

2D Shape Editor – Final Project Report

1. Overview

This project is a web-based 2D shape editor built using HTML5, CSS3, and JavaScript. It allows users to draw, transform, group, and manipulate various geometric shapes on a canvas with interactive controls. Advanced features like zooming, exporting to PDF, clipboard support, and grid toggling were implemented as extra credit functionalities. The demo link is mentioned below <https://www.youtube.com/watch?v=XeXEvPBfXjk>

2. Issues Faced

- Maintaining shape selection and interaction logic during transformations (rotate, scale, move) was challenging.
- Toggle functionality for grid visibility initially conflicted with canvas rendering and visibility of shapes.
- Managing consistent rendering with zooming and panning required careful redrawing using scaling.
- Clipboard operations (copy/paste) required handling deep shape object cloning and redraw logic.
- Canvas export (PDF/image) needed dimension matching to avoid cut-off or stretched outputs.

3. Lessons Learned

- Gained practical understanding of canvas API for 2D graphics rendering.
- Learned how to implement zooming using transformations (scale) and how to redraw objects accordingly.
- Developed JavaScript event handling skills for real-time interactions like mouse events and keyboard shortcuts.
- Learned how to implement undo-redo stacks using deep copies of shape arrays.
- Understood how object-oriented programming can structure complex shape behavior using inheritance.

4. Remaining Bugs or Limitations

- Group operations are limited to transformation only and not yet extended to attribute editing of grouped objects.

5. Extra Credit Implemented

- Multi-select: Ability to select shapes one by one (via mouse) and apply operations.
- Grouping of objects: Shapes can be grouped for collective transformation (move, rotate, scale).
- Operations on group: Move, scale, and rotate operations apply to grouped shapes.
- Support for text: Labels or text are not currently integrated into shapes (could be added).
- Zoom in/Zoom out: Implemented with canvas scaling and redrawing logic.
- Pan/Scrollbar: Panning handled via shape movement, scrollbars not added.
- Additional shape families: Includes Polyline, Polygon, Curve, and Ellipse in addition to basic shapes.
- Clipboard support: CTRL+C and CTRL+V support for copy-paste with shape translation.
- Export as PDF: Canvas can be exported using jsPDF with appropriate scaling.
- Export to LaTeX is not implemented.

6. Conclusion

This project provided a strong foundation in interactive graphics programming, UI control logic, and canvas manipulation. The implementation covers both required and extra credit features and demonstrates good modular design. There is still room for refinement in UI polish and LaTeX export capability.