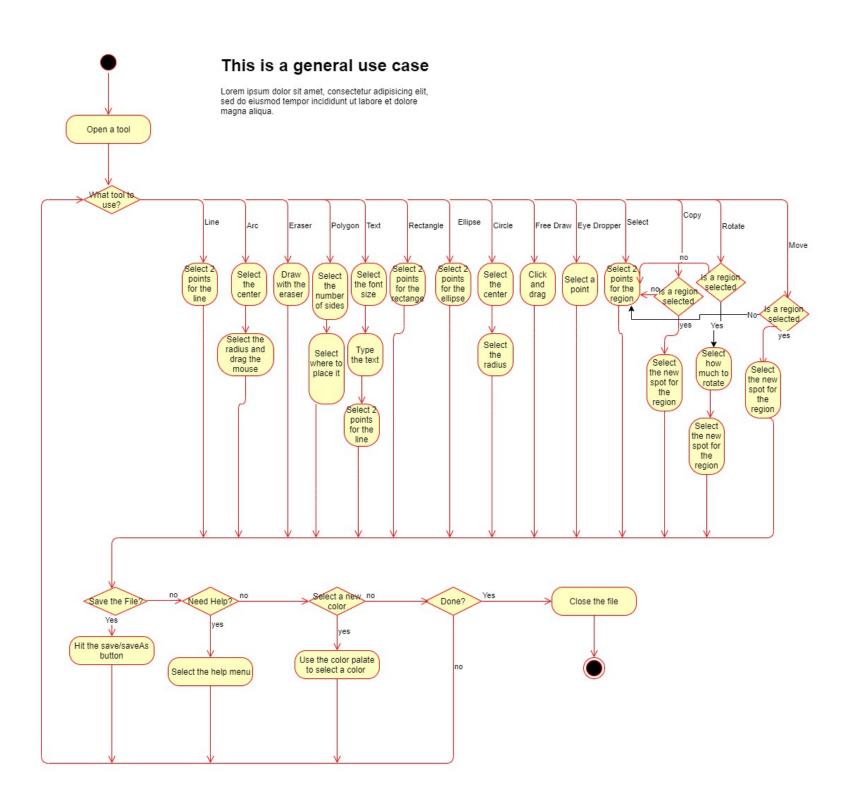
## In-depth class UML

This is a look at two of the most detailed image processors. The Image Editor opens up a tab and Shape Pane opens the image and manages how to draw shapes.

☐ ImageEditor	☐ Shape Pane
+ field: File currentFile	+ field: type
Boolean isSaved	private AnchorPane anchorPaneObject
PrimaryController mainFrame	private int arcClick
·	private javafx.scene.Node currentShape
Double MAX_HEIGHT  Event onCloseEvent	private java.lang.String currentTool
ZoomableScrollPane scrollPaneObject	private ImageEditor ImageEditorObject
,	private javafx.scene.image.lmage imageShown
ShapePane ShapePaneObject	private PrimaryController mainFrame
File tempFile	private javafx.scene.image.lmageView mainImage
+ method(type): Void autoSave()	private javafx.scene.shape.Rectangle recSelect
Int close(boolean askOnClose, boolean recursiveClose)	private java.util.Stack <javafx.scene.node> RedoStack</javafx.scene.node>
private void closeTab()	private java.util.Stack <javafx.scene.node> UndoStack</javafx.scene.node>
void copyFileUsingChannel(java.io.File source, java.io.File	private double xPressedLocation
dest)	private double yPressedLocation
String getFileExtension()	+ method(type): type
String getFileName()	void clearUndo()
Double getImageHeight()	private void drawdrag(double x2, double y2)
Double getImageWidth()	private void drawFinal()
Int loadImage()	private void drawStart(double x1, double y1, boolean isPrimaryClicked)
void redo()	javafx.scene.image.WritableImage flatten() //Flattens th
void reopenedCall()	current image.
void resize(double new_width, double new_height)	Double getImageHeight()
int save()	double getImageWidth()
int saveAsImage(File file)	javafx.scene.image.WritableImage getWritableImage()
private int saveImage(java.io.File file)	private Boolean isRedoEmpty()
private int saveImage(java.io.File file, boolean autoSave)	private Boolean isUndoEmpty()
private int setFile(java.io.File new_File)	void loadImage(java.io.File file)
void setNotSaved()	void pushUndo(javafx.scene.Node n)
void setPannable(boolean isPannable)	int redo()
private void setSaved()	void resizeImage(double new_width, double new_heigl
private void setup_draw()	private static void setPolygonSides(javafx.scene.shape
void undo()	polygon, double centerX, double centerY, double radius
	int undo()
	void updateUndoBtn()



## Classes that are being used

This shows the overall hierarchy that is used in class structure.

A few classes were left out for simplicity

