Final Project Documentation

Members: Kenji Uchida, Quang Nguyen

Section 1: Functionalities

- Changing the size of the source image
    - Scaling up
    - Scaling down
- Rotating the source image 90 degrees
- Flipping the source image horizontally
- Edit source image filter to:
    - Grayscale
    - Blue
    - Darker
    - Brighter
- Reset the source image back to original image

Section 2: Architecture & Design

- Image Editor App can be used by any Android device
- Software requirements require Android SDK API11 or above
- Hardware requirements include running on Pixel 3A emulator or android device (configurations can be adjusted to different types of devices)
- Button definition and methods wrapped to follow object oriented design
- Future code elaboration will include data transfer from android database to select different images. Option to take new photos will also be included.
- Following import statements used by Image Editor App:
    - import android.graphics.Bitmap
    - import android.graphics.Color
    - import android.graphics.Matrix
    - import android.graphics.drawable.BitmapDrawable
    - import android.os.Bundle
    - import android.view.View
    - import android.widget.ImageView
    - import androidx.appcompat.app.AppCompatActivity

Section 3: GitHub Location of Code

· https://github.com/Kenji-uchida/CPSC411SemesterProject

Section 4: Test Results

- **Scenario 1:** Testing different image types
    - Trying if the application can work with PNG and JPG

  Result: PNG and JPG image appear in the image view of the application

- **Scenario 2**: Changing image color
    - Testing functionality for buttons to change image colors

  Result: image changed colors accordingly to which button pressed

- **Scenario 3:** Changing the image's size
    - Testing functionality of decreasing size and increasing size buttons

  Result: image changed it size accordingly to which button pressed

- **Scenario 4:** Rotating the image
    - Testing functionality for button to rotate image

  Result: image can be continuously rotated at 90degree increments

- **Scenario 5:** Setting the image back to original image
    - Testing the functionality of the button to return the image back to original after edits

  Result: image returns to its default size, color and rotation after the button is pressed

- **Scenario 6:** Testing overlap buttons pressed
    - Testing functionality of pressing buttons multiple times before resetting image

  Result: image editor app successfully handles overlap button functionality

Section 5: *Future Work*

- Creating more color schemes options
- Making a slide scroll/drop down menu for color schemes instead of buttons
- Making a scroll for changing the image's size more freely
- Add more functionality:
    - Filters
    - Markup capabilities including pens/highlighters/pencils
    - Image dimensions and details
    - Dynamic zoom in/out capabilities
- Being able to crop the image

- Develop more user-friendly UI
- Functionality to select images or take new images to edit