Usage of the Kevin Drawing app:

Click on the top left hamburger menu button to access the side menu. The side menu allows you to use different drawing mode and control the size and color of different objects. Tap the area outside the side menu to exit the side menu. Top right corner back button allows you to access the previous step of your painting.

Caution!!!!!!!:

This app work with SDK version 26-28!!!!!

When you select any of the mode in the side menu, you are at the mode and you can only make changes to the item from that mode only. For example, when you are in Rectangle mode, you can only move and create rectangles on the screen, you cannot move the circle in the screen. And the back button only works on the object from the mode you selected.

File will be saved in the internalStorage/Pictures/Kevin_drawing_app, please use file explorer in the phone to find the images. In the android emulator, go to the Files app, click on top right three dot logo and click Show internal storage, you can get access to the Pictures file in the internal storage

!!!!!!!!!!!!!! If Android studio give error like Application Installation Failed, please go to File > Settings > Build,Execution,Deployment > Instant Run > Un-check (Enable Instant Run to hot swap code) !!!!!!!!!!!!!!!!!!!!!!!!!

This android app is a single activity application and consisted of 6 java classes. The MainActivity.java control the UI and front page of the app. The activity_main.xml is split into two part, the NavigationView which is the side menu UI which is made of a header with spinners and a list menu, and the PaintView which control all the drawing and printing onto the screen. The Circle, Rectangle_shape and the FingerPath class store the necessary information of the circles, rectangles and lines that are drawn by the users as multiple objects in three different array-lists. This helps to redraw what has drawn in the screen by the onDraw() function, and determent whether user is pointing inside a shape or not. Most of the stuff that I codded are in: com.example.android_drawing_app and the res folder.

Previous Versions:

version 1:

- exploring the android Canvas class
- create a prototype that can draw three type of paths, but the emboss path is very lag
- use linked list to store all the paths
- use android toolbar built in menu class to control the selection of line type

version 2:

- join all the line selection into a sub menu of Line Art
- First attempt to implement movable circle, encounter problem because this method can only detect one circle only, it doesn't use the array list properly, didn't store center point of the circles, the touch point in circle judgement doesn't support multiple circles, so cannot implement multiple draggable circle in the canvas, need to seek for new approach

version 3:

- Upgraded the circle implementation, use the linked list more efficiently, now it stores each circle radius and centerX and Y coordinate. So when user touch down, get the fingertip x and y coordinate and loop along all the circle data in the array list and use (centerX - touchX)^2 + (centerY - touchY)^2 <= clc.radius^2 to determine whether the user is touching a circle, if yes obtain the data of that circle and do the movement and store the final destination coordination. if user touch a place outside any circle, create a new circle at that place

version 4:

- Upgraded the menu, get rid of the default toolbar dropdown menu, create a custom side menu, with a custom menu header and 4 android spinner to control the Line size , Line color, shape size and Shape color, user spinner because dont want to let user to type in wrong input and crash the program. the spinner is set up but the onclick part of the spinner is not set but the menu has a left swipe gesture which bring inconvenient in drawing, need to disable it

version 5:

- Improve the art work, get rid of the default green bar design, unify the design with multiple grey color to make the program look nice and clean

- Set up the spinner onclick part so the color can be change, unify the shape color and line color, they share the same spinner now, implement a new background color spinner to let user change background color
- Disabled the swipe gesture

version 6:

- Implement the rectangle shape function, similar idea as the circle function but slightly different, to draw rectangle more points need to be store, like the circle only need to store the center but rectangle needs to store both top left X Y and bottom right X Y, use an equation to check user touch rectangle or not , touchX > rec.topLeftX && touchX < rec.bottomRightX && touchY > rec.bottomRightY && touchY < rec.topLeftY
- Implement a new last step button on the right of the tool bar, it depends on the mode you select, like select circle mode will remove the last circle implemented, line will delete the last line drawn etc.

version 7:

- Improve the touch_mode control using enum instead of integers
- Implement the save function, which can save the canvas into phone storage, Storge/Pictures.

Version 8(final):

- Implement functions to ask user for permission on writing files into their phone internal storage