Post Project Write Up
Ian Trapp
CSCI 1302 – Plaue

Functionality:

This program has a top bar that contains the 6 buttons: a button to open the file chooser, 3 radio buttons to choose the desired pixel size, and a button to pixelate. The reset button clears the window of all photos. The 3 radio buttons are in a button group, so only one may be selected at any given time. Upon selecting a file, that image will appear below the top bar. Selecting a size and pressing the "Pixelate!" button will display your pixelated image below the original. The top bar is its own JPanal, the two images are ImageIcons placed on JLabels. To pixelize each image, I set a final int "size", and loop through the photo sectioning out areas of the photo in boxes with the dimensions of size using a writeable raster to scan a pixel and apply its color to its neighboring pixels. This method then saves the image file and returns the string representing the save file and displays it in a JLabel. Before the image is selected and manipulated and after the reset button is pressed, the label's image icons are set to null. They get reassigned to the respective image when an image is selected or manipulated. The program will allow you to pixelize the same photo and keep it in the window, as well as select new photos and keep them in the window until you press the reset button. I have a toolbar with 2 menus: file and help. File has an option to quit the program, while help has information about the program as well as information on how to get started. The program will not allow you to pixelate a non-existent photo thankfully, so upon attempting to do so the user will see a JOptionPane popup asking them to select a photo. Since the program cannot pixelate a photo without a specified pixel size, the program will also ask the user to select a size before pixelization. The program will not allow it to be ran with any command line arguments except flags, and will print an error to the command line upon attempting to run the program with arguments that aren't flags.

GUI Design:

As an intro project, I decided to make the UI as simple as I could while still retaining all the functionality I needed. Instead of getting lost in the world of swing, I decided simplicity was key to improve my understanding of swing. I started off extremely confused, but by hammering the basic concepts I think I gained a solid understanding of the general "UI" idea, and of how things are intended to work together. I elected to stick with the original layout manager. This leads to some issues when resizing the window, but ultimately it allows everything to fit nicely when kept at the stock size, 800x800. Swing was very confusing at first, but I ended up learning quite a lot as the project progressed. I decided to make the JFileChooser a popup as opposed to being imbedded in the window, as it made no sense to have it take up space when it wasn't being used the majority of the time. The buttons are laid out in logical order of selection: you choose your file, then your size, then pixelate it. Following this, you reset and start over again. I elected to use JOptionPane for the help menu because it allows me to create efficient popup dialogs with two or less lines of code to display getting started, author, and program information. I decided to keep both photos on there as opposed to replacing the non-pixelated photo with the pixelated photo. In doing so, it allows the user to compare the two or more photos. I also believe this would have created too much awkward white space in the program.