Summary Report

This summary report explain how to develop a simple Diagram application using WPF that allows the user to draw diagrams using the available shapes on a canvas, select individual shapes and modify their properties, select several shapes or use a custom selection mode before modifying the shapes, and save and load up a diagram from a previously saved one.

So let proceed as follows by:

First creating new WPF project in Visual Studio 2017 adding a Canvas also known as my MyDesignerCanvas, toolbox, toolbar and Canvas Item

1. Adding Color Picker to Toolbar by installing Xceed.wpf.Toolkit library and adding the link to window object:

xmlns:xctk="http://schemas.xceed.com/wpf/xaml/toolkit"

**Xceed ToolKit**: Is an Extended WPF Toolkit by Xceed is the number one collection of

wpf controls,components and utilities for creating next generation Windows applications.

1. Creating shapes and making them drag gable from the toolbox to the canvas using Content and thumb Control.

**Content Control**: serves as a container for the object that we want to place on the canvas and it is actually this Content Control that we are going to move and resize. And because the content of a Content Control can be of any type, we will be able to move and resize objects of any type on our canvas.

**Thumb** class that is another base class of WPF layout controls and visual controls. It defines methods, properties and events for controls that can be dragged by the user, using a mouse or similar pointing device. The dragging is usually done to resize or move controls.

1. **Toolbox**: is an Items Control that uses the Toolbox Item class as default container to display its items.

**Toolbox Item**: is the place where drag operations are actually started if you want to drag an item from the toolbox and drop it on the canvas.

1. Adding drag and drop method to the canvas using RubberBand Adorner.

**Rubberband** Adorner. Create a Rubberband adorner by inheriting from Adorner class. **Adorners**: are a special type of FrameworkElement, used to provide visual cues to a user. Among other uses, Adorners can be used to add functional handles to elements or provide state information about a control.

1. Coloring the shape strokes and fill using color picker by passing in the data object of the shape that was drag and drop on the canvas. In to the color picker method
2. **Populating Toolbox** with shape by binding the expander to the FlowChartSymbols in designerItem file where the shapes are created and store.
3. Adding shape to the Canvas Item using the MyDesignerCanvas\_previewDrop to capturing the items dropped on the canvas and ensue that each shape is not added more than once to the Canvas Item not minding the number of time a shape is been drag and dropped on the canvas.
4. Adding Selection Criteria using algorithm in the SelectionCriteriaComboBox\_SelectionChanged method to check the shape details, it position in the array and add the shape to the criteria list where the whole item on MydesignerCanvas are been stored.
5. Saving and Loading of Diagram from local Database using serialization method to convert the shape in to JSON object that can be saved in to a local file in .txt format and can be loaded from there
6. I would have saved the serialized data to the specified ms access database file but I don’t have ms access installed on my computer at the time of developing this solution and would be glad to do so but for the constraint of time.