic:

- Users can draw rectangles, ellipses, and triangles by selecting a shape and then clicking and dragging on the canvas.
- Erasing allows users to remove shapes from the canvas.
- Selecting enables users to modify the position and size of existing shapes.
- The drawing properties (color and thickness) are controlled through the ViewModel.

4.6. File Operations:

- Users can load existing PNG images into the canvas or save the current canvas as a PNG image file.

4.7. Data Binding:

- Data binding is used extensively to connect the ViewModel properties and commands to the View elements, ensuring synchronization between the UI and application logic.

4.8. Commands:

- Commands, such as the DrawCommand, are implemented to handle user interactions and execute corresponding actions.

4.9. Error Handling:

- The application includes error handling to notify users of any issues, such as invalid input or file save errors.

4.10. Maintainability and Extensibility:

- The use of the MVVM pattern and separation of concerns make the application modular and maintainable. Adding new shapes or features can be done with minimal impact on existing code.

5. Known issues and next phase:

Title	Descriptions
Color menu can not show the title.	Color menu should show the title ”Color” when startup.
Thickness needs to click other items for taking effect.	Thickness can not take effect after typing immediately. Need to leave focus to trigger event for this item.
Buttons besides “Draw” need to add binding for command event.	Only draw button binding to Model. Need to add functions for others.