Qt lab #2

Jinghan Liu 6/10/24

Note: This project uses CMake for configuration, hence no .pro file is present.

Overview

This application is a basic drawing tool developed using the Qt framework. It allows users to draw different shapes (lines, rectangles, and ellipses) interactively on a canvas, change their attributes (color, thickness, style), and save/load shapes to/from a file.

How to Use

1. Drawing Mode

- Use the toolbar to select the color, thickness, style and type of shapes before drawing.
- Hold the left mouse button to start drawing the shape.
- Release the left mouse button to complete the shape.

2. Edit Mode

- Right-click on any existing shape to select it.
- Hold the right mouse button and drag to move the shape.
- Use the mouse wheel to resize the selected shape.
- Use the toolbar to change the color, thickness and style of the selected shape.
- Right-click on empty space to exit edit mode and enter drawing mode.

3. Saving and Loading

- Save: Click on "File" -> "Save" to save the current shapes to a `.path` file.
- Load: Click on "File" -> "Open" to load shapes from a `.path` file. The canvas will be reset with the loaded shapes.

Code Structure

- MainWindow: Handles the main application window and its UI components, including the toolbar and menu.
- Canvas: A `QWidget` subclass where all drawing occurs. Handles mouse events for drawing, selecting, and manipulating shapes.
- Shape Classes:
    - Shape: The base class for all shapes, providing common properties and methods.
    - Line, Rectangle, Ellipse: Derived classes from `Shape` representing specific shape types.

Questions Answered from Lab

- Step 1: Created a `QWidget` subclass (`Canvas`) and set up its minimum size.
- Step 2: Enabled drawing a line interactively using mouse events.
- Step 3: Added controls for setting color, thickness, and style before drawing.
- Step 4: Allowed drawing different shapes: lines, rectangles, and ellipses.
- Step 5: Implemented a display list to manage multiple shapes with varying attributes.
- Step 6: Added edit mode to select, move, resize, and modify shapes.
- Step 7: Confirmed saving changes before exiting the program.
- Step 8 (Optional): Used Qt Designer to add a control panel for graphical attributes.
- Step 9 (Optional): Implemented saving/loading shapes to/from `.txt` files.

Future Improvements

- Add support for additional shape types (e.g., polygons, curves).
- Implement advanced editing features like undo/redo.
- Provide an option for grouping shapes.

Testing and Compatibility

- The project has been tested on Linux (at Télécom Paris) to ensure compatibility and correctness.
- Please ensure you have the necessary permissions to read/write files when saving or loading shapes.