SUMMARY REPORT

Over the course of the project, I learned a great deal about C# Windows Forms library-based programming and the creation of a program that generates SVG files. My original goal was to build a program that could create and modify images using a range of tools, including shapes, colors, and lines, and to incorporate features like shape correction, normalization, auto-completion, and the ability to edit existing SVG files.

As I worked on the project, I successfully implemented various features like the ability to draw shapes like ellipse, rectangle, square, circle, line, and freeform with different stroke width, stroke color, and fill color. The program also includes a ruler and guideline feature, a fun button to generate random shapes. Users can select different stroke and fill colors and thicknesses, and there is an option to choose a background color. The Menu bar in my project was an essential component of the user interface. It allows users to access various features of the program, such as file management, editing options, and view settings. One of the main advantages of the Menu bar is that it makes it easier for users to find and use the program's features without having to navigate through complex menus or dialog boxes. With my Menu bar, users can quickly access features such as creating a new file, opening an existing file, saving their work, undoing, or redoing changes, selecting, and deleting elements, and more.

In terms of features that had to be dropped due to time constraints, the scroll bar was one such feature. Originally, I had planned to implement a scroll bar in the picture box, which would have allowed users to zoom in and out of the image, as well as pan around the canvas. Unfortunately, due to time constraints, I had to drop this feature from the project. I prioritized other features, such as the ability to draw different shapes and use various colors, which I believed were more essential for users. Another feature that had to be dropped was the print dialog. Although it would have been useful for users to be able to print their creations directly from the program, I encountered difficulties implementing this feature. The NuGet packages I used for printing did not work as expected, which consumed a lot of time trying to resolve. As a result, I decided to drop the print dialog from the project.

Lastly, the implementation of the Open file feature in different formats had to be dropped as well, as it proved to be too complex to implement within the timeframe I had. This feature would have allowed users to open and edit different file formats such as PNG or BMP files, but the complexity of the code required for this feature was beyond my initial estimates. I had to prioritize other features to ensure that the program was completed within the given timeframe. However, I believe that the current functionality of the program provides users with enough tools and options to create and edit images effectively.

I learned a lot about the importance of organization and patience in programming. Separating code into different files helped me call methods, properties, and classes in different files, making the project more organized and manageable. Although, I was not able to create any NUnit Tests, as there was a compatibility issue with Windows Form and my .NET Core or .NET 5+ application. Despite this limitation, I still made sure to perform tests myself visually to ensure that the program was performing as expected. I also provided screenshots in the Tests folder to document the testing process. I learned a great deal from this project, including the importance of patience, perseverance, and breaking complex tasks down into smaller parts. The project was a valuable learning experience that helped me to improve my skills in C# programming, and I am proud of the final product that I was able to deliver. While the program did not include all of the features outlined in the initial proposal, I believe that the final product is still an excellent resource for users who wish to create vector graphics for web design, logos, and other applications.

In conclusion, the project was a success, despite some features being dropped due to time constraints. The program provides various tools and capabilities that can be used to create vector drawings and is user-friendly. As I continue to work on the project, I plan to incorporate third-party libraries and develop a strong data structure to manage the SVG files, allowing for the addition of more complex features in the future.