Design Summary for Disk File Line Count:

Overview

This project counts the number of lines in large text files. The application features a graphical user interface (GUI) that allows users to specify the file path, start the counting process, and view the running and final line count. The application is built using Avalonia UI and C#.

Project Structure

1. **MainWindow (GUI Window)**
   - FilePathTextBox: TextBox for entering the file path.
   - CountLinesButton: Button to start the counting process.
   - RunningLineCountTextBlock: TextBlock to display the running line count.
   - FinalLineCountTextBlock: TextBlock to display the final line count.

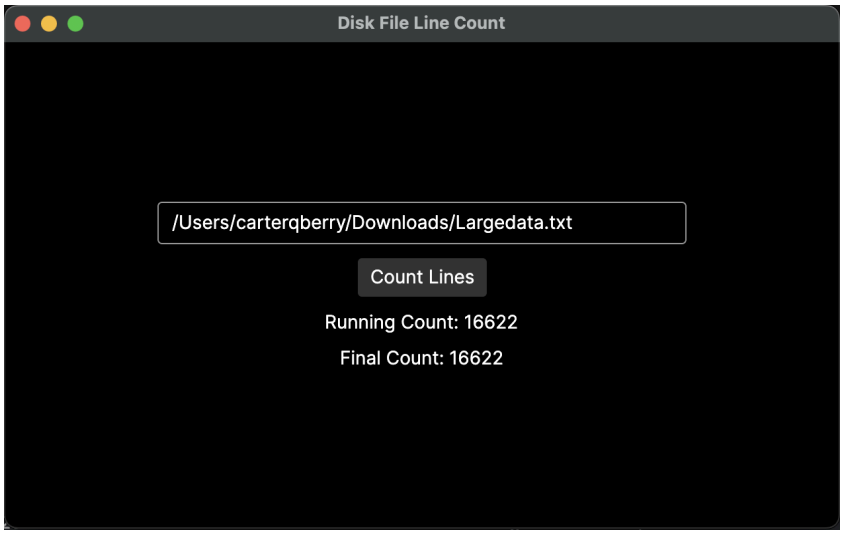
2. **MainWindow.axaml**
   - Layout: Defines the layout and styling of the main window using Avalonia XAML.
   - Styling: Sets background colors and text colors

3. **MainWindow.axaml.cs**
   - Event Handlers: Handles user interactions and file processing.
   - Asynchronous Line Counting: Performs line counting in a background task to avoid blocking the UI thread.
   - Error Handling: Shows an error dialog if the file path is invalid or other issues occur.

Features

- File Path Validation: Checks if the provided file path is valid and if the file exists.
- Asynchronous Counting: Uses a Task to perform line counting asynchronously to prevent blocking the UI thread. Updates the running line count in real-time with a slight delay to ensure visibility.
- Error Handling: Provides feedback through an error dialog if the file path is invalid.

Screenshot of App



Conclusion

The Disk File Line Count Application provides an efficient and user-friendly solution for counting lines in large text files. It has a graphical user interface built using Avalonia, where users can easily input file paths, start the counting process, and view both real-time and final line counts. The application ensures responsiveness by performing line counting asynchronously, preventing any disruptions to the user experience.