



Code Architecture: Key Implementations

- Most of the interaction between the tools and GUI Components happen using the Observer Pattern
 - example: All the shapes, Pen, Airbrush, and the fill tools **observe** changes that happen from the colour swatches or the colour chooser
- The ToolsManager sort of acts like a **Façade** between the GUI Components and the Tools
 - some of the roles of the ToolsManager include:
 - creating instances of some of the GUI Components
 - linking the observers to the observables
- Each Tool contains a Clickable Instance.
- Changing the behaviour or appearance of the tools can be done through clickable.

Code Architecture: Key Implementations contd.

- The ClickableTool Class is used to represent common properties and behaviour of all the tools
 - Such as:
 - Specifying if the tool needs to update the Undo History
 - Specifying when a shape layer should be rasterised (for example drawing using a pen on a rectangle layer)
 - Specifying if the Tool needs to update the Layers Preview Panel when used
- Every entity in the application that is rendered into the canvas through some sort of drawing defines its drawing behaviour using a **Graphic Class**
 - example of such entities are lines, circles, text, etc..
 - an example of such class is the **CircleGraphics** Class that has a method that defines the drawing behaviour of a circle in a Graphics Context
 - **Pros of this Design:**
 - can have multiple types of entities that can be drawn easily into a single type of layer (for example can draw a rectangle entity and a circle entity into a single layer)
 - **Cons of this Design:**
 - this design over complicates the code and makes the code not DRY
 - could have specified the drawing behaviour of each layer inside the layer itself
 - When a layer requires a composition of some of the drawing entities could simply created a new Layer Type

Code Architecture: Key Implementation Mistakes

- Every Tool that needs to update the Undo History when used has a reference of the undo tool itself
 - creates unnecessary code duplication
 - needing to pass the undo tool to other tools through the constructor
 - **Solution:** the undo history mainly stores the current state of the layers from the Layers Handler, therefore the Layers Handler and the undo could have simply communicated together instead
- Every Tool that needs to update the Layers Preview Panel when used has a reference of the LayersOptions Panel
 - same problems as the first bullet point
 - **Solution:** the Layers Preview Panel needs to be up-to date with the layers state from the Layers Handler, so the Layers Handler and the Layers Preview Panel could have simply communicated together instead
- Every Tool that needs to refresh the Canvas when used, needs to explicitly inform the LayersHandler through a function call
 - creates unnecessary code duplication
 - source of bugs if a tool forgets to refresh the canvas when finishing its operation
 - **Solution:** LayersHandler should refresh the Canvas by itself internally whenever a change in the layers happen and if a tool affects a layer directly then the layer and the Layers Handler can observe the layer when its state changes

Classes

Number of classes: ~67

Tools Classes

- Airbrush
- BlurTool
- CircleTool
- CropTool
- Clear
- CutterTool
- Delete
- DrawingTool
- EyeDropperTool
- FillTool
- SelectionTool
- TextTool
- PenTool
- UndoTool
- EraserTool
- TriangleTool
- RectangleTool
- ToolsManager
- ShapeTool
- ClickableTool

Layers & Graphics Classes

- LayersHandler
- LayerData
- ShapeLayer
- TextLayer
- CircleLayer
- ImageLayer
- DrawingLayer
- RectangleLayer
- TriangleLayer
- LayerGraphics
- RectangleGraphics
- TriangleGraphics
- LineGraphics
- CircleGraphics
- SpecificGraphic
- TextGraphics
- ShapeLayerGraphics

Gui/Frontend Classes

- Clickable
- Display
- LayerOption
- LayersOptions
- OurCanvas
- ToolsPanel
- OptionsPanel
- CanvasHandler
- ColorGui
- ColorsPanel
- PenOptionsPanel
- MenuPanel

Classes

Number of classes: ~67

Menu Tools Classes

- ImageLoader
- SaveAs
- Help
- ProjectLoader
- ProjectSaver

Observer Pattern Classes

- Observer
- Observable
- CanvasObserver
- CanvasObservable
- LayerObserver

Miscellaneous

- Pen
- Brush
- Resize
- Constants
- Footer
- ImageFilters