Faculty of Engineering - Ain shams University

Mechatronics Department

Machine Vision CSE480

Milestone-1

Name	ID
Mohand Aymn Abd El-kader	18P6298

Submitted To:

Dr. Hossam Hassan

Eng. Yehia

Contents

Abstract

Development of a mobile app for Cam scanning documents using machine vision techniques over the years was interesting, to make an app that is designed to provide users with a fast and easy way to digitize physical documents into digital formats. Machine vision algorithms were implemented to detect and extract text from images captured by the Mobile device's camera. the app's user interface was designed to be intuitive and user-friendly. allowing user to easily scan documents and save them in various formats. the performance of the app was evaluated through Testing and Studying Several cam scanning apps. results demonstrate that the app provide a reliable and efficient solution for Cam scanning documents using machine vision.

At the end, Machine Vision exists in our Everyday Tasks in Life, so we need to implement it in an easy and useful way for anyone to Use it.

Problem Definition

Problem here is to convert a physical (hard-copy) document into a digitized form (soft-copy) to use it in several different applications after , The challenge was to detect the writing and process it more further to detect the key points of the Documents to find out the Biggest Contour and the edges of the document if were it to be scanned as a whole and implement another method simultaneously letting the user to adjust the points surrounding the targeted text to be cropped and processed and saved as pdf in the end. All that in a user-Friendly app which is easy on the eyes of the user containing Nice GUI (Graphical User Interface)

Importance

Many existing mobile scanner apps rely on simple image processing techniques that can lead to low quality or distort scans especially if the lighting conditions are poor or the document is not placed correctly on the camera. Machine vision techniques can help to overcome these limitations by analyzing the image data captured by the camera and applying sophisticated algorithms to correct for distortion remove noise and enhance the overall image quality. The aim was to develop a mobile scanner app using machine vision techniques to produce high quality scans of documents and other paper-based materials. our app uses advanced image processing algorithms to detect the edges of the document, correct for distortion and perspective and then enhance the overall image quality by using machine vision techniques, our app can produce scans that are more accurate more readable and more consistent even under challenging lighting conditions. The importance of this app lies in the ability to provide a fast, convenient and high quality scanning solutions for users who need to scan documents on the go whether you are a student who needs to scan notes for a Class, A business professional who needs to scan receipts for expense reports or anyone else who need to digitize paper based materials our app provides a simple and effective solution by using machine vision techniques and easy to use as an overall app.

Methods and Algorithm

Processing The Doc-image

The First thing to do is to Capture the Document as an Image to apply Machine Vision Techniques on it By Using **OpenCV-python** module and **Numpy** module and handling that image as data, by Applying Simple steps on that data we managed to get the edges of the Document as A whole and defining the contours then modifying key points then wrapping and cropping the image to save it as pdf

Algorithm steps

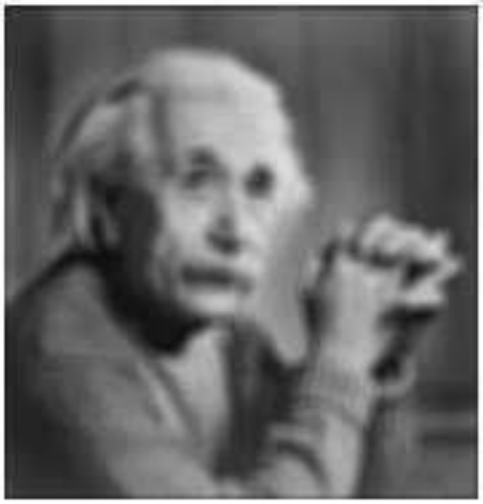
1. Capture the Doc. To be scanned from the mobile Camera
2. Rescaling the image frame (works better this way with opencv)
3. Apply opencv on it to get rid of the colors and turn it to gray scale image for further processing
4. Apply gaussian Blur filter on the gray image to make it blur and remove all the noise as much as possible
5. Apply Canny algorithm for edge detection for detecting all edges, defining the kernel size, Max and min Threshold. Then further processing to get the threshold image
6. Get Contours that exist in the image, loop over all contours to find the biggest contour that consists of 4 corner points (key-points), that's going to be the area to be processed further and then we draw line connecting those points together
7. Waiting The user to re-order those key-points by moving them, then when satisfied with that area, crop away that area
8. Reading The coordinates of the new points and processing the distance between each point and returning the new points position and area bounded by them
9. Get the perspective view of the Doc (bird's eye view), Then crop it
10. Sharpen the image by subtracting the blur from the original gray image to get a high-pass Filter, then add that filter to the original gray image or use Laplacian filter by subtracting it from original (cropped and wrapped image), Then save it as Pdf

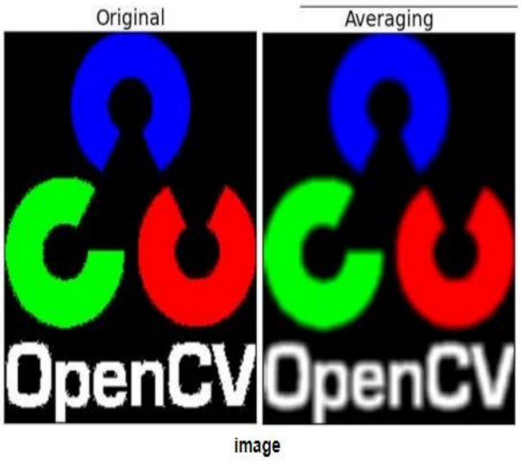
Methods

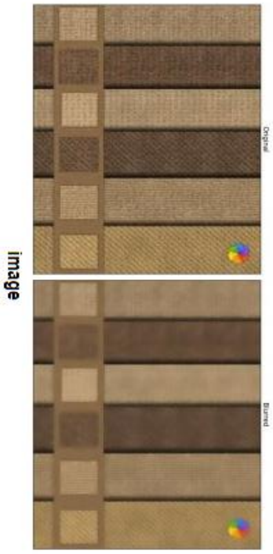
- **Capture the image and convert it to gray scale**

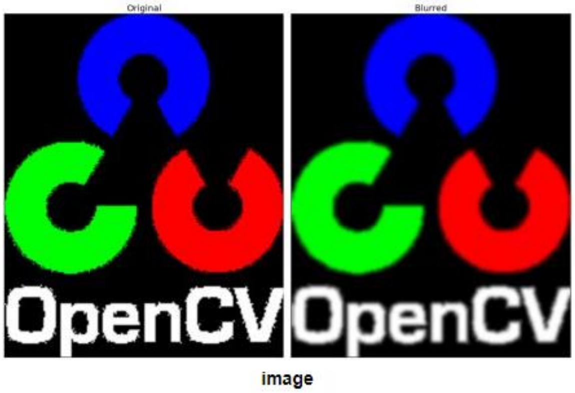
  After Rescaling the image for OpenCV to work on and then Removing the colors from the image to get its raw gray scale data for further processing and that way it’s easier to handle

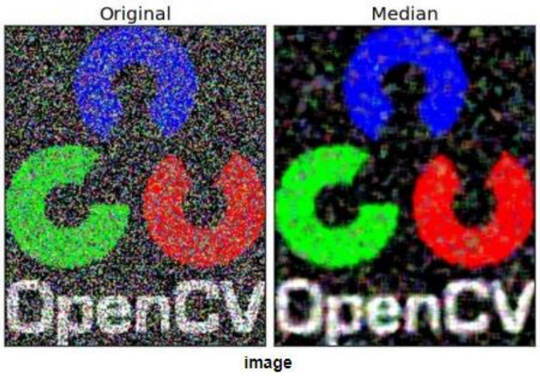
- **Blurring** The Gray Image to Get Rid of the Noise in that image as much as possible there is many ways for blurring such as Averaging, Gaussian Blur, median Blur (salt and pepper noise), bilateral filtering











- **Edge Detection Using Canny Edge Detection Algorithm to detect Edges**

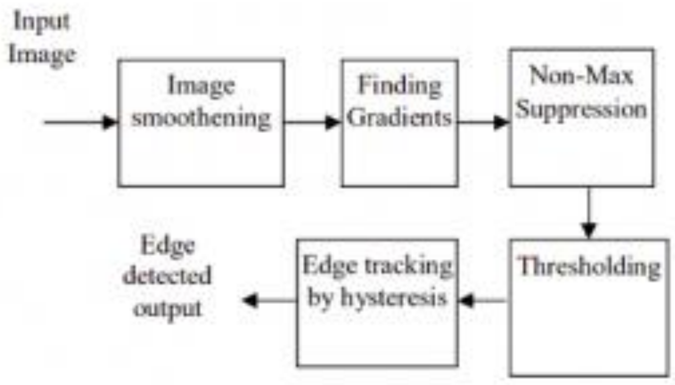
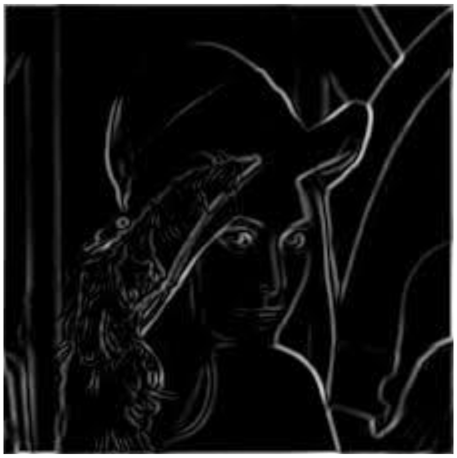
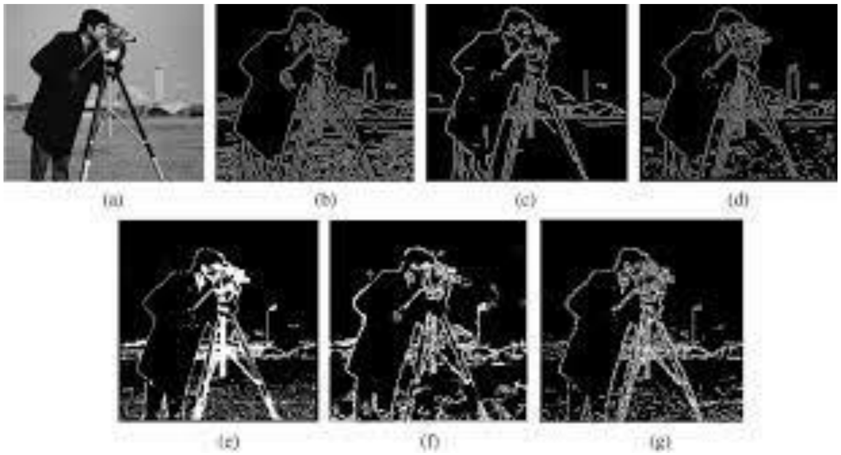
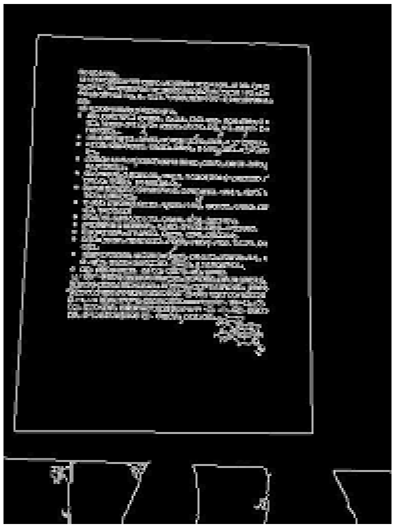


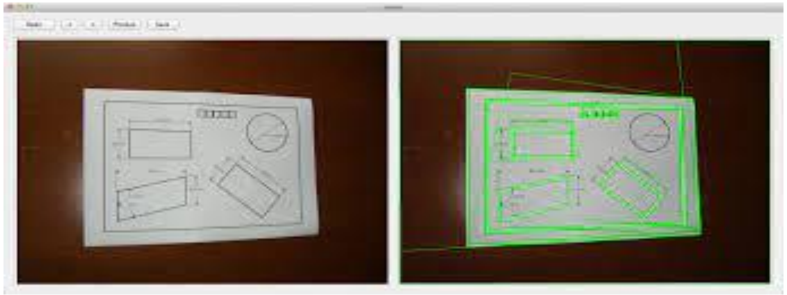
Fig 1.2: Canny Edge Detection

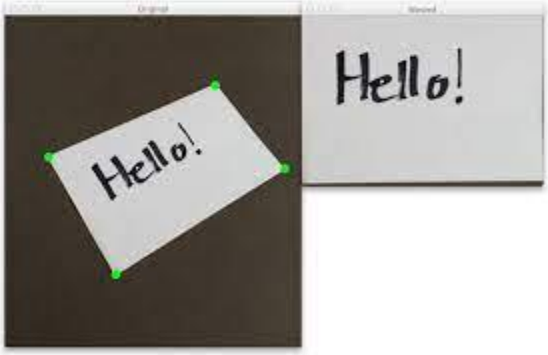


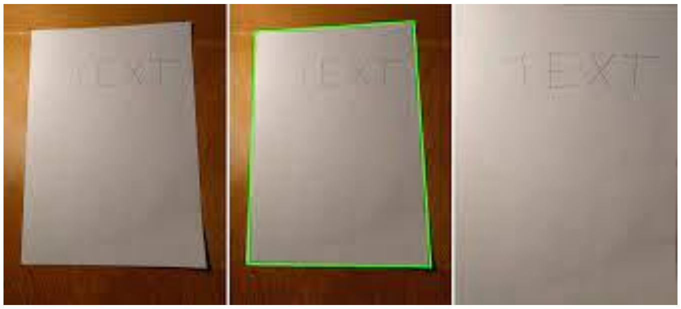




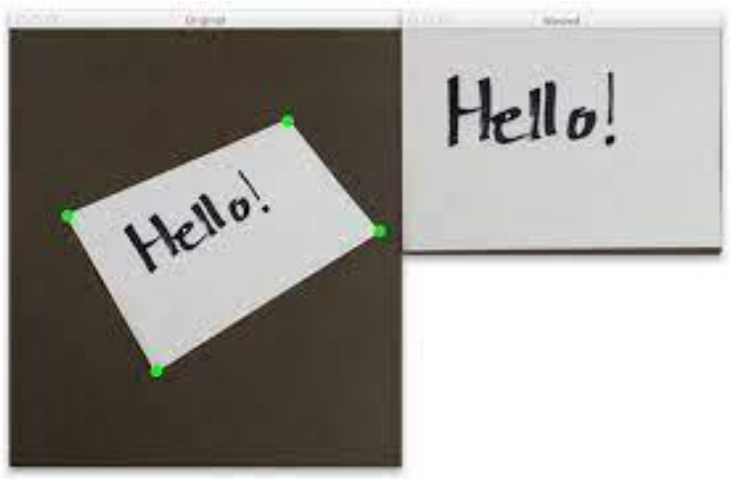
- Next, we define and loop around the contours and find out which of them is the biggest contour from edge detection, this way we found the borders of the document key points of the Document (The four corners = rectangle)



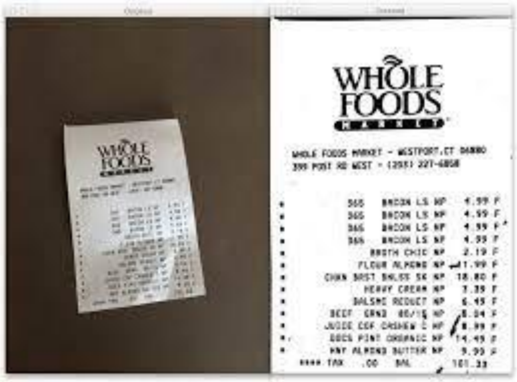




- **Letting The user to re-order the four points (document corners) to adjust the wrapping space to be cropped**

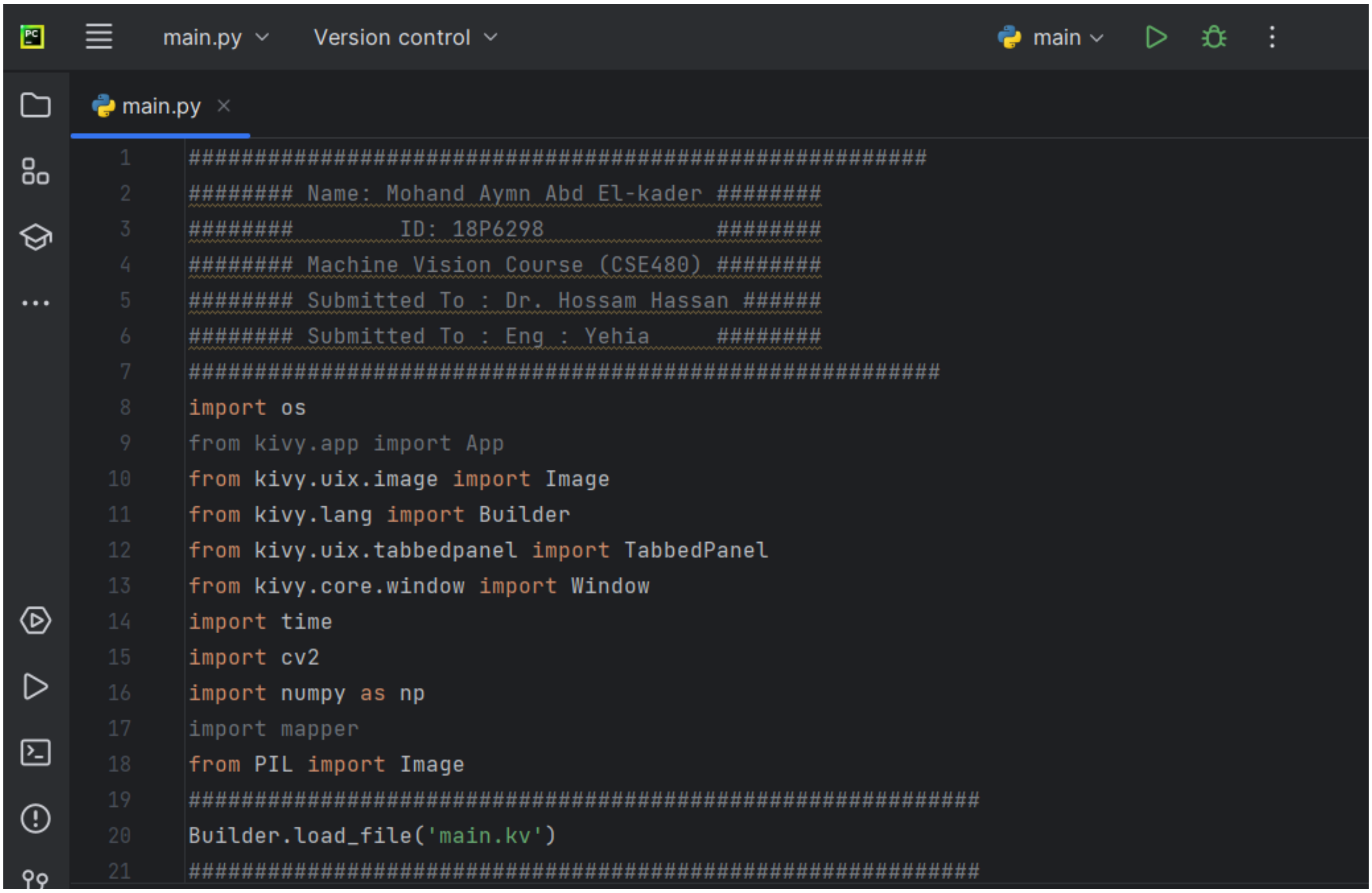


- **Cropping The Bounded Space by the Final Contour and save it as A4 pdf for Document Reading**

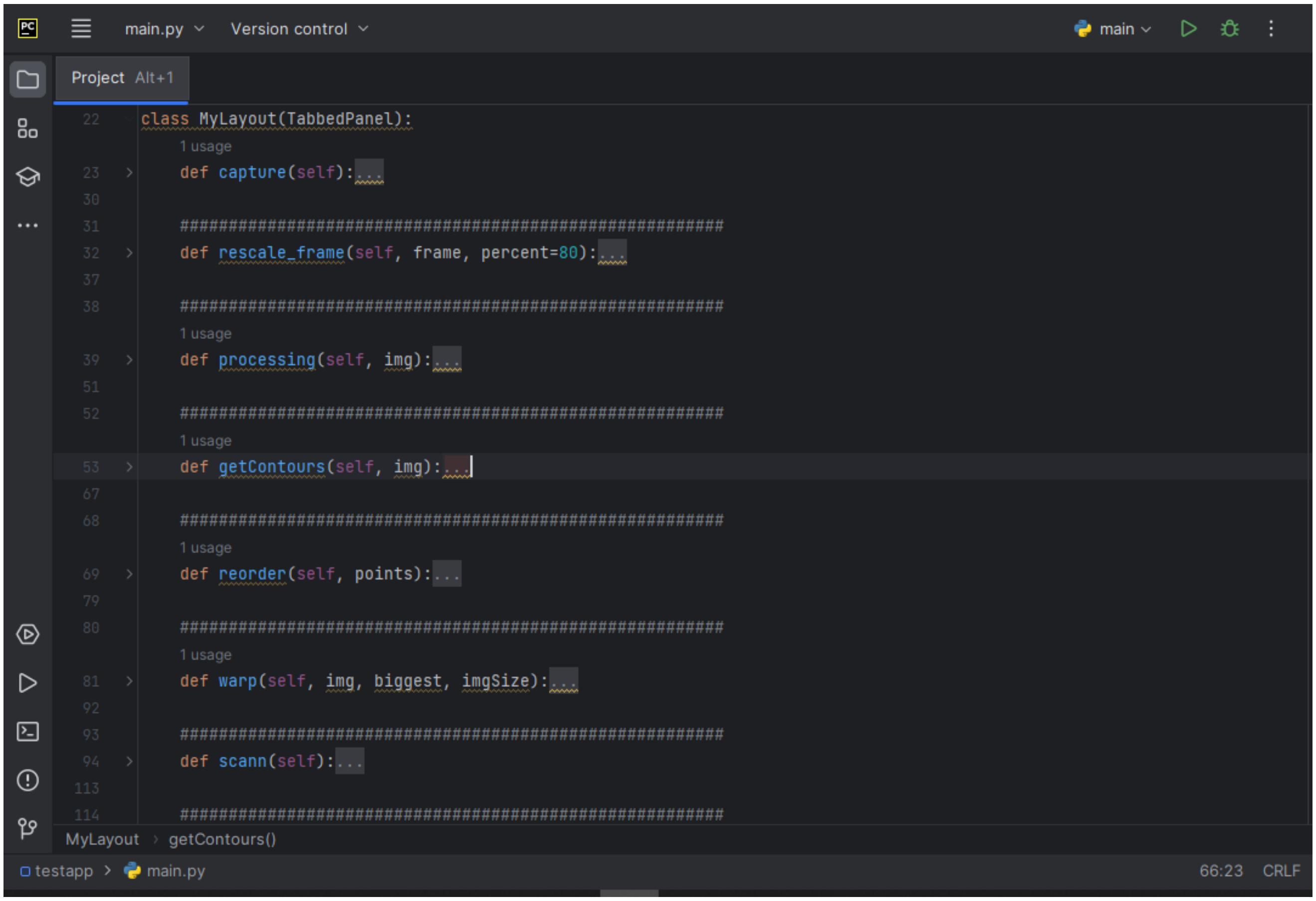


```
WHOLE
FOODS
MARKET

WHOLE FOODS MARKET - WESTPORT,CT 06880
399 POST RD WEST - (203) 227-6858

*        365   BACON LS NP      4.99 F
*        365   BACON LS NP      4.99 F
*        365   BACON LS NP      4.99 F
*        365   BACON LS NP      4.99 F
*           BROTH CHIC NP       2.19 F
*         FLOUR ALMOND NP      11.99 F
*   CHKN BRST BNLSS SK NP      18.80 F
*          HEAVY CREAM NP       3.39 F
*        BALSMC REDUCT NP       6.49 F
*      BEEF  GRND 85/15 NP      5.04 F
*    JUICE COF CASHEW C NP      8.99 F
*    DOCS PINT ORGANIC NP      14.49 F
*    HNY ALMOND BUTTER NP       9.99 F
   **** TAX   .00   BAL       101.33
```

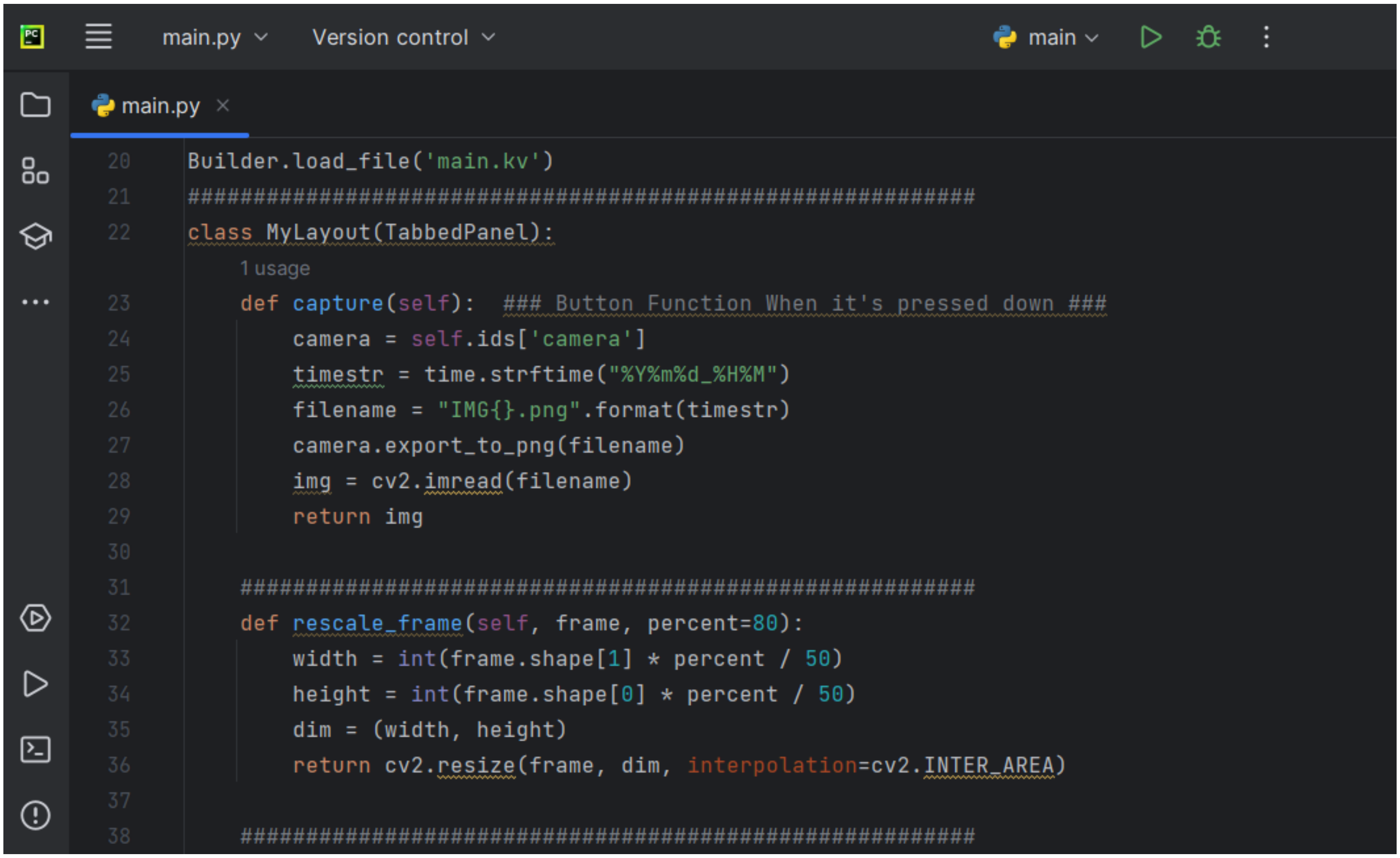
Snippets of The Mobile Scanning App (main.py) Code:



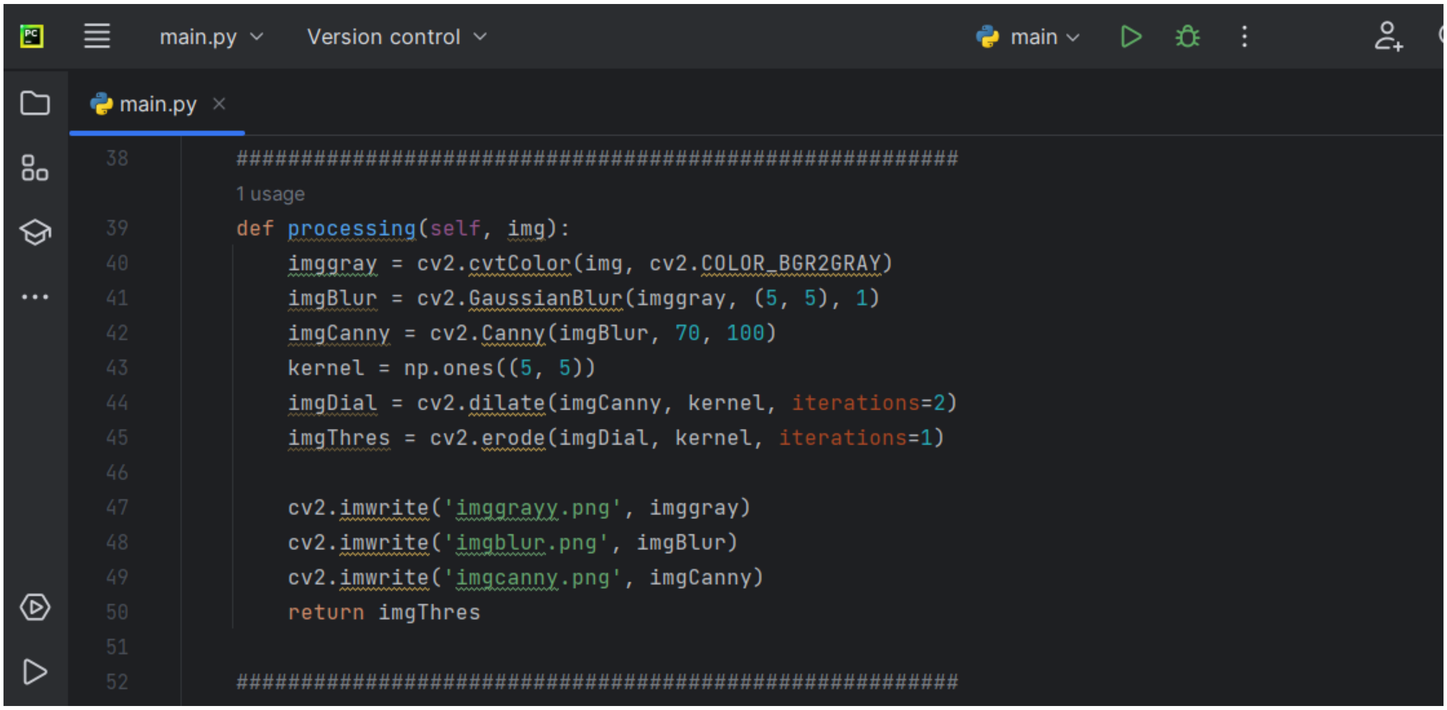
```python
########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
#########################################################
import os
from kivy.app import App
from kivy.uix.image import Image
from kivy.lang import Builder
from kivy.uix.tabbedpanel import TabbedPanel
from kivy.core.window import Window
import time
import cv2
import numpy as np
import mapper
from PIL import Image
############################################################
Builder.load_file('main.kv')
############################################################
```



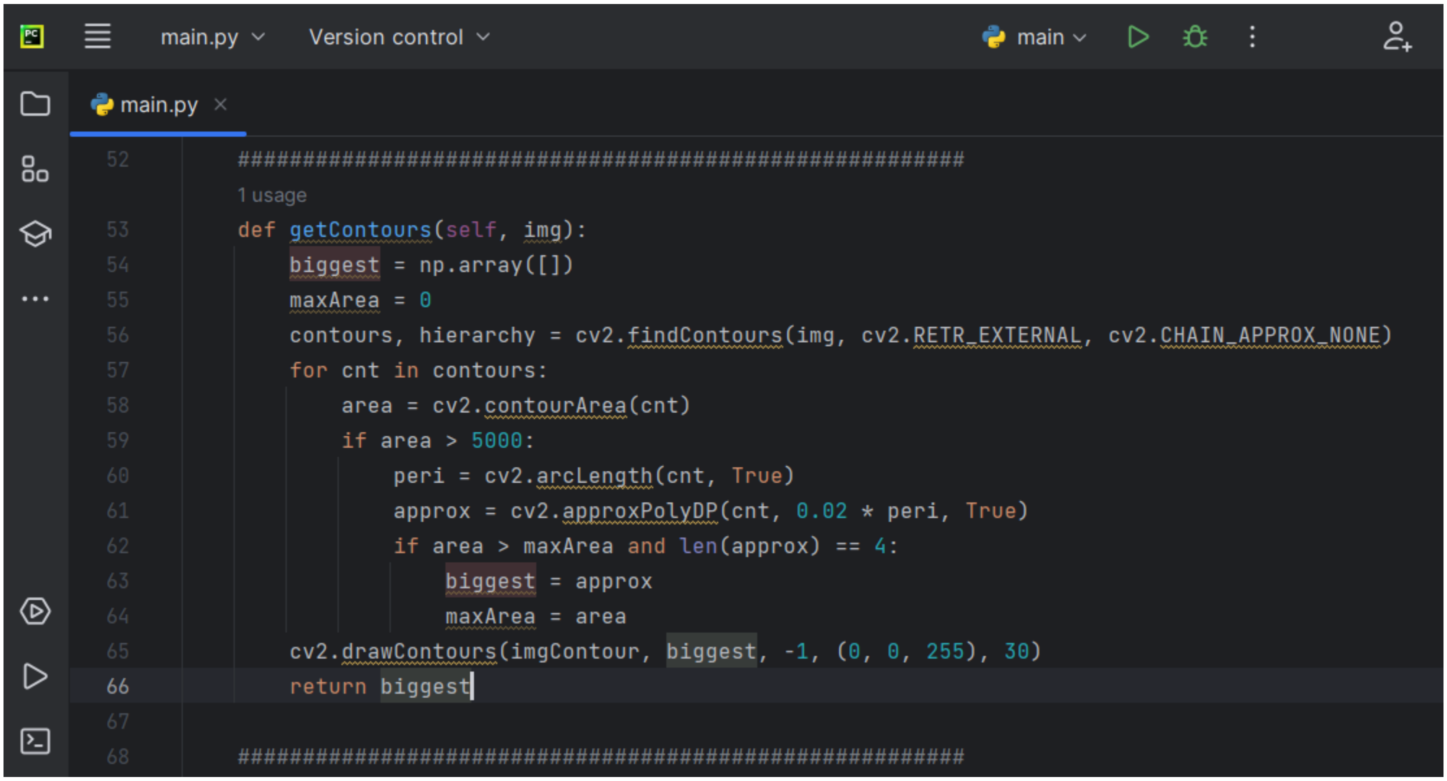
```python
class MyLayout(TabbedPanel):
    1 usage
    def capture(self):...

    ########################################################
    def rescale_frame(self, frame, percent=80):...

    ########################################################
    1 usage
    def processing(self, img):...

    ########################################################
    1 usage
    def getContours(self, img):...

    ########################################################
    1 usage
    def reorder(self, points):...

    ########################################################
    1 usage
    def warp(self, img, biggest, imgSize):...

    ########################################################
    def scann(self):...

    ########################################################
```



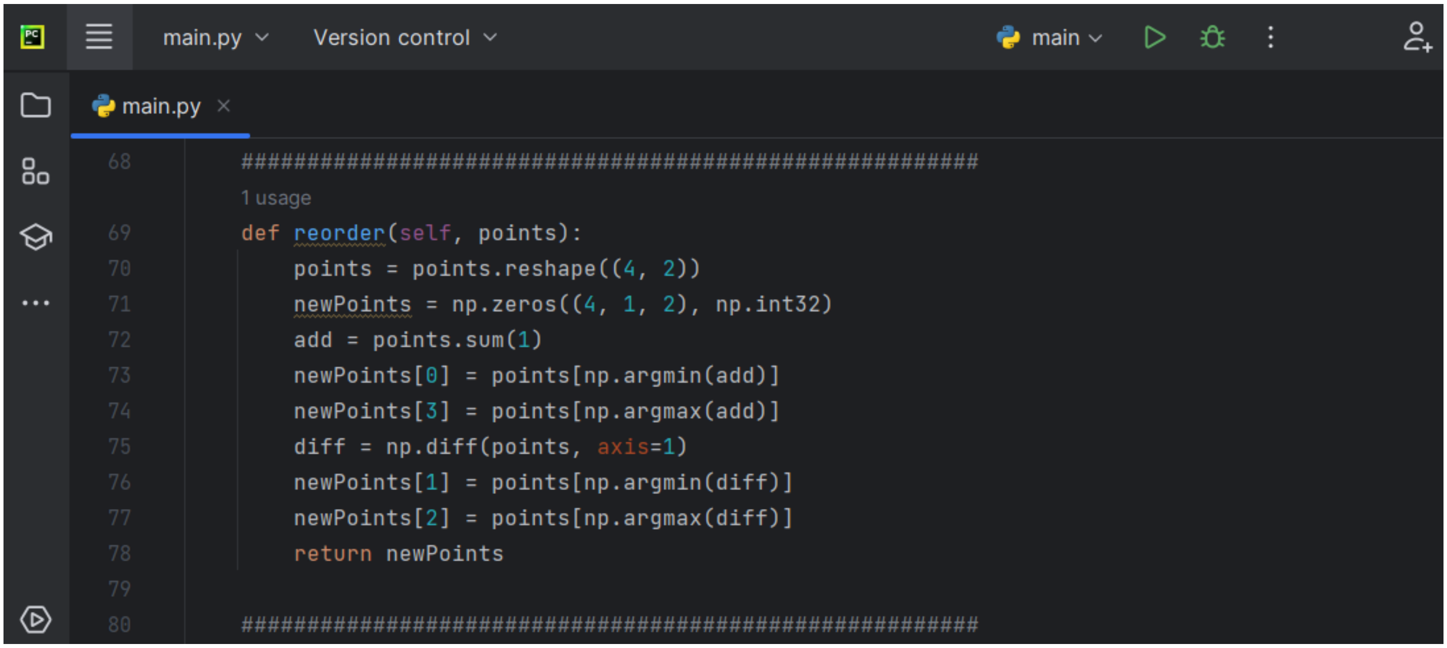
```python
Builder.load_file('main.kv')
############################################################
class MyLayout(TabbedPanel):
    def capture(self):  ### Button Function When it's pressed down ###
        camera = self.ids['camera']
        timestr = time.strftime("%Y%m%d_%H%M")
        filename = "IMG{}.png".format(timestr)
        camera.export_to_png(filename)
        img = cv2.imread(filename)
        return img

    ########################################################
    def rescale_frame(self, frame, percent=80):
        width = int(frame.shape[1] * percent / 50)
        height = int(frame.shape[0] * percent / 50)
        dim = (width, height)
        return cv2.resize(frame, dim, interpolation=cv2.INTER_AREA)

    ########################################################
```



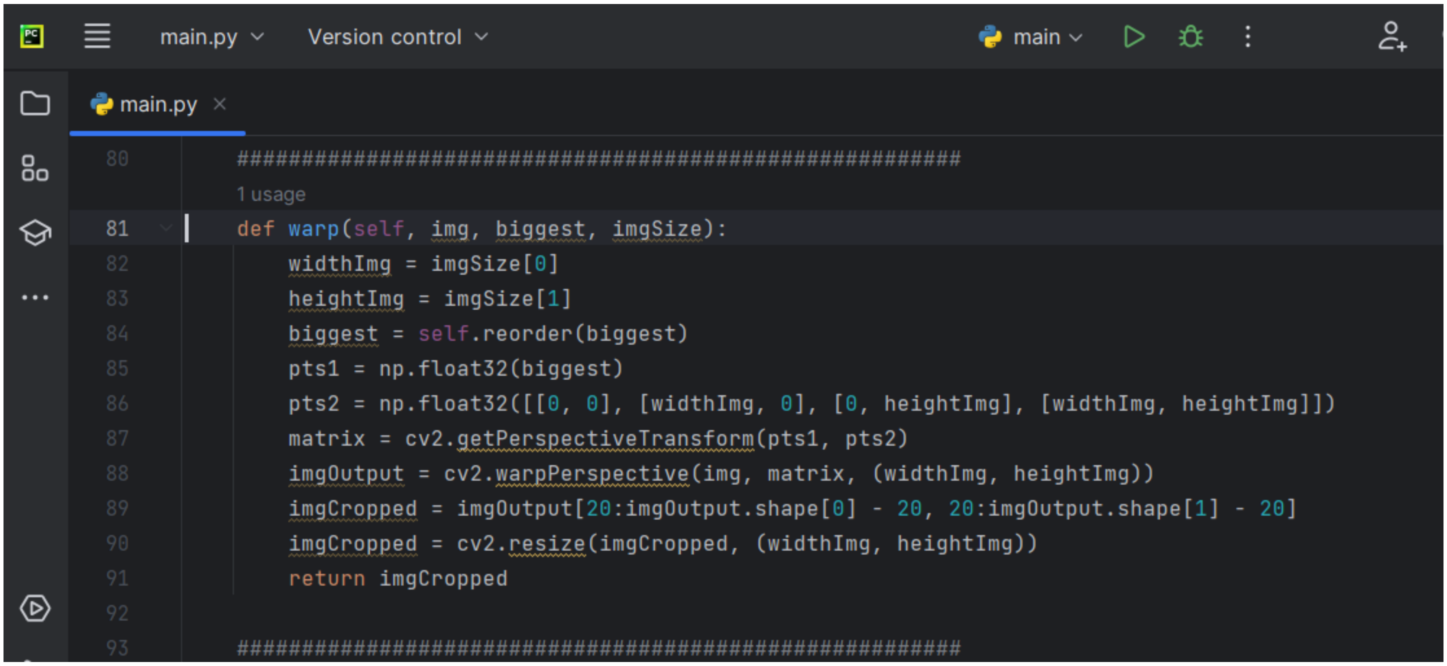
```python
    ########################################################
    def processing(self, img):
        imggray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        imgBlur = cv2.GaussianBlur(imggray, (5, 5), 1)
        imgCanny = cv2.Canny(imgBlur, 70, 100)
        kernel = np.ones((5, 5))
        imgDial = cv2.dilate(imgCanny, kernel, iterations=2)
        imgThres = cv2.erode(imgDial, kernel, iterations=1)

        cv2.imwrite('imggrayy.png', imggray)
        cv2.imwrite('imgblur.png', imgBlur)
        cv2.imwrite('imgcanny.png', imgCanny)
        return imgThres

    ########################################################
```



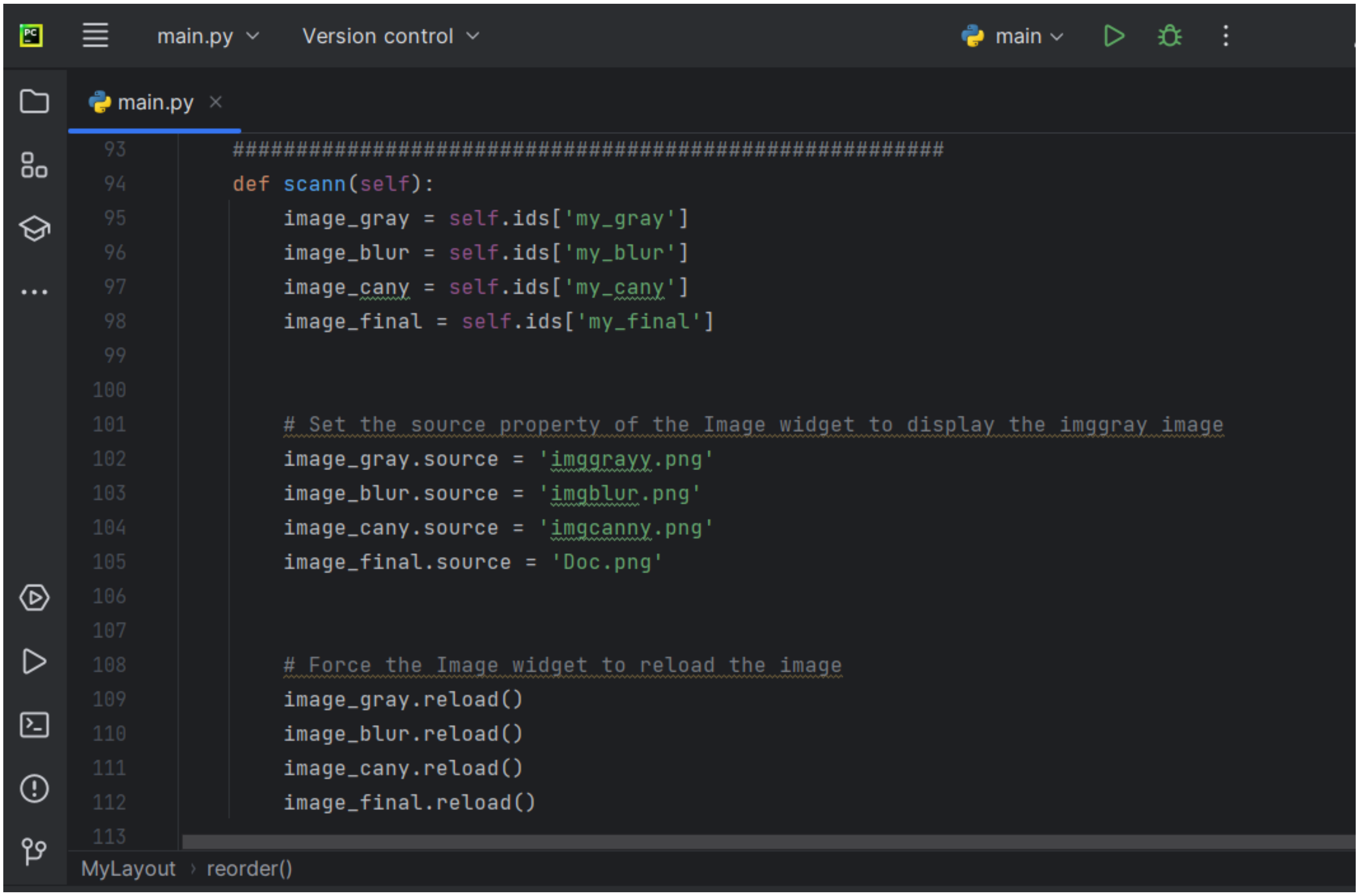
```python
    ########################################################
    def getContours(self, img):
        biggest = np.array([])
        maxArea = 0
        contours, hierarchy = cv2.findContours(img, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)
        for cnt in contours:
            area = cv2.contourArea(cnt)
            if area > 5000:
                peri = cv2.arcLength(cnt, True)
                approx = cv2.approxPolyDP(cnt, 0.02 * peri, True)
                if area > maxArea and len(approx) == 4:
                    biggest = approx
                    maxArea = area
        cv2.drawContours(imgContour, biggest, -1, (0, 0, 255), 30)
        return biggest

    ########################################################
```



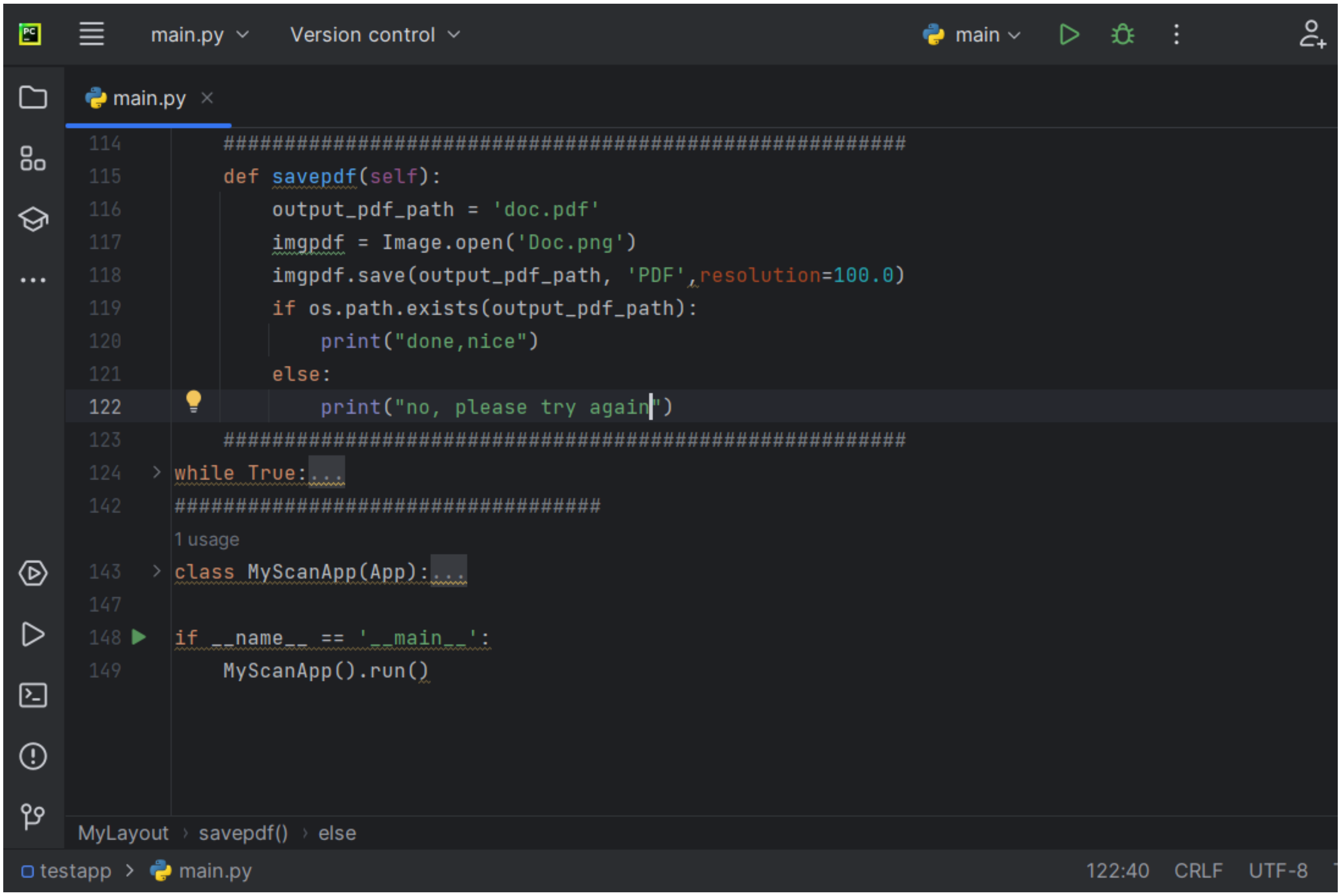
```python
    ########################################################
    def reorder(self, points):
        points = points.reshape((4, 2))
        newPoints = np.zeros((4, 1, 2), np.int32)
        add = points.sum(1)
        newPoints[0] = points[np.argmin(add)]
        newPoints[3] = points[np.argmax(add)]
        diff = np.diff(points, axis=1)
        newPoints[1] = points[np.argmin(diff)]
        newPoints[2] = points[np.argmax(diff)]
        return newPoints

    ########################################################
```



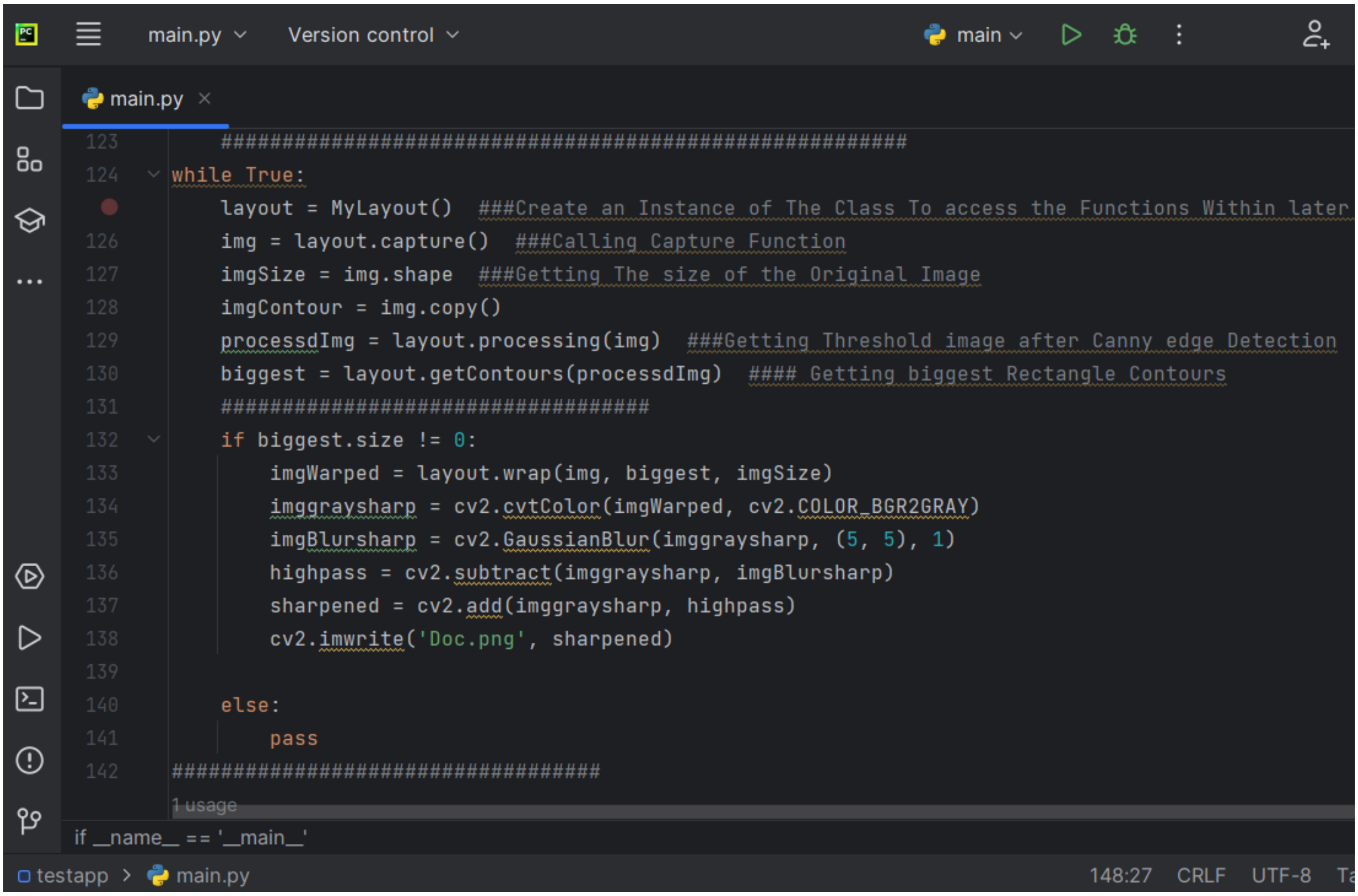
```python
    ########################################################

    def warp(self, img, biggest, imgSize):
        widthImg = imgSize[0]
        heightImg = imgSize[1]
        biggest = self.reorder(biggest)
        pts1 = np.float32(biggest)
        pts2 = np.float32([[0, 0], [widthImg, 0], [0, heightImg], [widthImg, heightImg]])
        matrix = cv2.getPerspectiveTransform(pts1, pts2)
        imgOutput = cv2.warpPerspective(img, matrix, (widthImg, heightImg))
        imgCropped = imgOutput[20:imgOutput.shape[0] - 20, 20:imgOutput.shape[1] - 20]
        imgCropped = cv2.resize(imgCropped, (widthImg, heightImg))
        return imgCropped

    ########################################################
```



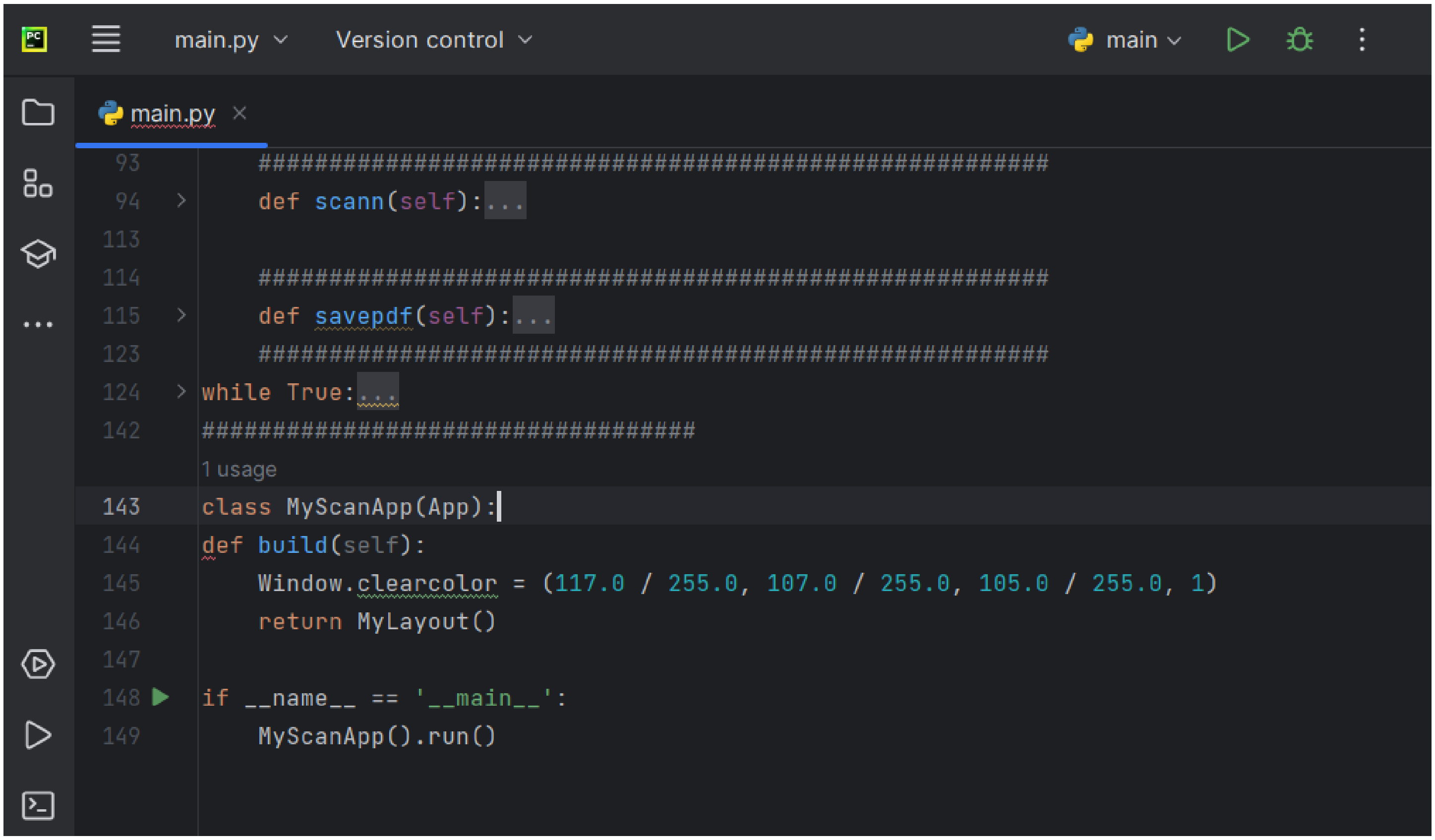
```python
    ########################################################
    def scann(self):
        image_gray = self.ids['my_gray']
        image_blur = self.ids['my_blur']
        image_cany = self.ids['my_cany']
        image_final = self.ids['my_final']


        # Set the source property of the Image widget to display the imggray image
        image_gray.source = 'imggrayy.png'
        image_blur.source = 'imgblur.png'
        image_cany.source = 'imgcanny.png'
        image_final.source = 'Doc.png'


        # Force the Image widget to reload the image
        image_gray.reload()
        image_blur.reload()
        image_cany.reload()
        image_final.reload()
```



```
    ########################################################
    def savepdf(self):
        output_pdf_path = 'doc.pdf'
        imgpdf = Image.open('Doc.png')
        imgpdf.save(output_pdf_path, 'PDF',resolution=100.0)
        if os.path.exists(output_pdf_path):
            print("done,nice")
        else:
            print("no, please try again")
    ########################################################
while True:...
###################################
1 usage
class MyScanApp(App):...

if __name__ == '__main__':
    MyScanApp().run()
```



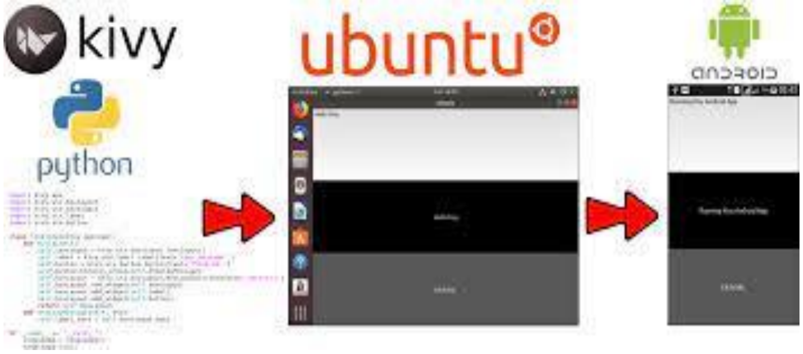
```
    ########################################################
while True:
    layout = MyLayout()  ###Create an Instance of The Class To access the Functions Within later
    img = layout.capture()  ###Calling Capture Function
    imgSize = img.shape  ###Getting The size of the Original Image
    imgContour = img.copy()
    processdImg = layout.processing(img)  ###Getting Threshold image after Canny edge Detection
    biggest = layout.getContours(processdImg)  #### Getting biggest Rectangle Contours
    ###################################
    if biggest.size != 0:
        imgWarped = layout.wrap(img, biggest, imgSize)
        imggraysharp = cv2.cvtColor(imgWarped, cv2.COLOR_BGR2GRAY)
        imgBlursharp = cv2.GaussianBlur(imggraysharp, (5, 5), 1)
        highpass = cv2.subtract(imggraysharp, imgBlursharp)
        sharpened = cv2.add(imggraysharp, highpass)
        cv2.imwrite('Doc.png', sharpened)

    else:
        pass
###################################
1 usage
```



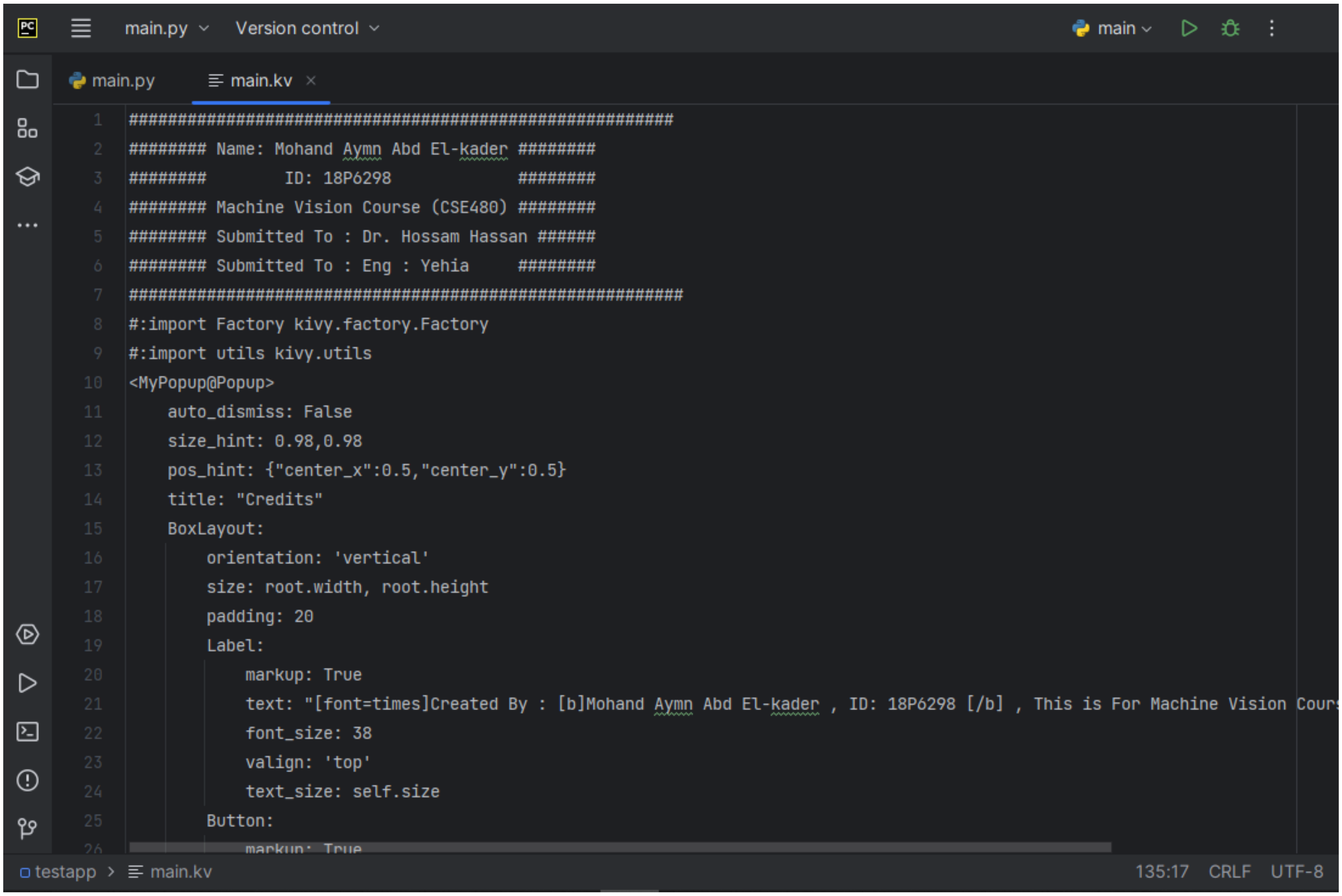
```
    ########################################################
    def scann(self):...

    ########################################################
    def savepdf(self):...
    ########################################################
while True:...
###################################
1 usage
class MyScanApp(App):
def build(self):
    Window.clearcolor = (117.0 / 255.0, 107.0 / 255.0, 105.0 / 255.0, 1)
    return MyLayout()

if __name__ == '__main__':
    MyScanApp().run()
```

Kivy App

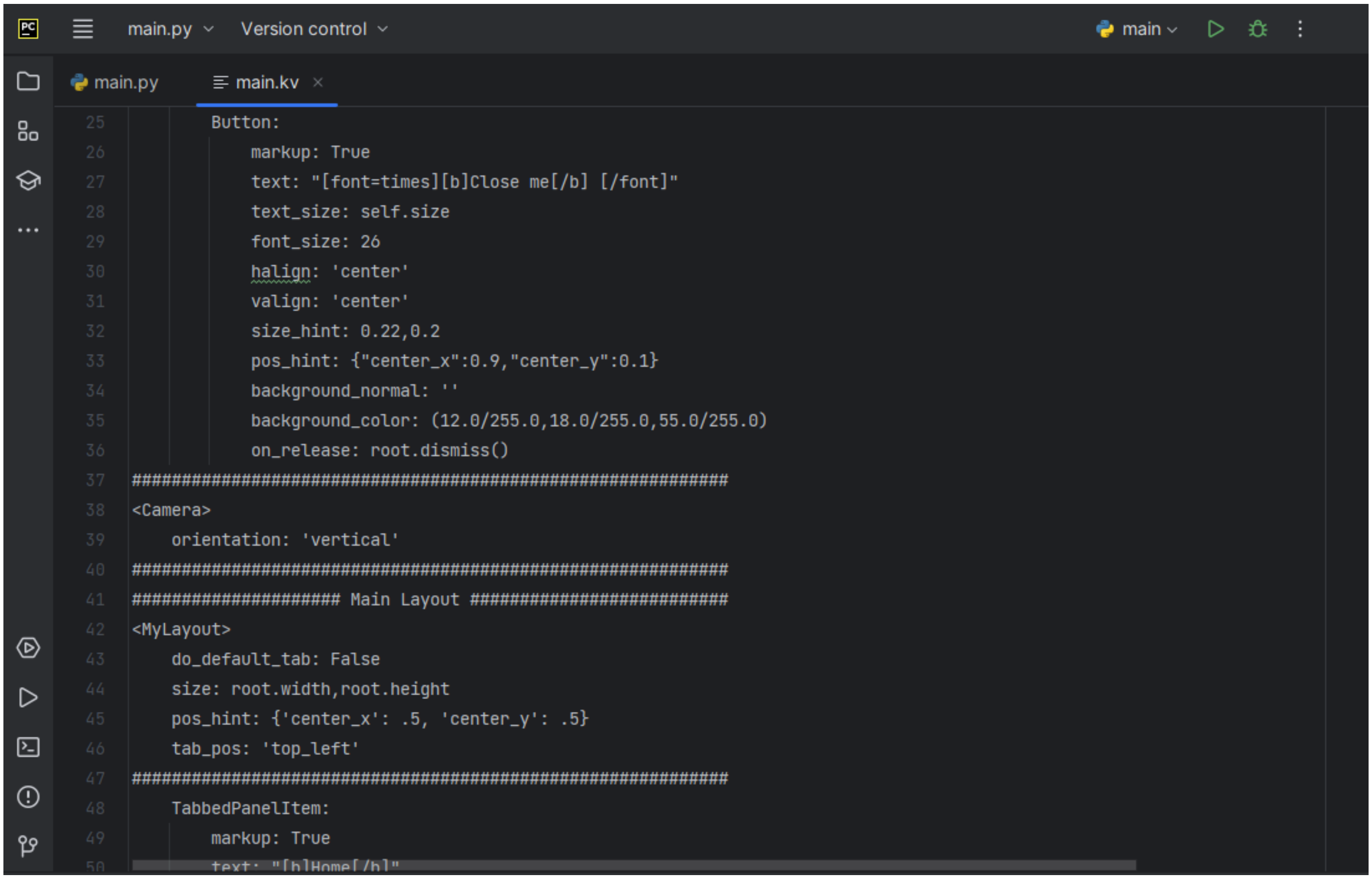
Using Kivy For building an app as it’s an open-source cross-platform python-app Development Framework and Although its an working windows App (.exe) but Used Buildozer For Debugging and Deployment of the python, kivy project to Convert it to a working APK (Android Package Kit) Through Linux OS.



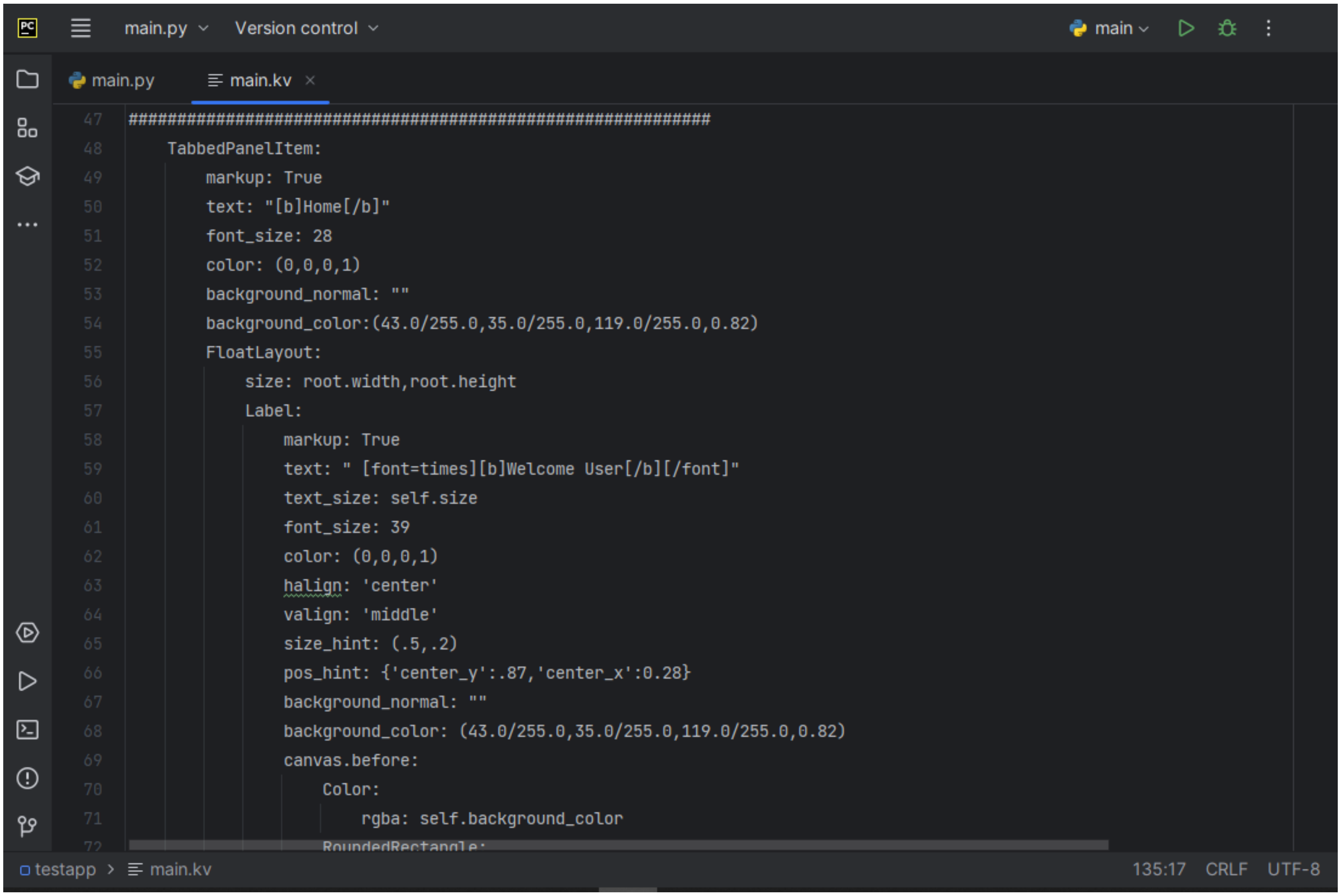
- **Snippets of Kivy file (main.kv):**



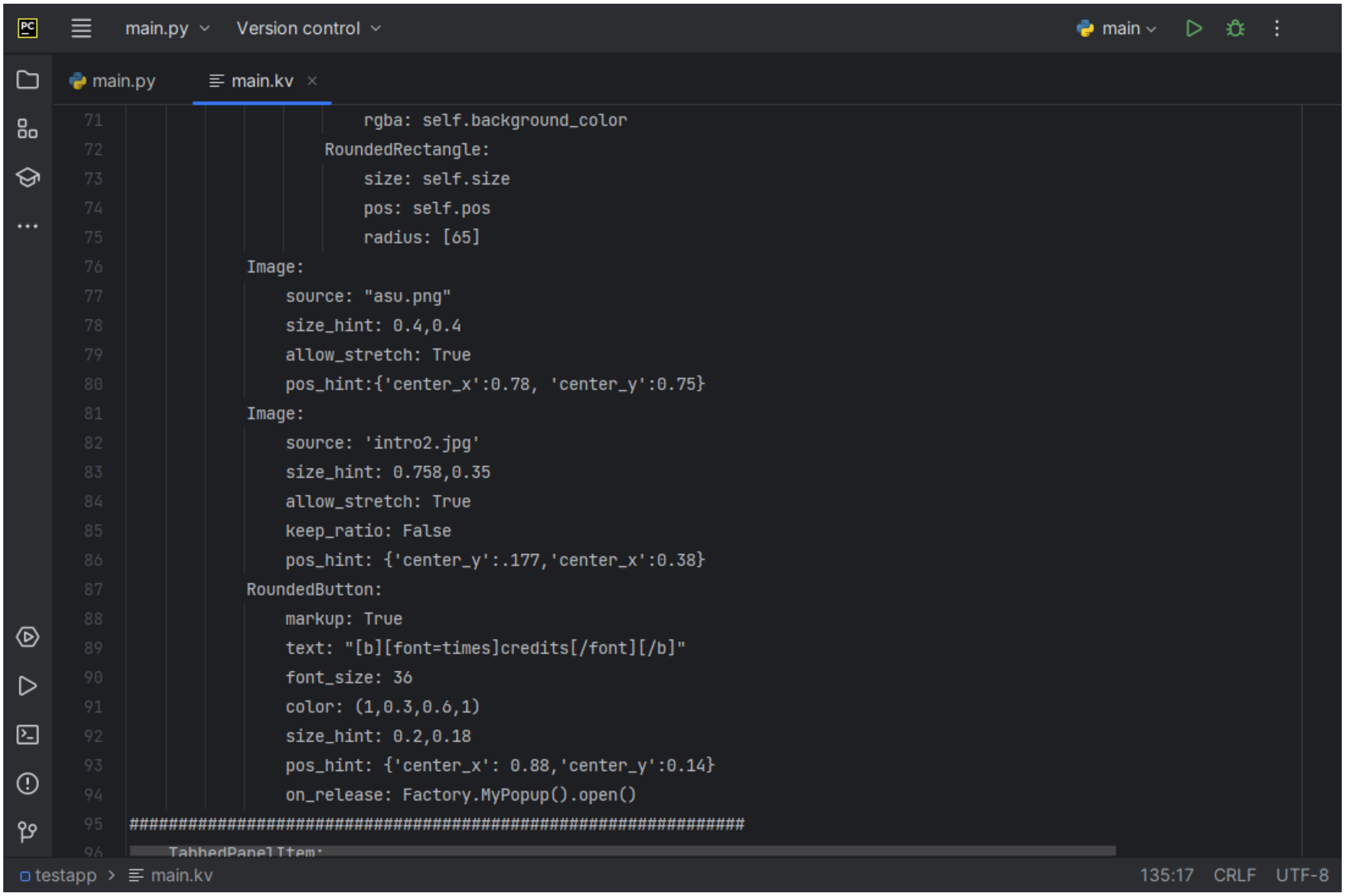
```
#########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
##########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
        size: root.width, root.height
        padding: 20
        Label:
            markup: True
            text: "[font=times]Created By : [b]Mohand Aymn Abd El-kader , ID: 18P6298 [/b] , This is For Machine Vision Cour
            font_size: 38
            valign: 'top'
            text_size: self.size
        Button:
            markup: True
```



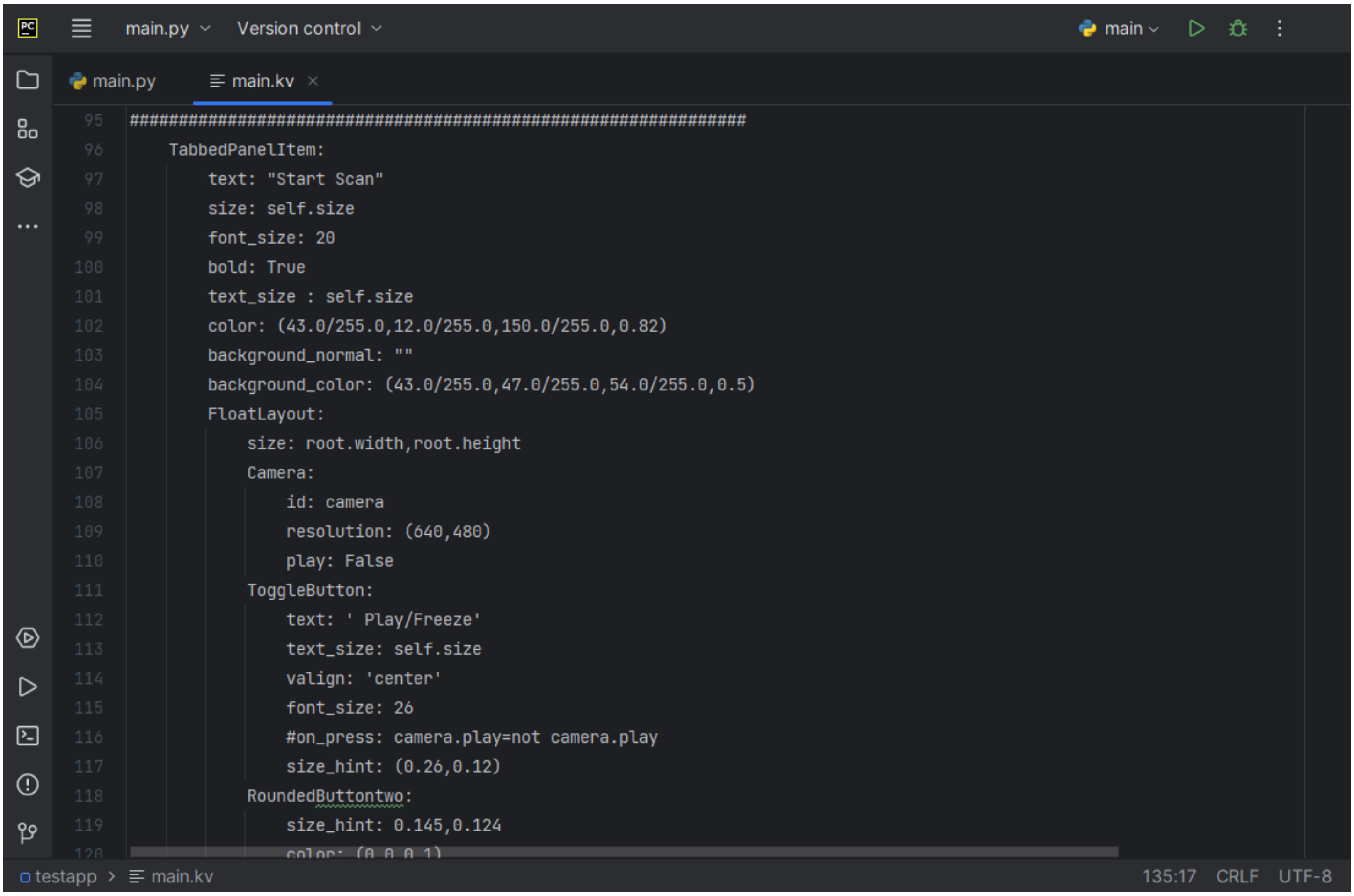
```
        Button:
            markup: True
            text: "[font=times][b]Close me[/b] [/font]"
            text_size: self.size
            font_size: 26
            halign: 'center'
            valign: 'center'
            size_hint: 0.22,0.2
            pos_hint: {"center_x":0.9,"center_y":0.1}
            background_normal: ''
            background_color: (12.0/255.0,18.0/255.0,55.0/255.0)
            on_release: root.dismiss()
##########################################################
<Camera>
    orientation: 'vertical'
##########################################################
#################### Main Layout #########################
<MyLayout>
    do_default_tab: False
    size: root.width,root.height
    pos_hint: {'center_x': .5, 'center_y': .5}
    tab_pos: 'top_left'
##########################################################
    TabbedPanelItem:
        markup: True
        text: "[b]Home[/b]"
```



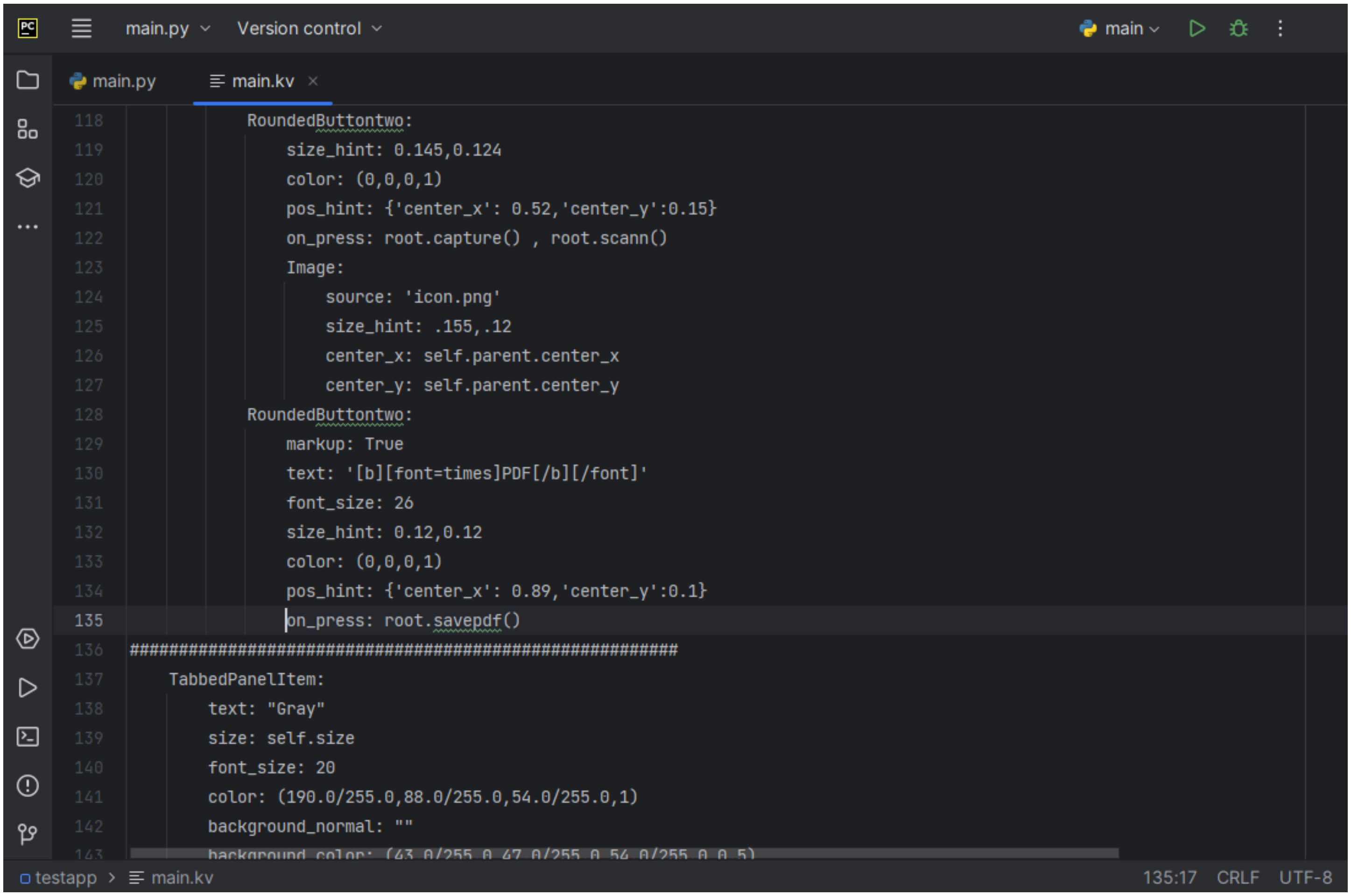
```
###########################################################
    TabbedPanelItem:
        markup: True
        text: "[b]Home[/b]"
        font_size: 28
        color: (0,0,0,1)
        background_normal: ""
        background_color:(43.0/255.0,35.0/255.0,119.0/255.0,0.82)
        FloatLayout:
            size: root.width,root.height
            Label:
                markup: True
                text: " [font=times][b]Welcome User[/b][/font]"
                text_size: self.size
                font_size: 39
                color: (0,0,0,1)
                halign: 'center'
                valign: 'middle'
                size_hint: (.5,.2)
                pos_hint: {'center_y':.87,'center_x':0.28}
                background_normal: ""
                background_color: (43.0/255.0,35.0/255.0,119.0/255.0,0.82)
                canvas.before:
                    Color:
                        rgba: self.background_color
                    RoundedRectangle:
```



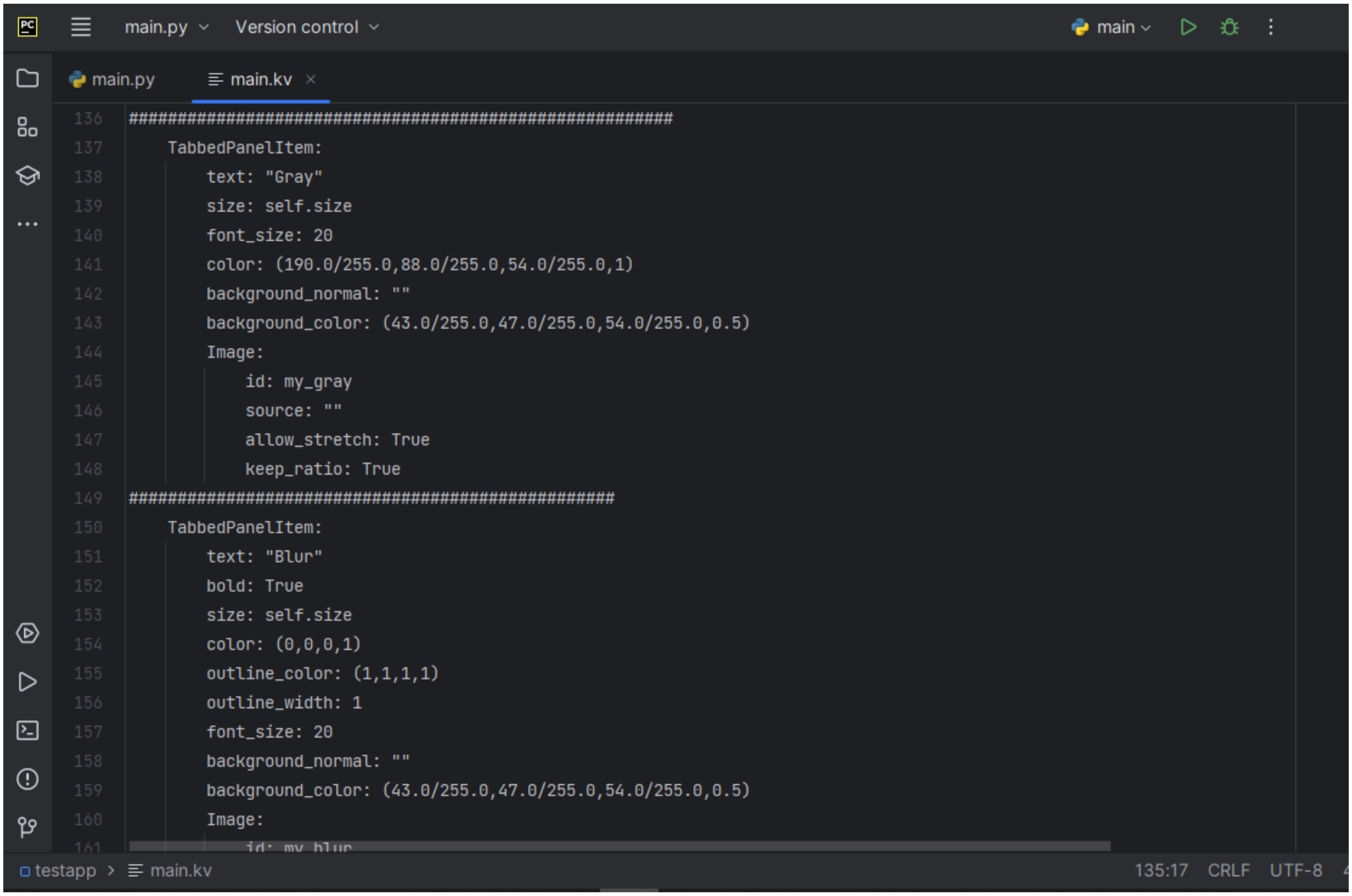
```
                        rgba: self.background_color
                    RoundedRectangle:
                        size: self.size
                        pos: self.pos
                        radius: [65]
            Image:
                source: "asu.png"
                size_hint: 0.4,0.4
                allow_stretch: True
                pos_hint:{'center_x':0.78, 'center_y':0.75}
            Image:
                source: 'intro2.jpg'
                size_hint: 0.758,0.35
                allow_stretch: True
                keep_ratio: False
                pos_hint: {'center_y':.177,'center_x':0.38}
            RoundedButton:
                markup: True
                text: "[b][font=times]credits[/font][/b]"
                font_size: 36
                color: (1,0.3,0.6,1)
                size_hint: 0.2,0.18
                pos_hint: {'center_x': 0.88,'center_y':0.14}
                on_release: Factory.MyPopup().open()
###############################################################
    TabbedPanelItem:
```



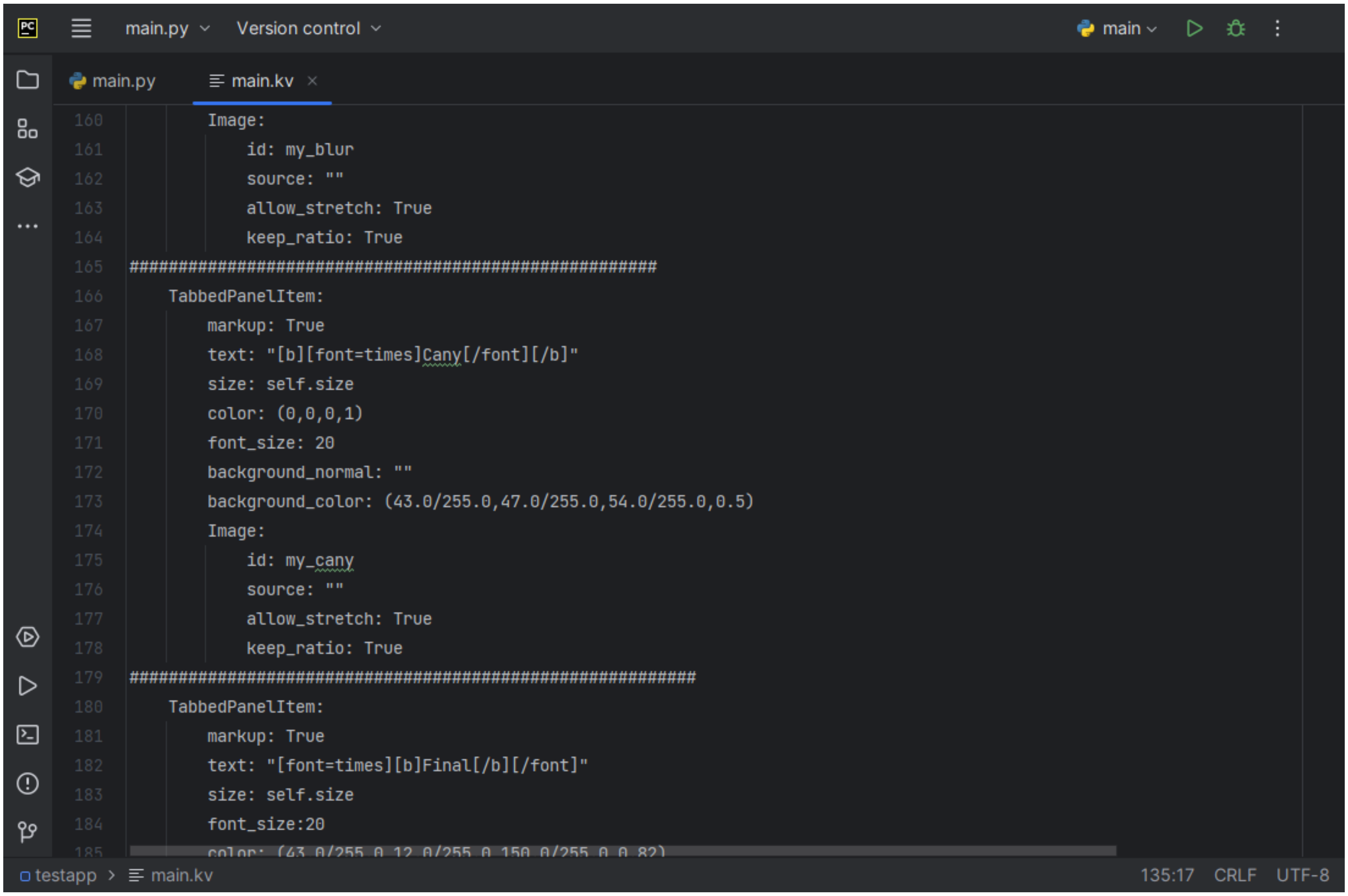
```
################################################################
    TabbedPanelItem:
        text: "Start Scan"
        size: self.size
        font_size: 20
        bold: True
        text_size : self.size
        color: (43.0/255.0,12.0/255.0,150.0/255.0,0.82)
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
        FloatLayout:
            size: root.width,root.height
            Camera:
                id: camera
                resolution: (640,480)
                play: False
            ToggleButton:
                text: ' Play/Freeze'
                text_size: self.size
                valign: 'center'
                font_size: 26
                #on_press: camera.play=not camera.play
                size_hint: (0.26,0.12)
            RoundedButtontwo:
                size_hint: 0.145,0.124
                color: (0,0,0,1)
```



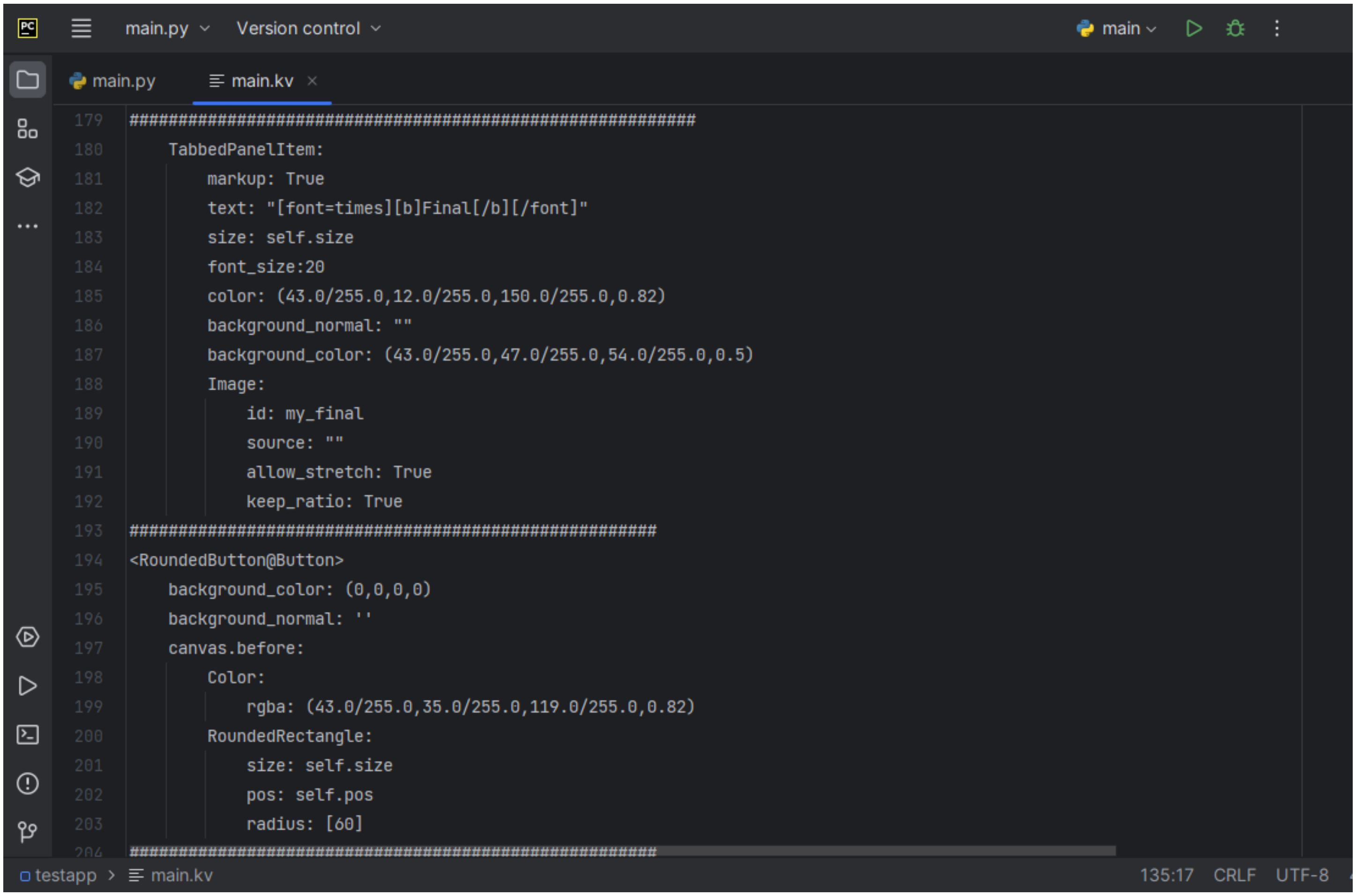
```
            RoundedButtontwo:
                size_hint: 0.145,0.124
                color: (0,0,0,1)
                pos_hint: {'center_x': 0.52,'center_y':0.15}
                on_press: root.capture() , root.scann()
                Image:
                    source: 'icon.png'
                    size_hint: .155,.12
                    center_x: self.parent.center_x
                    center_y: self.parent.center_y
            RoundedButtontwo:
                markup: True
                text: '[b][font=times]PDF[/b][/font]'
                font_size: 26
                size_hint: 0.12,0.12
                color: (0,0,0,1)
                pos_hint: {'center_x': 0.89,'center_y':0.1}
                on_press: root.savepdf()
#########################################################
    TabbedPanelItem:
        text: "Gray"
        size: self.size
        font_size: 20
        color: (190.0/255.0,88.0/255.0,54.0/255.0,1)
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
```



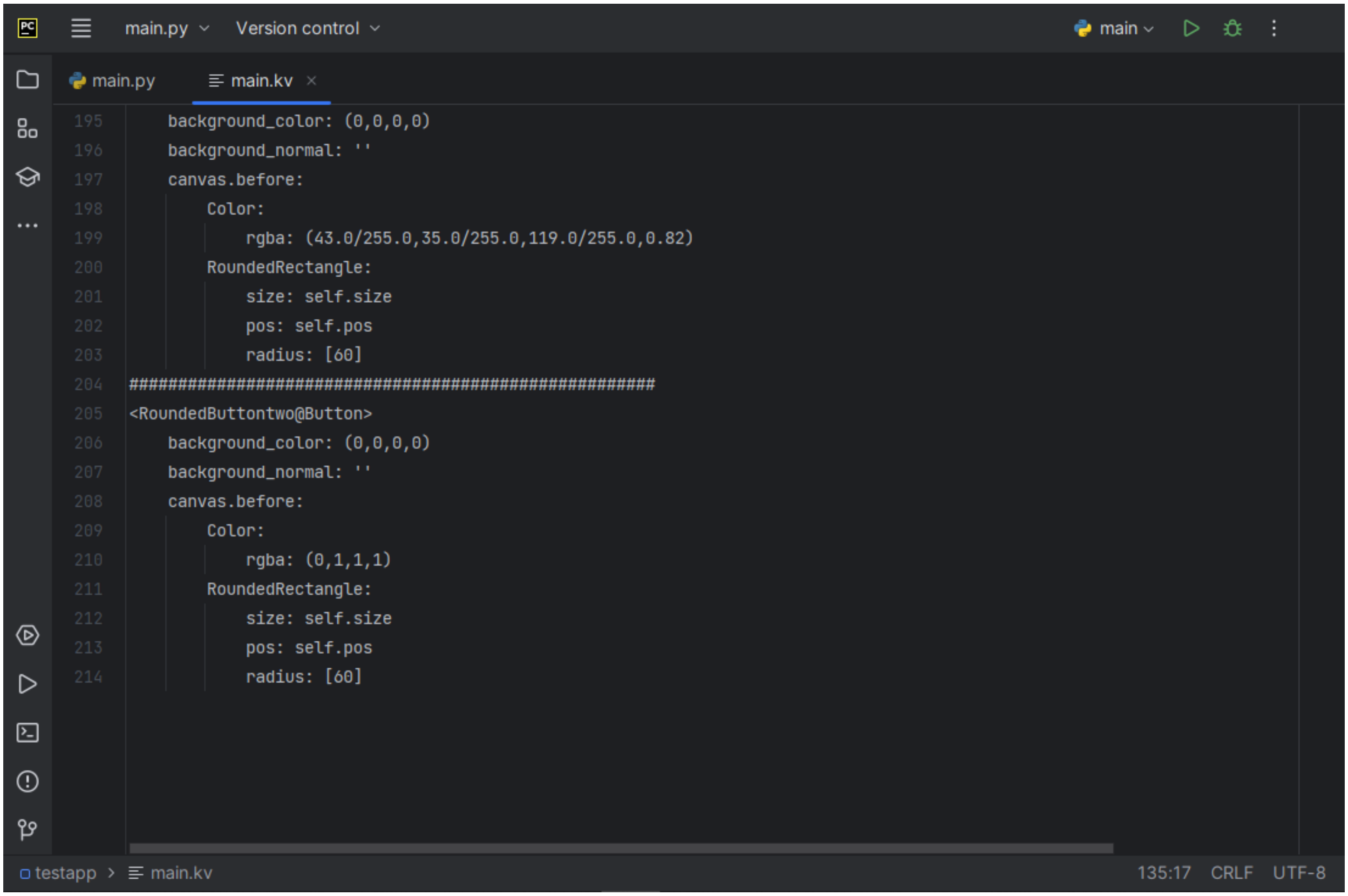
```
########################################################
    TabbedPanelItem:
        text: "Gray"
        size: self.size
        font_size: 20
        color: (190.0/255.0,88.0/255.0,54.0/255.0,1)
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
        Image:
            id: my_gray
            source: ""
            allow_stretch: True
            keep_ratio: True
##################################################
    TabbedPanelItem:
        text: "Blur"
        bold: True
        size: self.size
        color: (0,0,0,1)
        outline_color: (1,1,1,1)
        outline_width: 1
        font_size: 20
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
        Image:
            id: my_blur
```



```
        Image:
            id: my_blur
            source: ""
            allow_stretch: True
            keep_ratio: True
######################################################
    TabbedPanelItem:
        markup: True
        text: "[b][font=times]Cany[/font][/b]"
        size: self.size
        color: (0,0,0,1)
        font_size: 20
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
        Image:
            id: my_cany
            source: ""
            allow_stretch: True
            keep_ratio: True
##########################################################
    TabbedPanelItem:
        markup: True
        text: "[font=times][b]Final[/b][/font]"
        size: self.size
        font_size:20
        color: (43.0/255.0,12.0/255.0,150.0/255.0,0.82)
```



```
##########################################################
    TabbedPanelItem:
        markup: True
        text: "[font=times][b]Final[/b][/font]"
        size: self.size
        font_size:20
        color: (43.0/255.0,12.0/255.0,150.0/255.0,0.82)
        background_normal: ""
        background_color: (43.0/255.0,47.0/255.0,54.0/255.0,0.5)
        Image:
            id: my_final
            source: ""
            allow_stretch: True
            keep_ratio: True
######################################################
<RoundedButton@Button>
    background_color: (0,0,0,0)
    background_normal: ''
    canvas.before:
        Color:
            rgba: (43.0/255.0,35.0/255.0,119.0/255.0,0.82)
        RoundedRectangle:
            size: self.size
            pos: self.pos
            radius: [60]
######################################################
```

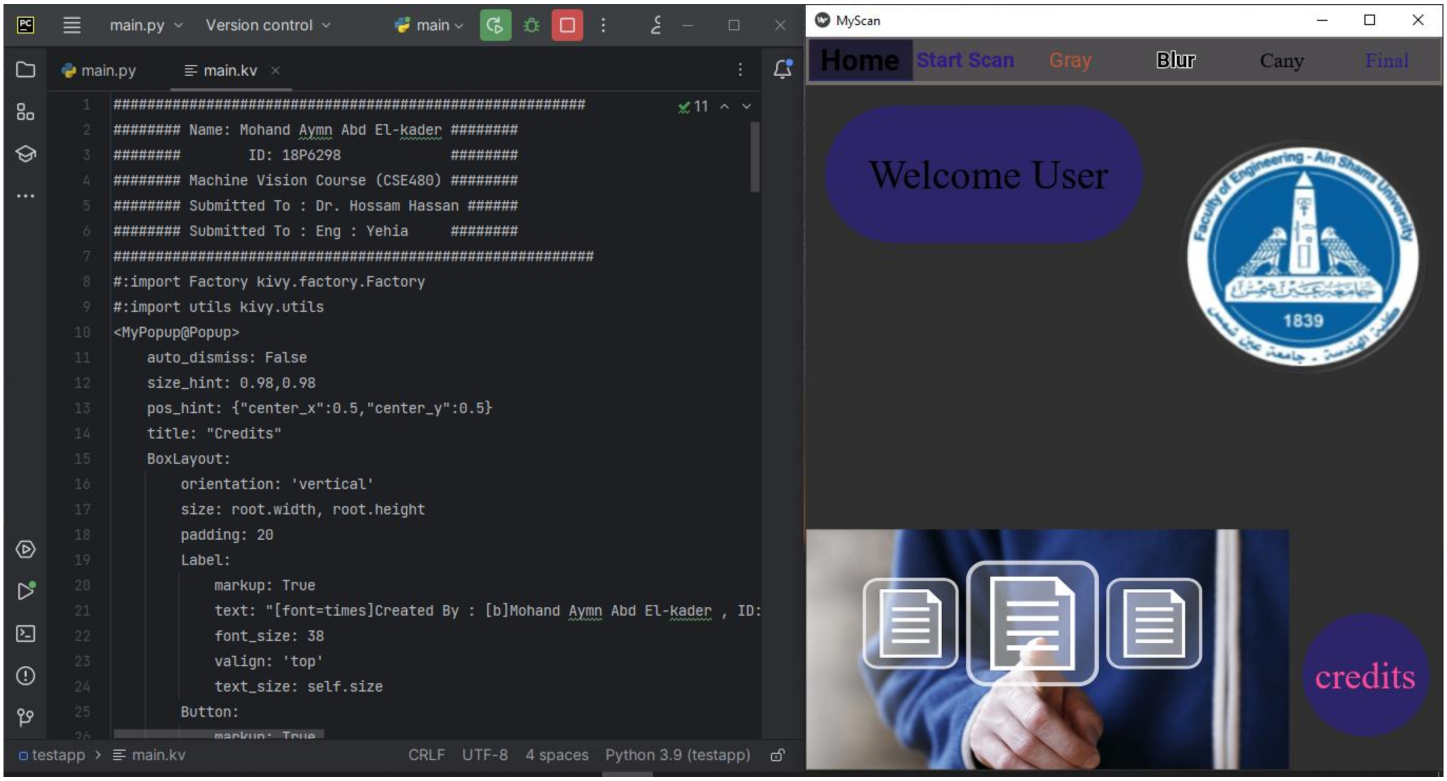


```
    background_color: (0,0,0,0)
    background_normal: ''
    canvas.before:
        Color:
            rgba: (43.0/255.0,35.0/255.0,119.0/255.0,0.82)
        RoundedRectangle:
            size: self.size
            pos: self.pos
            radius: [60]
```

```
######################################################
<RoundedButtontwo@Button>
    background_color: (0,0,0,0)
    background_normal: ''
    canvas.before:
        Color:
            rgba: (0,1,1,1)
        RoundedRectangle:
            size: self.size
            pos: self.pos
            radius: [60]
```

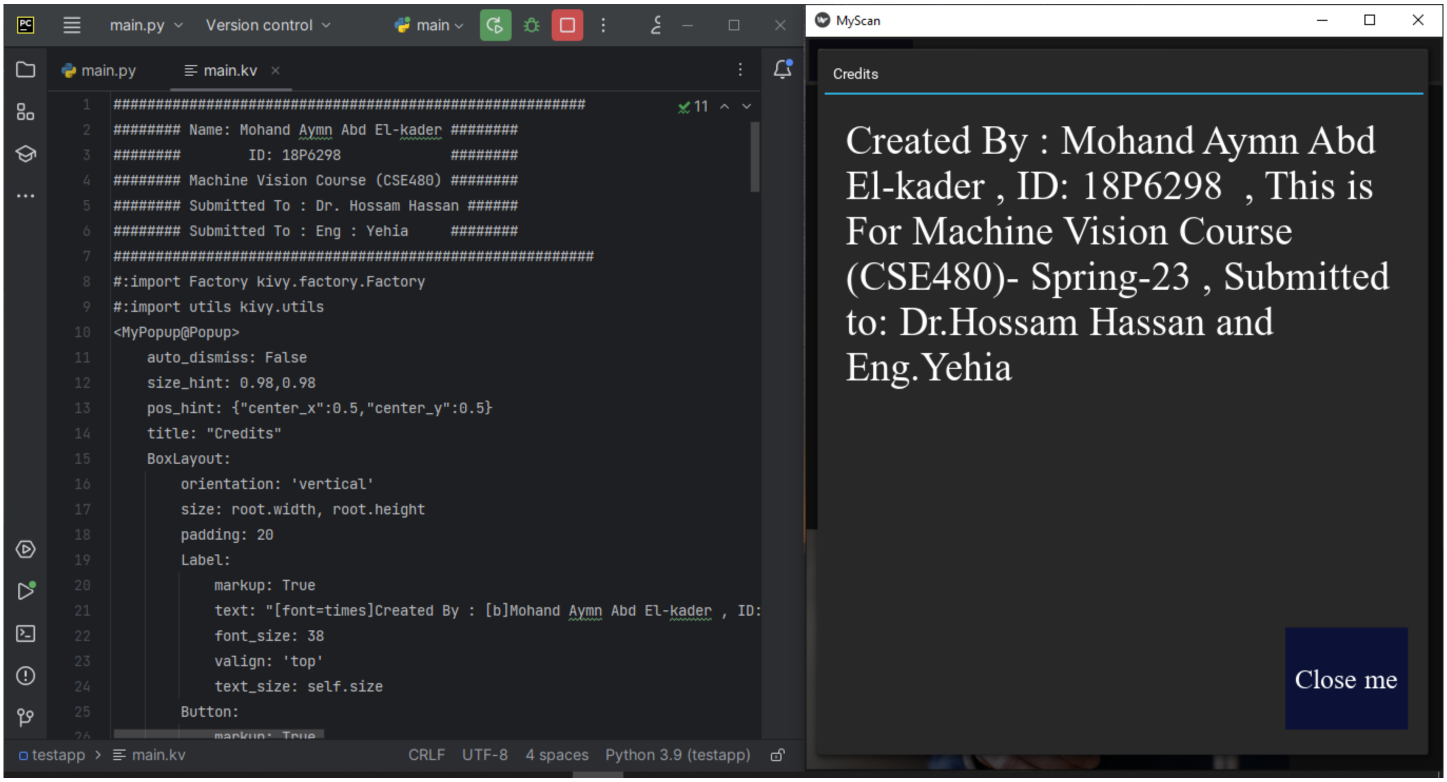
Kivy App Layout (Laptop)

- Home Tab



```
###########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298           ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
############################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
        size: root.width, root.height
        padding: 20
        Label:
            markup: True
            text: "[font=times]Created By : [b]Mohand Aymn Abd El-kader , ID:
            font_size: 38
            valign: 'top'
            text_size: self.size
        Button:
            markup: True
```

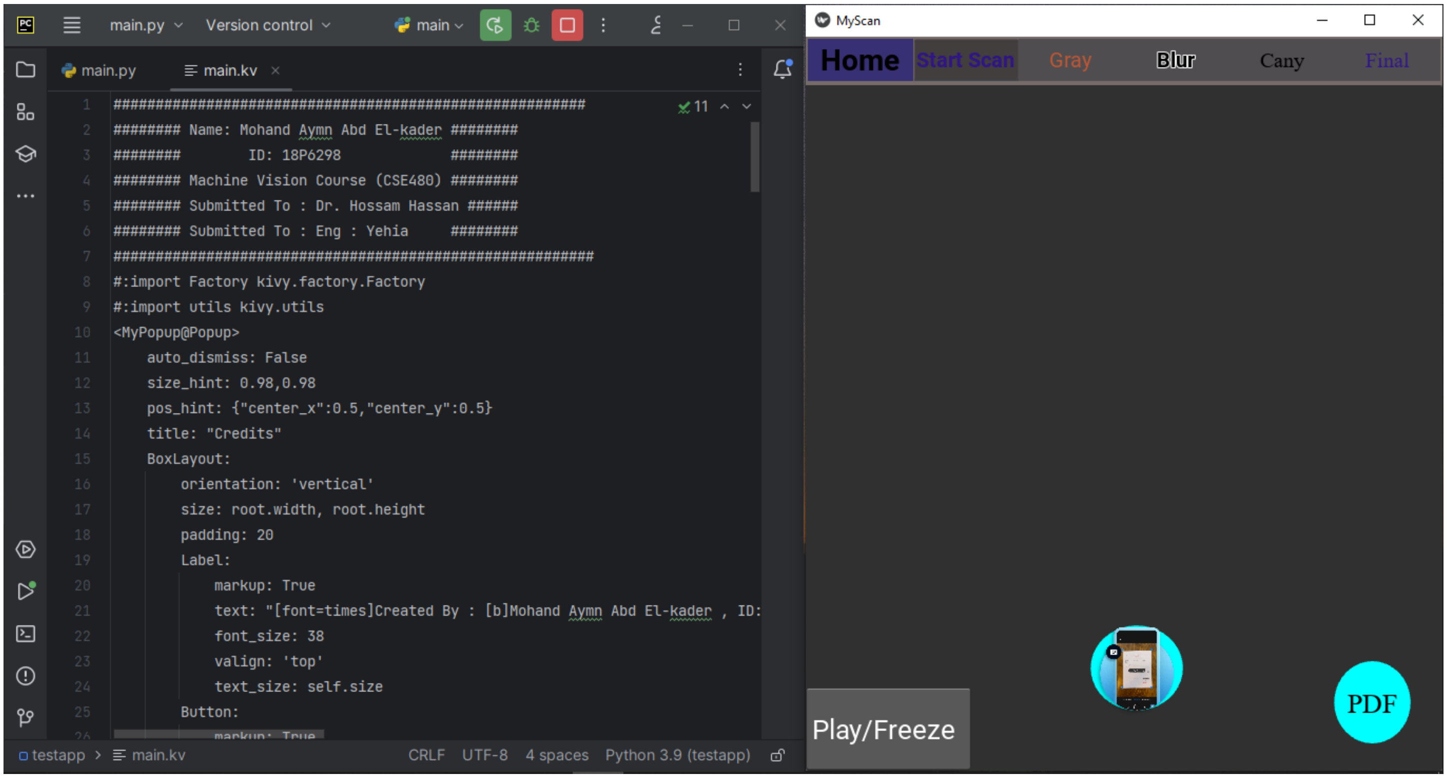
- **Credits Button opens a Pop-up Screen Containg Credits info.**



```
###########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298           ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
############################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
        size: root.width, root.height
        padding: 20
        Label:
            markup: True
            text: "[font=times]Created By : [b]Mohand Aymn Abd El-kader , ID:
            font_size: 38
            valign: 'top'
            text_size: self.size
        Button:
            markup: True
```

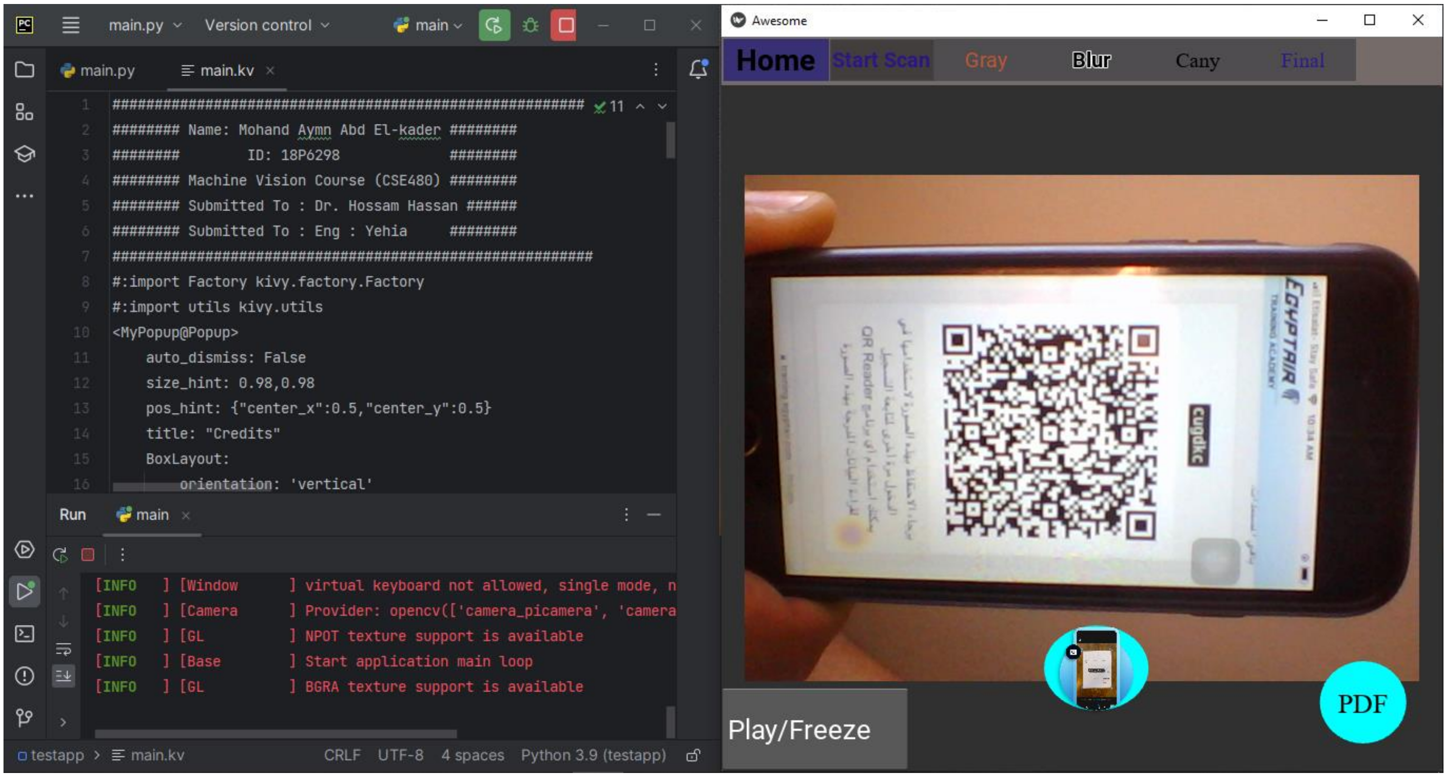
Created By : Mohand Aymn Abd El-kader , ID: 18P6298 , This is For Machine Vision Course (CSE480)- Spring-23 , Submitted to: Dr.Hossam Hassan and Eng.Yehia

- **Start Scan tab, Play/Freeze Button is a Toggle Button For Opening The Camera, The Logo is a Capture Button and The Pdf is For saving The Final processed and wrapped image as pdf**



```
########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
#########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
        size: root.width, root.height
        padding: 20
        Label:
            markup: True
            text: "[font=times]Created By : [b]Mohand Aymn Abd El-kader , ID:
            font_size: 38
            valign: 'top'
            text_size: self.size
        Button:
            markup: True
```

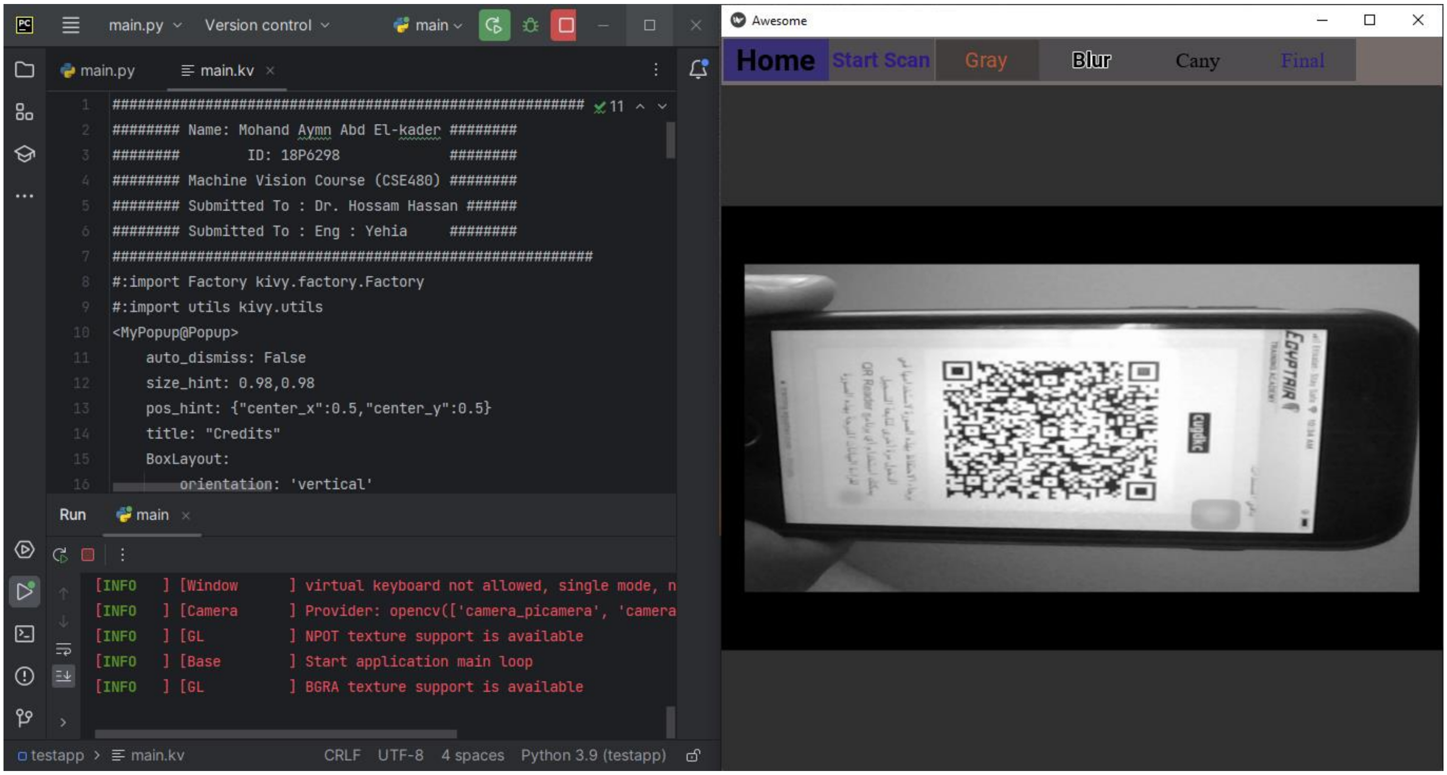
- **Testing The App**



```
########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
#########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
```

```
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [Camera      ] Provider: opencv(['camera_picamera', 'camera
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
[INFO   ] [GL          ] BGRA texture support is available
```

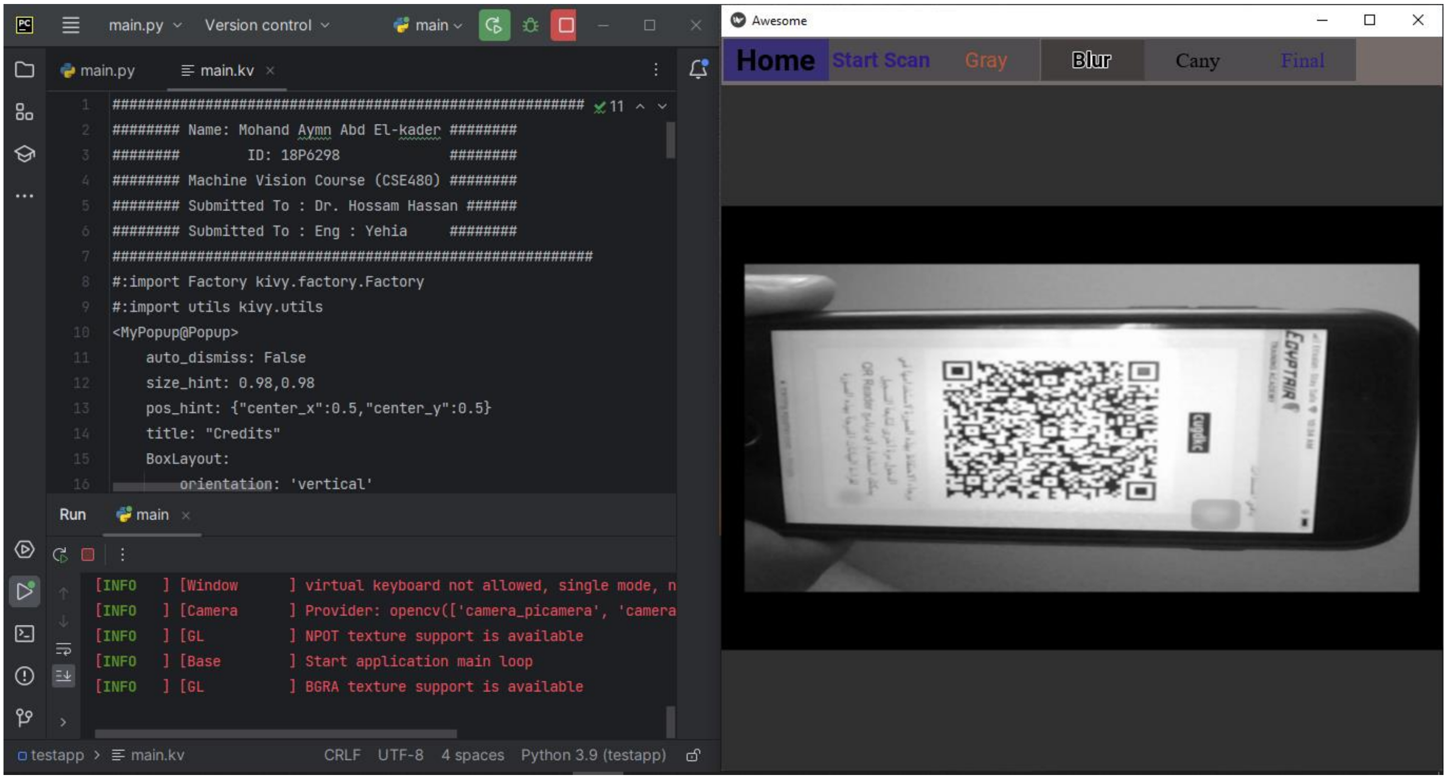
- **Process of Image Scanning (For Visualization and validation)**
- **Gray Image**



```
#########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298          ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
#########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
```

```
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [Camera      ] Provider: opencv(['camera_picamera', 'camera
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
[INFO   ] [GL          ] BGRA texture support is available
```

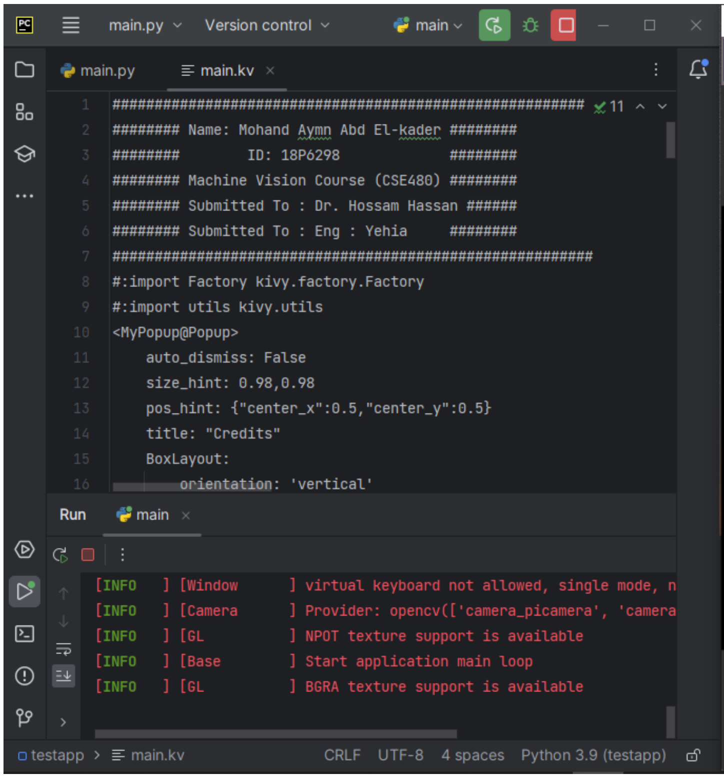
- **Blur Image**



```
#########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298          ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
#########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
```

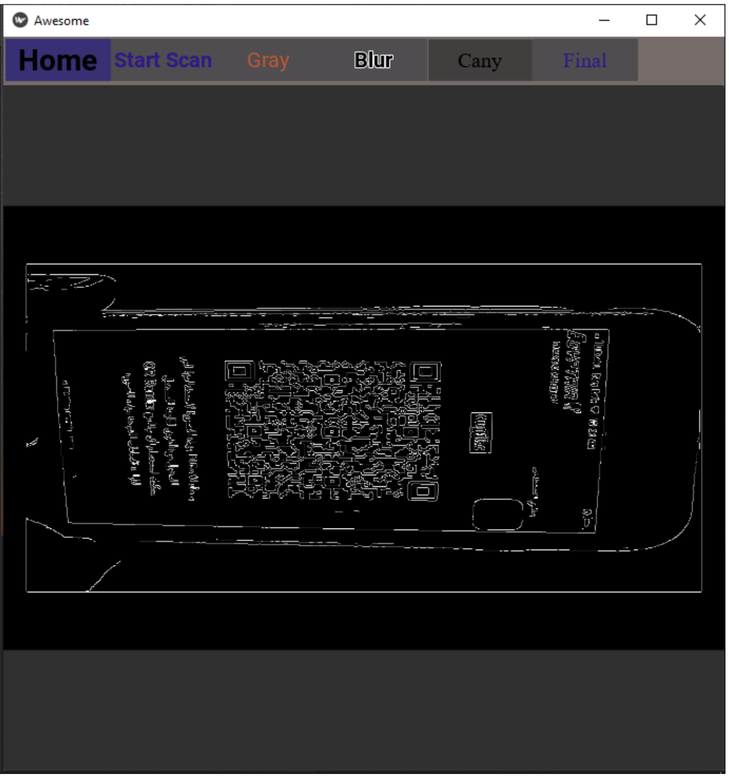
```
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [Camera      ] Provider: opencv(['camera_picamera', 'camera
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
[INFO   ] [GL          ] BGRA texture support is available
```

- **Canny Edge Detection**

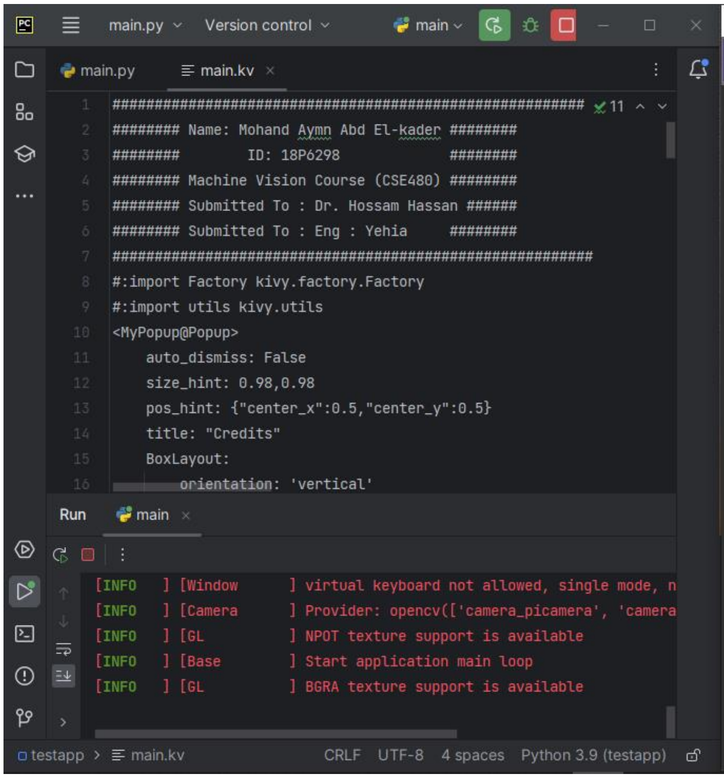


```
##########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
##########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
```

```
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [Camera      ] Provider: opencv(['camera_picamera', 'camera
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
[INFO   ] [GL          ] BGRA texture support is available
```

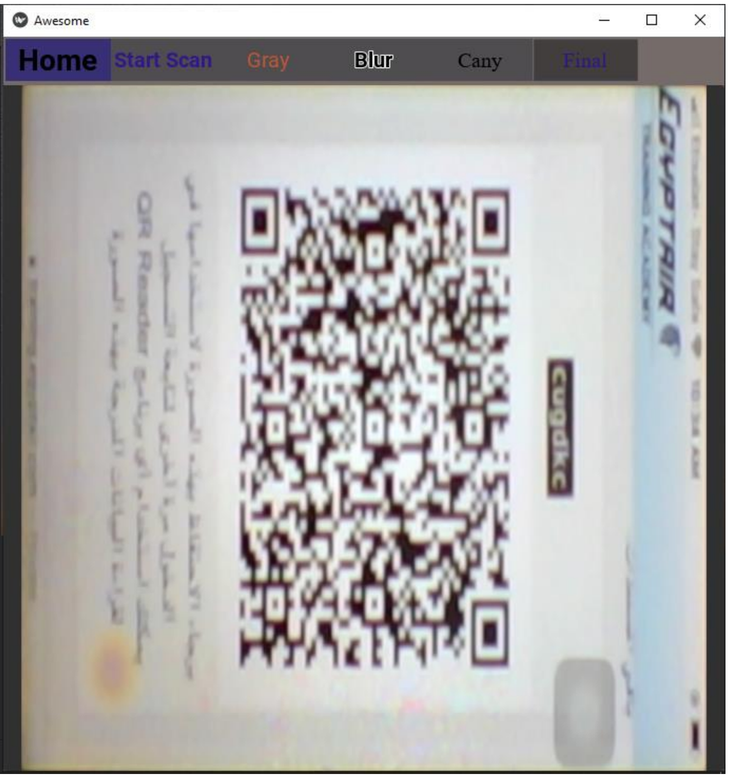


- **Result**



```
##########################################################
######## Name: Mohand Aymn Abd El-kader ########
########        ID: 18P6298             ########
######## Machine Vision Course (CSE480) ########
######## Submitted To : Dr. Hossam Hassan ######
######## Submitted To : Eng : Yehia     ########
##########################################################
#:import Factory kivy.factory.Factory
#:import utils kivy.utils
<MyPopup@Popup>
    auto_dismiss: False
    size_hint: 0.98,0.98
    pos_hint: {"center_x":0.5,"center_y":0.5}
    title: "Credits"
    BoxLayout:
        orientation: 'vertical'
```

```
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [Camera      ] Provider: opencv(['camera_picamera', 'camera
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
[INFO   ] [GL          ] BGRA texture support is available
```

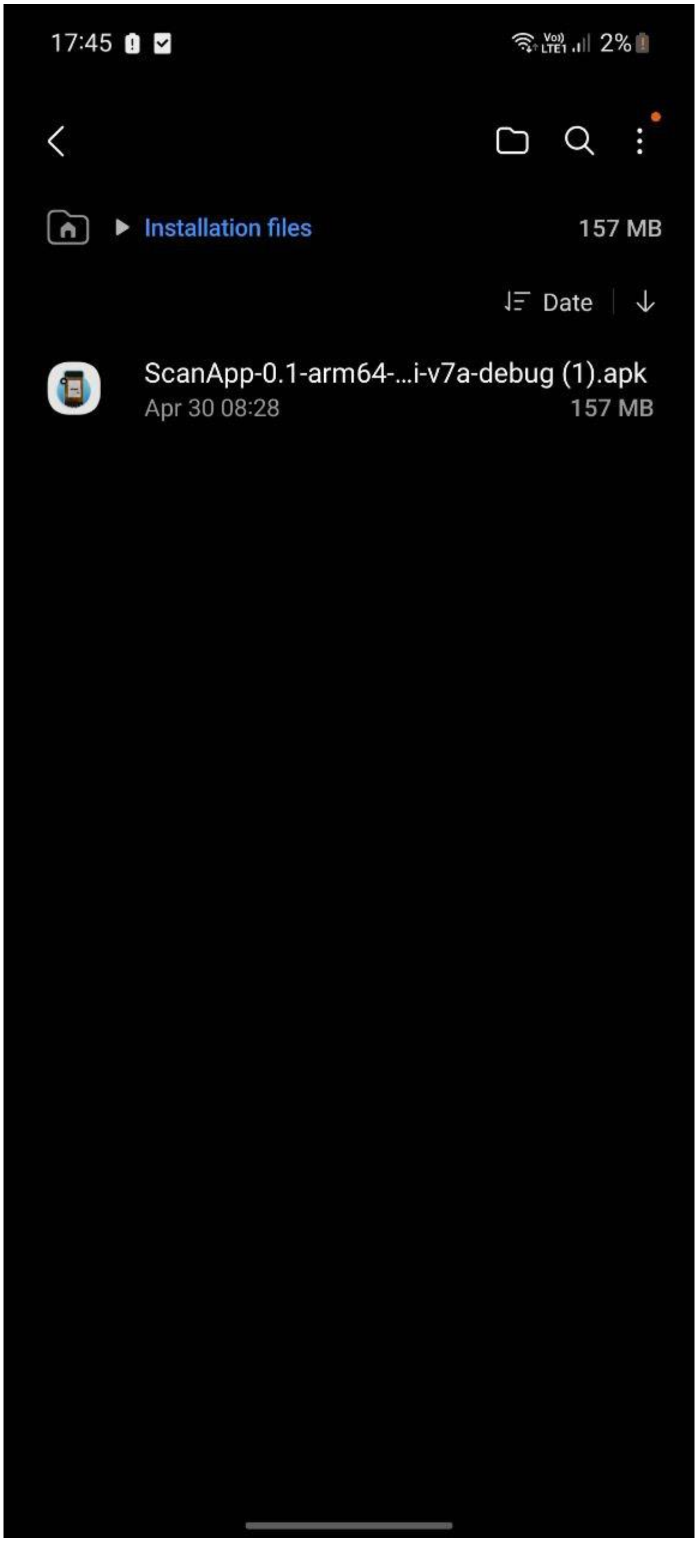


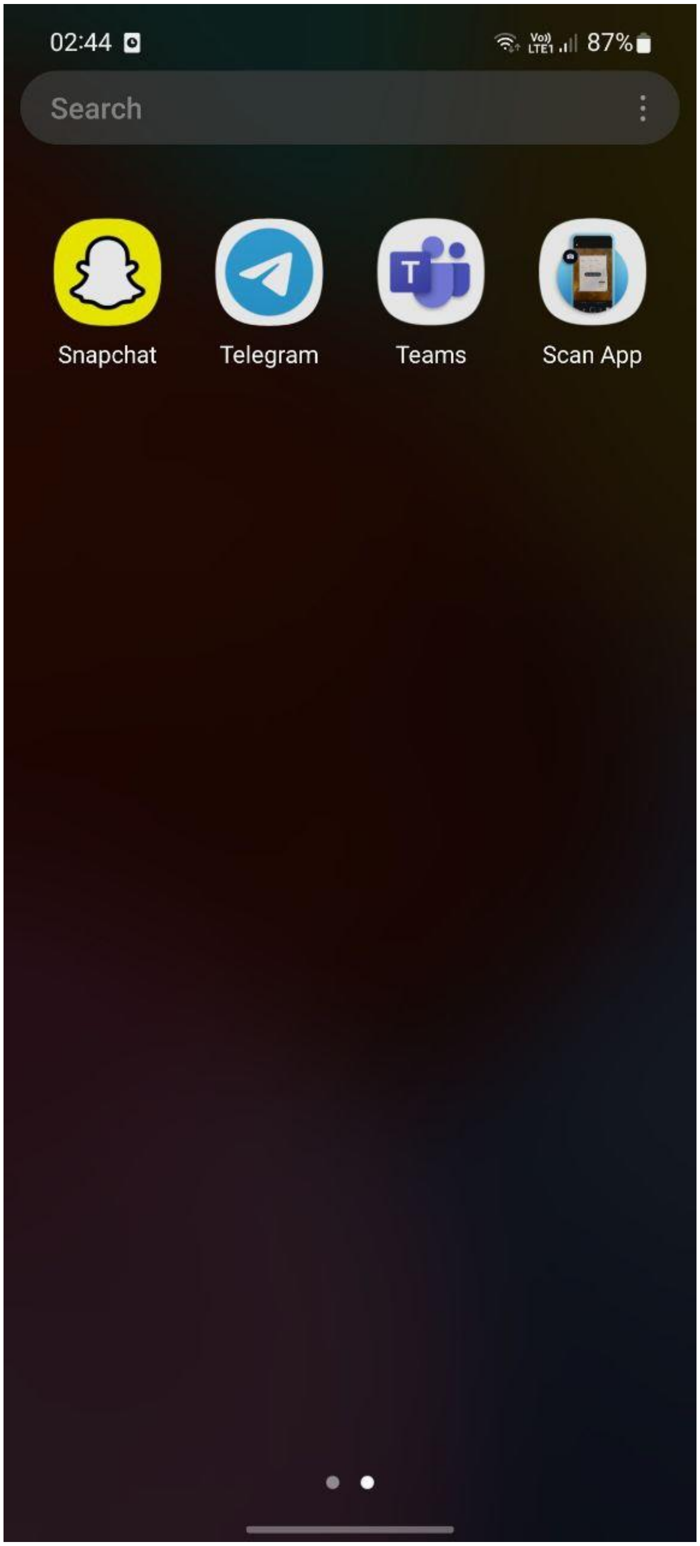
- Saved as Pdf ➔ 🖫 🗁

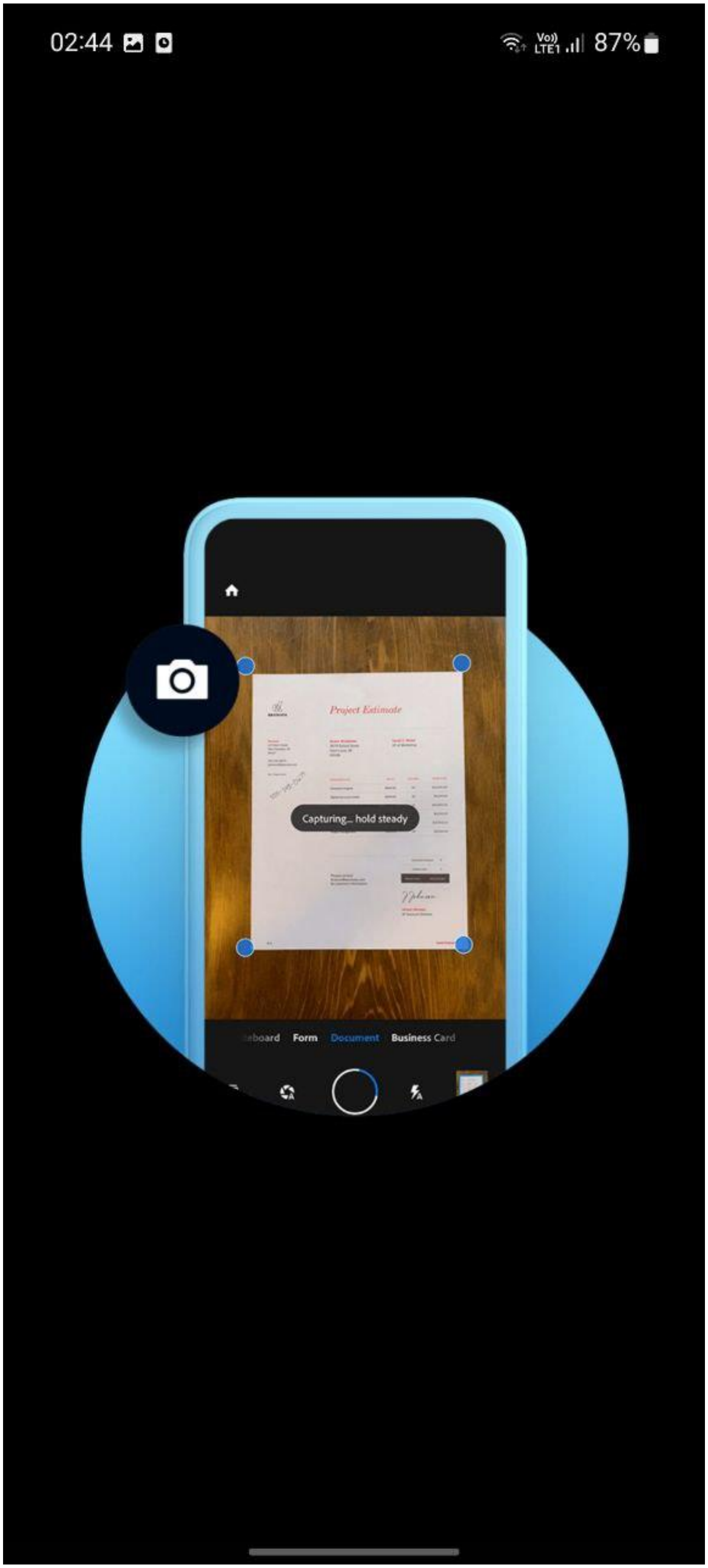
doc.pdf
Doc.png

Phone App Layout

- After Debugging and configuring the (**buildozer.spec** file) with Buildozer on linux Os to Get an APk for Android Devices (Caution: it Installs Correctly but it Worked few Times But it often Crashes with Different Android System so might not Work on Different API (versions of Android))

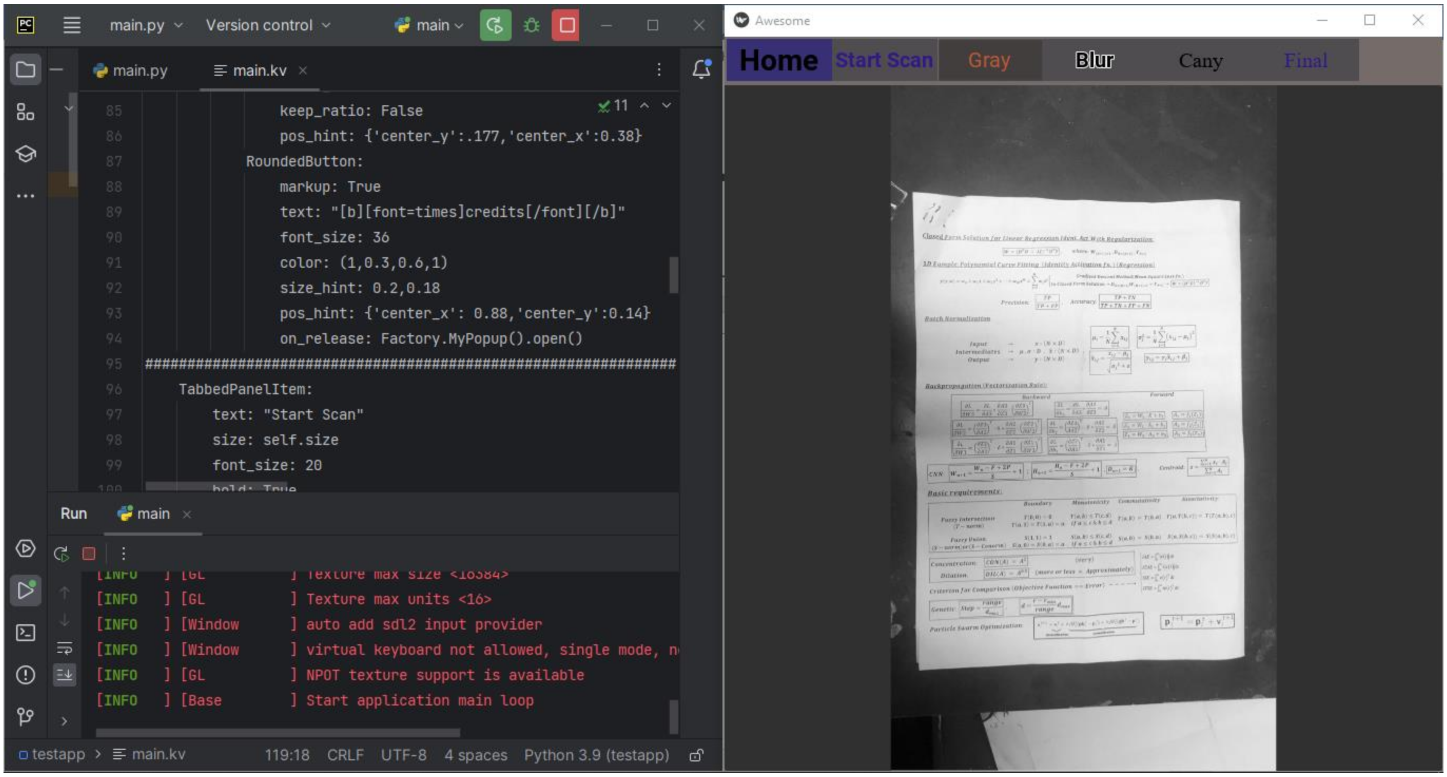





02:44
87%
Project Estimate
Capturing... hold steady
board
Form
Document
Business Card

Random Results From Results File:

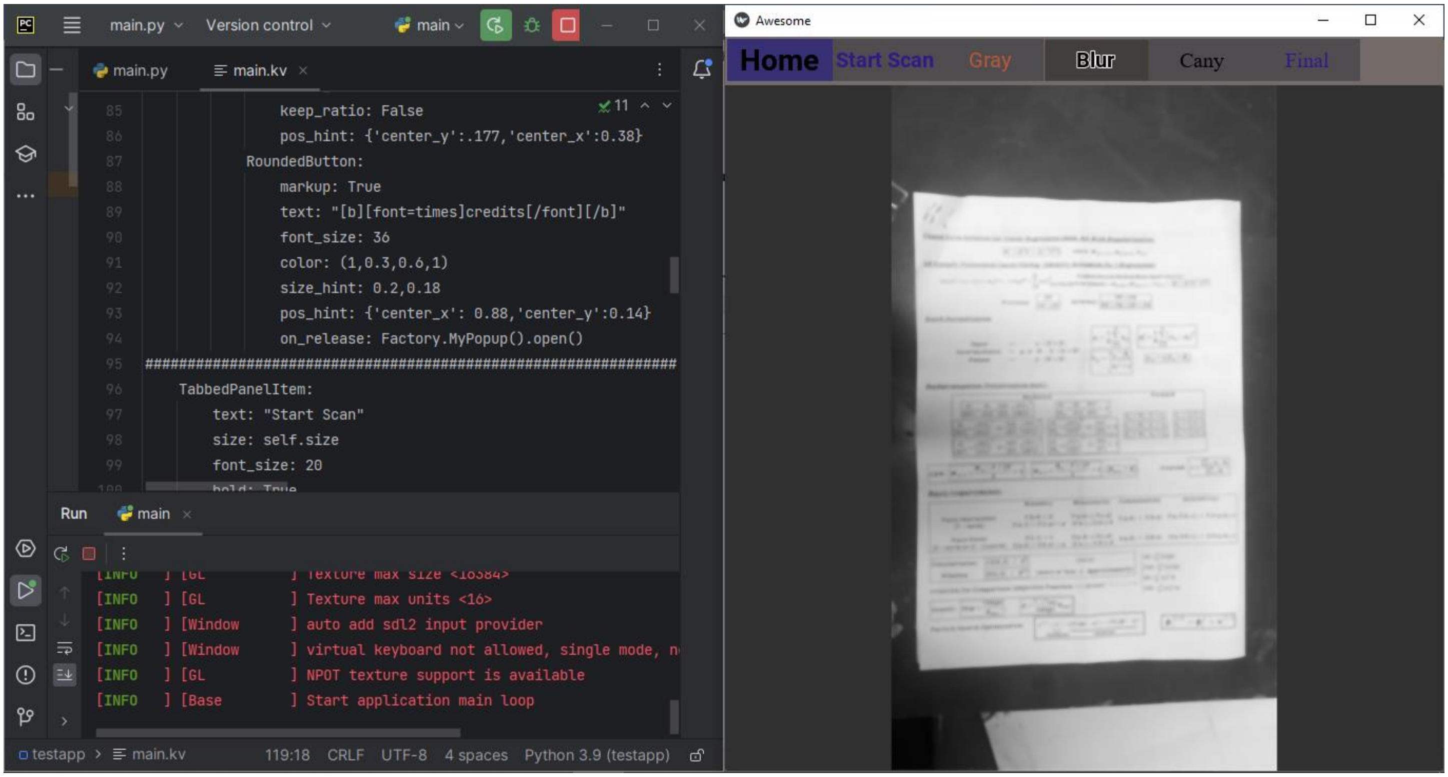
- **First test sample**



```
            keep_ratio: False
            pos_hint: {'center_y':.177,'center_x':0.38}
        RoundedButton:
            markup: True
            text: "[b][font=times]credits[/font][/b]"
            font_size: 36
            color: (1,0.3,0.6,1)
            size_hint: 0.2,0.18
            pos_hint: {'center_x': 0.88,'center_y':0.14}
            on_release: Factory.MyPopup().open()
###############################################################
    TabbedPanelItem:
        text: "Start Scan"
        size: self.size
        font_size: 20
        bold: True
```

```
[INFO   ] [GL          ] Texture max size <16384>
[INFO   ] [GL          ] Texture max units <16>
[INFO   ] [Window      ] auto add sdl2 input provider
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
```

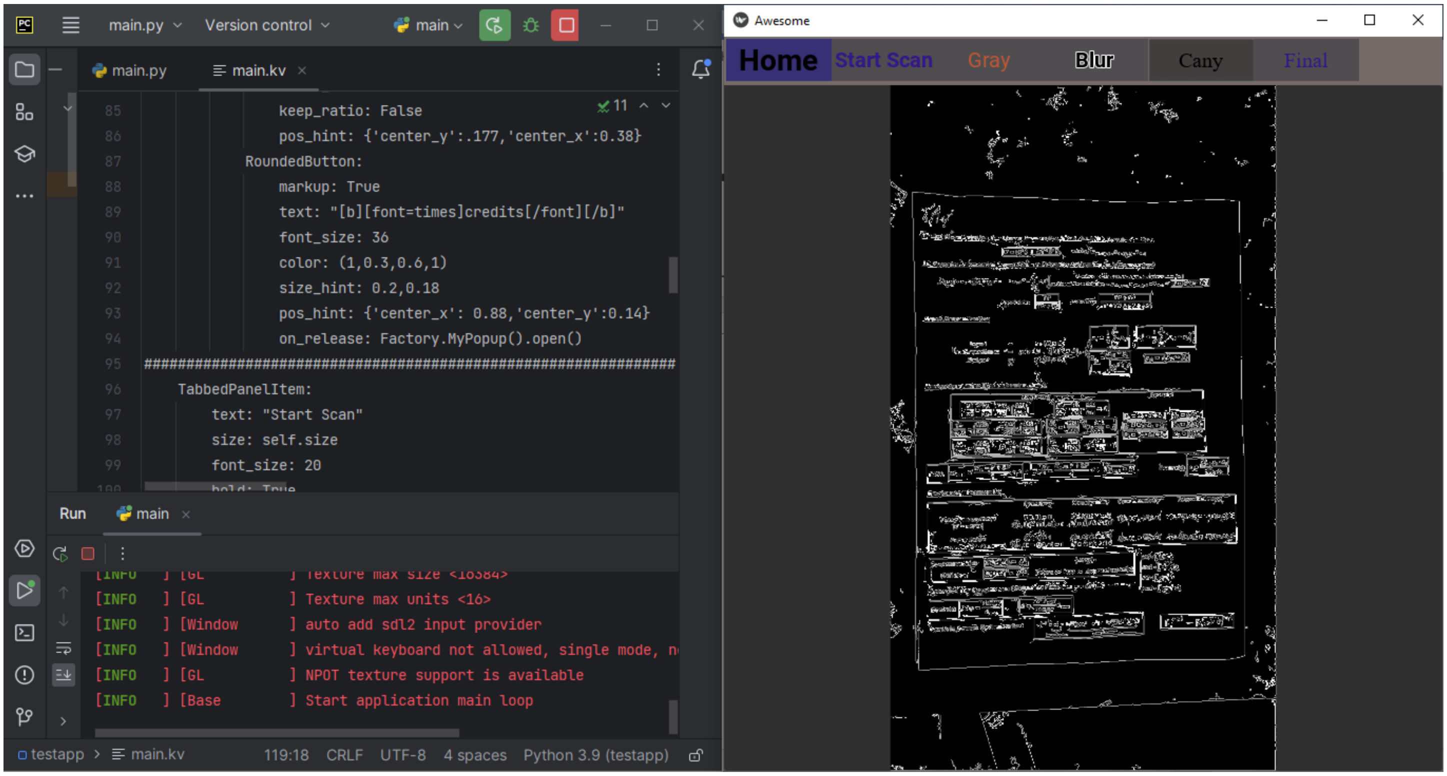
testapp > main.kv 119:18 CRLF UTF-8 4 spaces Python 3.9 (testapp)



```
            keep_ratio: False
            pos_hint: {'center_y':.177,'center_x':0.38}
        RoundedButton:
            markup: True
            text: "[b][font=times]credits[/font][/b]"
            font_size: 36
            color: (1,0.3,0.6,1)
            size_hint: 0.2,0.18
            pos_hint: {'center_x': 0.88,'center_y':0.14}
            on_release: Factory.MyPopup().open()
###############################################################
    TabbedPanelItem:
        text: "Start Scan"
        size: self.size
        font_size: 20
        bold: True
```

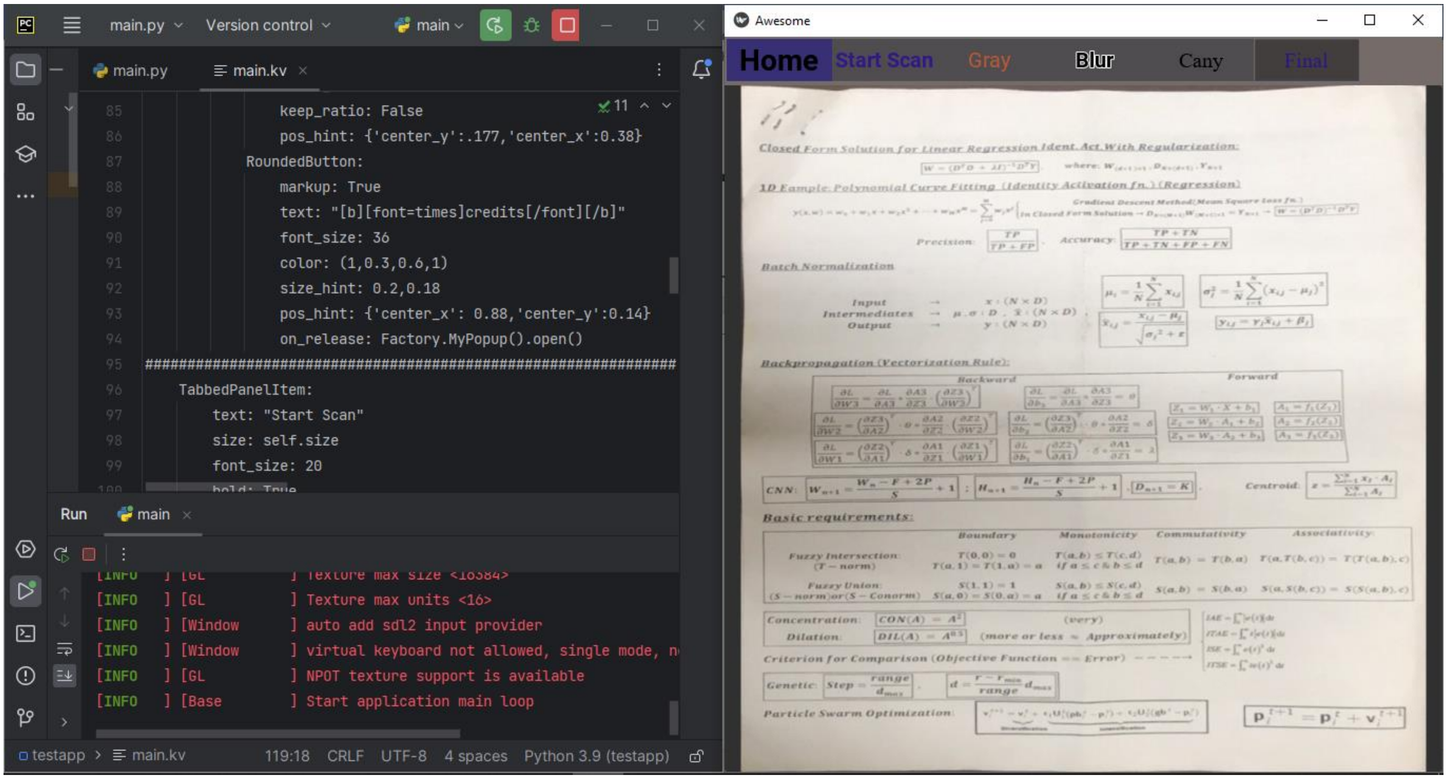
```
[INFO   ] [GL          ] Texture max size <16384>
[INFO   ] [GL          ] Texture max units <16>
[INFO   ] [Window      ] auto add sdl2 input provider
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
```

testapp > main.kv 119:18 CRLF UTF-8 4 spaces Python 3.9 (testapp)



```
            keep_ratio: False
            pos_hint: {'center_y':.177,'center_x':0.38}
        RoundedButton:
            markup: True
            text: "[b][font=times]credits[/font][/b]"
            font_size: 36
            color: (1,0.3,0.6,1)
            size_hint: 0.2,0.18
            pos_hint: {'center_x': 0.88,'center_y':0.14}
            on_release: Factory.MyPopup().open()
###############################################################
    TabbedPanelItem:
        text: "Start Scan"
        size: self.size
        font_size: 20
        bold: True
```

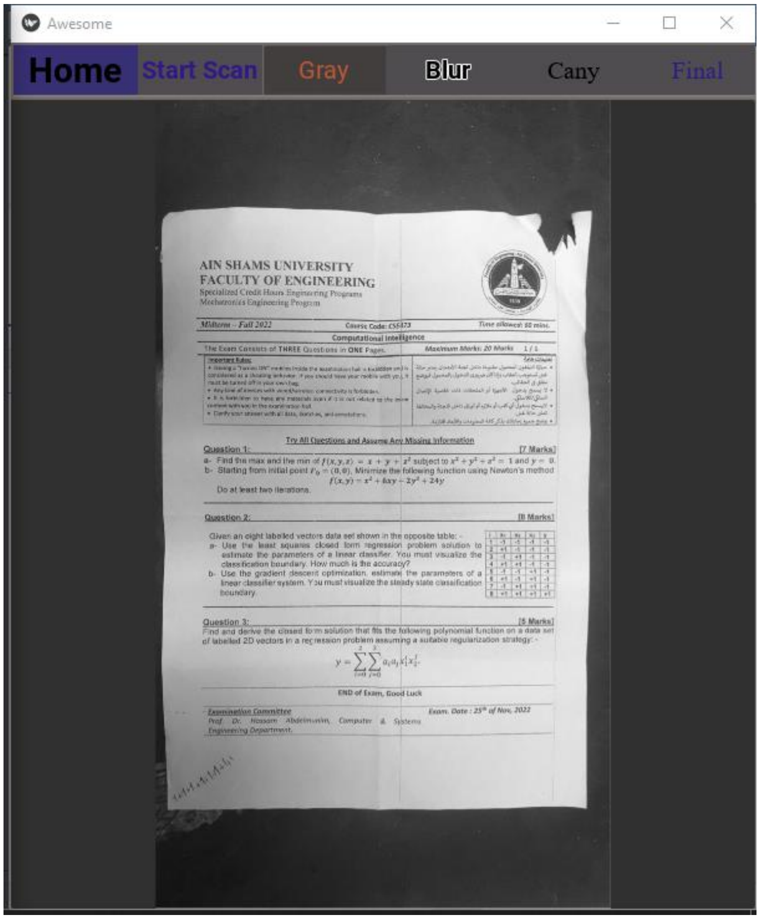
```
[INFO   ] [GL          ] Texture max size <16384>
[INFO   ] [GL          ] Texture max units <16>
[INFO   ] [Window      ] auto add sdl2 input provider
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
```

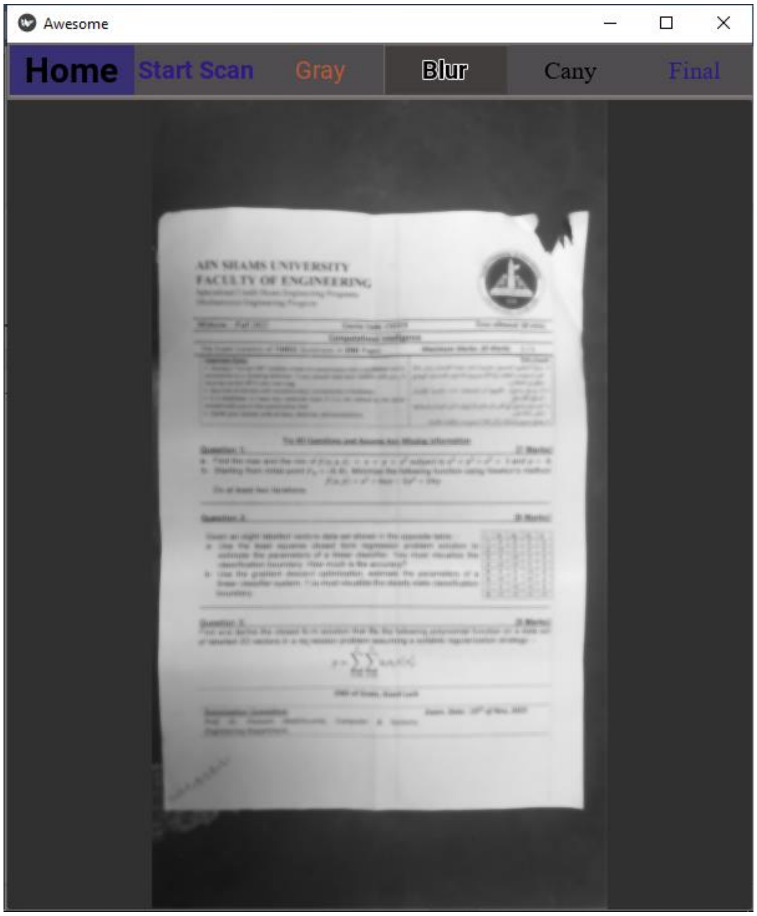


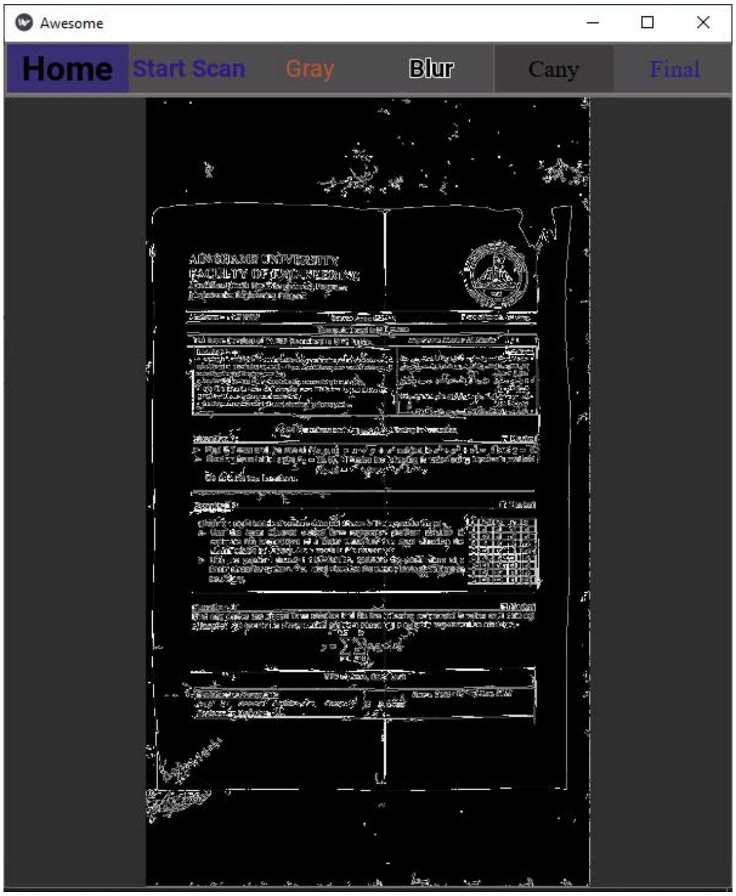
```
            keep_ratio: False
            pos_hint: {'center_y':.177,'center_x':0.38}
        RoundedButton:
            markup: True
            text: "[b][font=times]credits[/font][/b]"
            font_size: 36
            color: (1,0.3,0.6,1)
            size_hint: 0.2,0.18
            pos_hint: {'center_x': 0.88,'center_y':0.14}
            on_release: Factory.MyPopup().open()
###############################################################
    TabbedPanelItem:
        text: "Start Scan"
        size: self.size
        font_size: 20
        bold: True
```

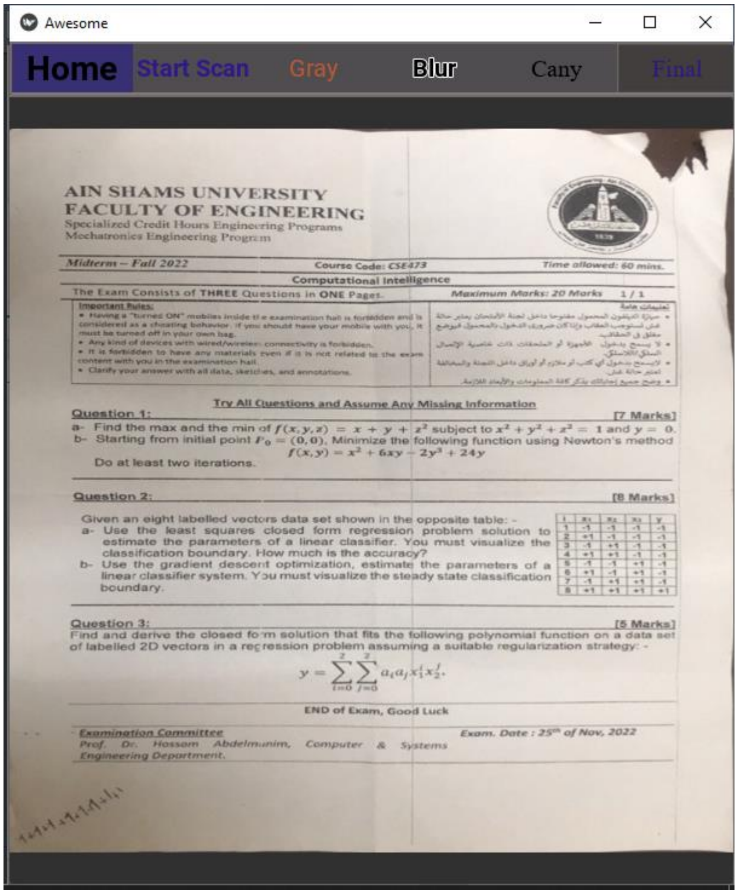
```
[INFO   ] [GL          ] Texture max size <16384>
[INFO   ] [GL          ] Texture max units <16>
[INFO   ] [Window      ] auto add sdl2 input provider
[INFO   ] [Window      ] virtual keyboard not allowed, single mode, n
[INFO   ] [GL          ] NPOT texture support is available
[INFO   ] [Base        ] Start application main loop
```

- **Second Test Sample**









AIN SHAMS UNIVERSITY
FACULTY OF ENGINEERING
Specialized Credit Hours Engineering Programs
Mechatronics Engineering Program

Midterm – Fall 2022 Course Code: CSE473 *Time allowed: 60 mins.*

Computational Intelligence

The Exam Consists of THREE Questions in ONE Pages. *Maximum Marks: 20 Marks* 1 / 1

Important Rules:
- Having a "turned ON" mobiles inside the examination hall is forbidden and is considered as a cheating behavior. If you should have your mobile with you, it must be turned off in your own bag.
- Any kind of devices with wired/wireless connectivity is forbidden.
- It is forbidden to have any materials even if it is not related to the exam content with you in the examination hall.
- Clarify your answer with all data, sketches, and annotations.

تعليمات عامة
- [illegible]
- [illegible]
- [illegible]
- [illegible]

Try All Questions and Assume Any Missing Information

Question 1: [7 Marks]

a- Find the max and the min of $f(x,y,z) = x + y + z^2$ subject to $x^2 + y^2 + z^2 = 1$ and $y = 0$.
b- Starting from initial point $P_0 = (0,0)$, Minimize the following function using Newton's method
$$f(x,y) = x^2 + 6xy - 2y^3 + 24y$$
Do at least two iterations.

Question 2: [8 Marks]

Given an eight labelled vectors data set shown in the opposite table: -
a- Use the least squares closed form regression problem solution to estimate the parameters of a linear classifier. You must visualize the classification boundary. How much is the accuracy?
b- Use the gradient descent optimization, estimate the parameters of a linear classifier system. Yͻu must visualize the steady state classification boundary.

i	x_1	x_2	x_3	y
1	-1	-1	-1	-1
2	+1	-1	-1	-1
3	-1	+1	-1	-1
4	+1	+1	-1	-1
5	-1	-1	+1	-1
6	+1	-1	+1	-1
7	-1	+1	+1	-1
8	+1	+1	+1	+1

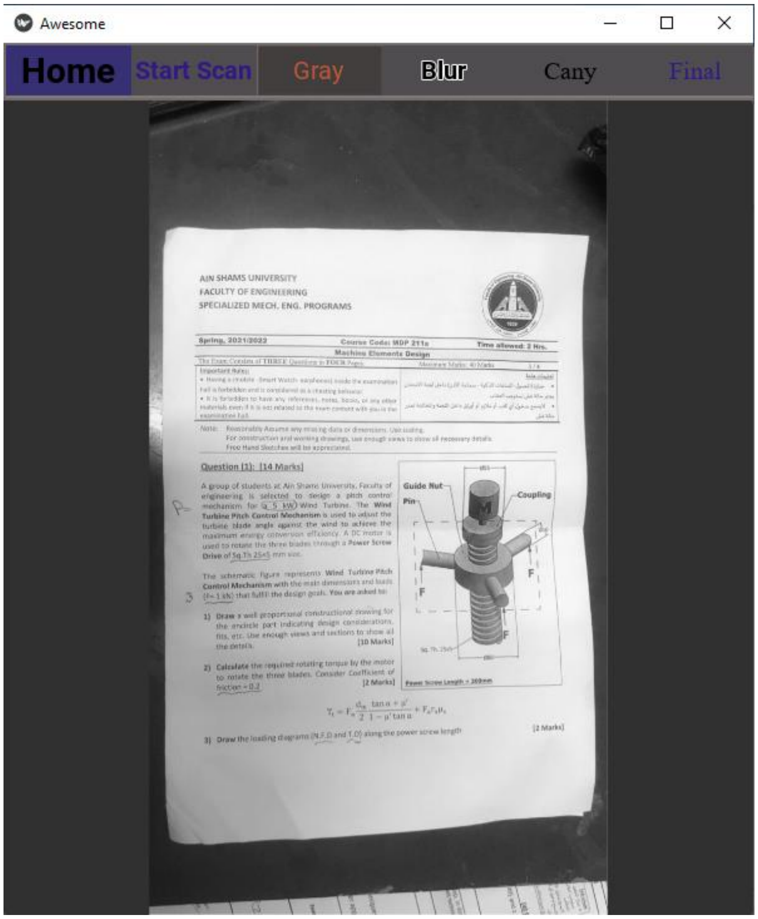
Question 3: [5 Marks]

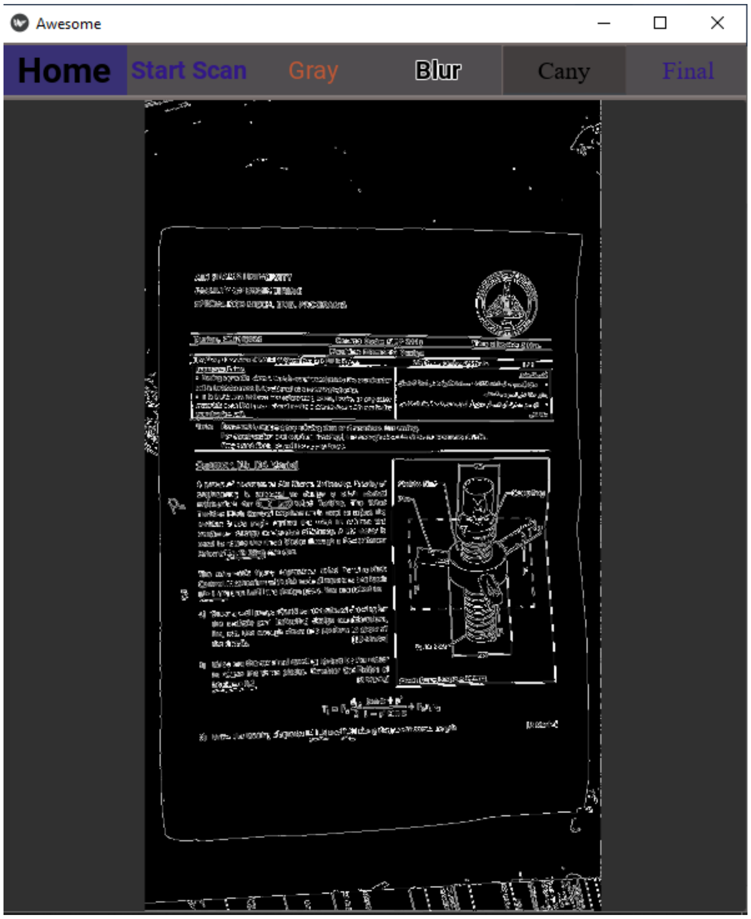
Find and derive the closed fo m solution that fits the following polynomial function on a data set of labelled 2D vectors in a regression problem assuming a suitable regularization strategy: -
$$y = \sum_{i=0}^{2}\sum_{j=0}^{2} a_i a_j x_1^i x_2^j.$$

END of Exam, Good Luck

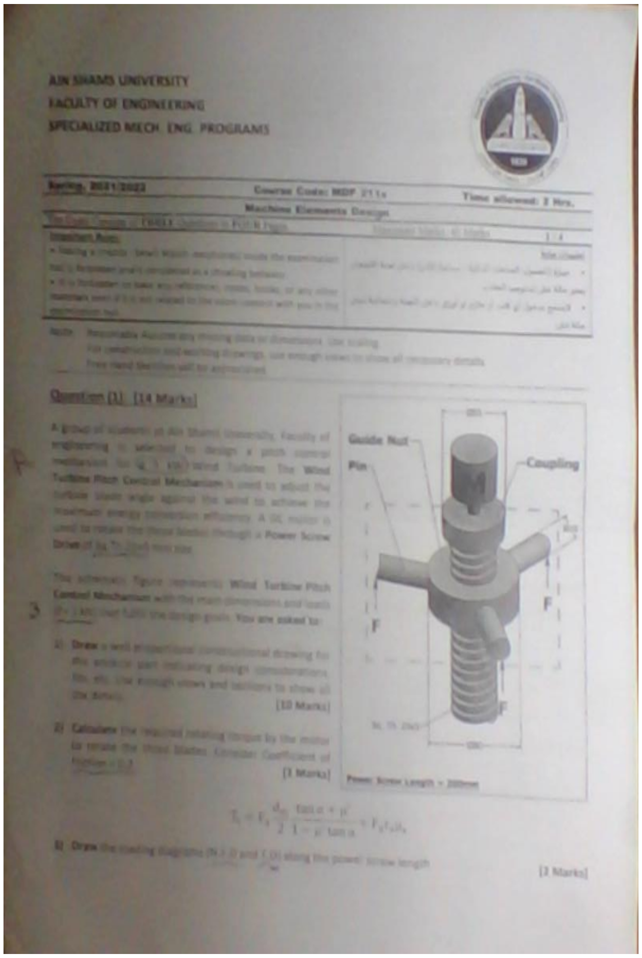
Examination Committee *Exam. Date : 25th of Nov, 2022*
Prof. Dr. Hossam Abdelmunim, Computer & Systems Engineering Department.

- **Third Test sample**

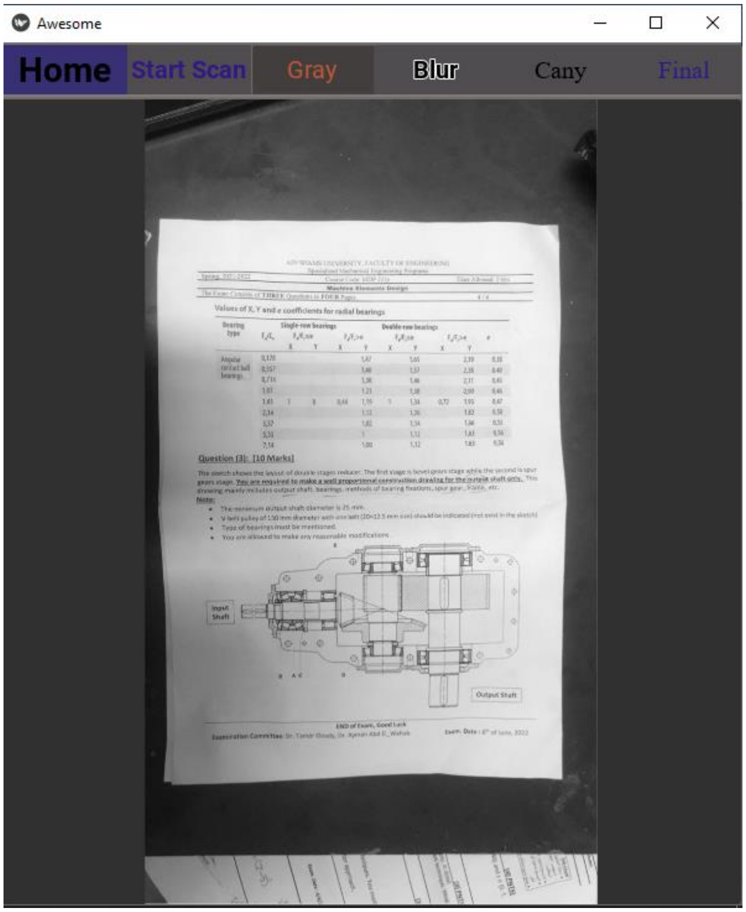


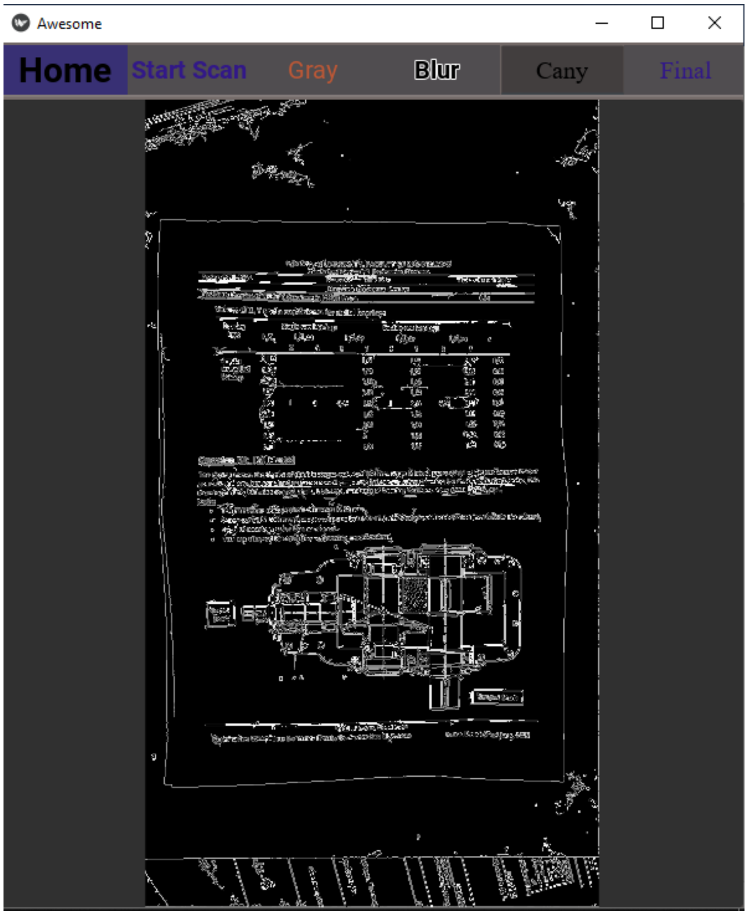


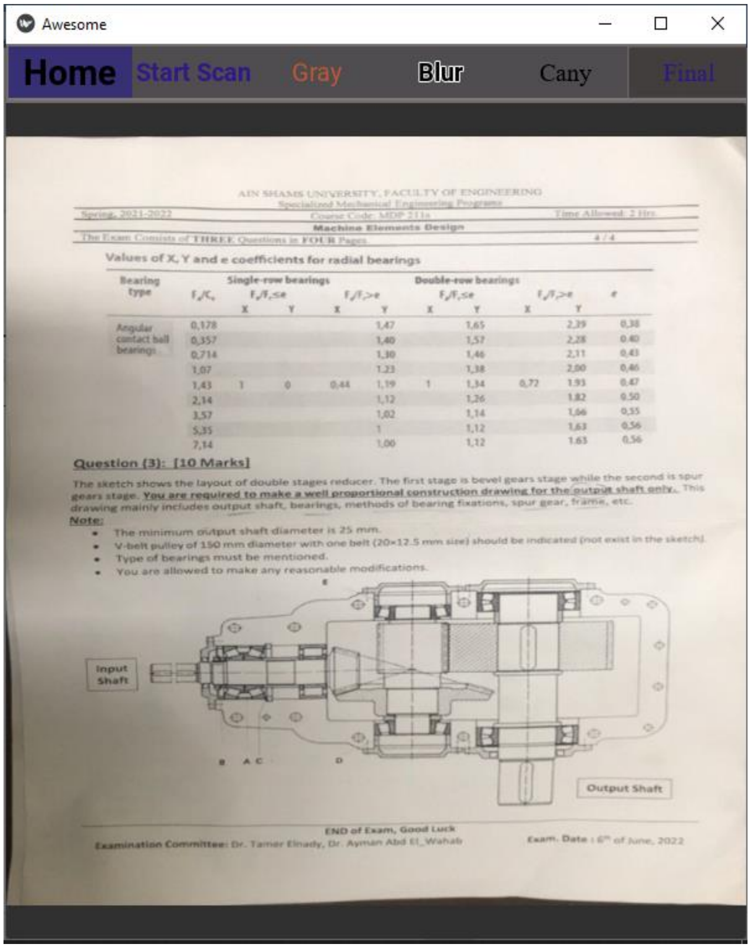
- **Failed To detect The Document Most Probably Because The illumination was too high (had a lamp just above it) and the Edges wasn’t strong Enough Although When the same Document was Tested in The Demo Video Demonstrating the App it was Detected Successfully and added the result in the Test Files so basically it depends on illumination Condition, position and quality of The Used Camera. →**



- **Fourth Test Sample**







AIN SHAMS UNIVERSITY, FACULTY OF ENGINEERING
Specialized Mechanical Engineering Programs

Spring, 2021-2022 | Course Code: MDP 211s | Time Allowed: 2 Hrs

Machine Elements Design

The Exam Consists of THREE Questions in FOUR Pages 4 / 4

Values of X, Y and e coefficients for radial bearings

Bearing type	F_a/C_o	Single-row bearings				Double-row bearings				e
		$F_a/F_r \le e$		$F_a/F_r > e$		$F_a/F_r \le e$		$F_a/F_r > e$		
		X	Y	X	Y	X	Y	X	Y	
Angular contact ball bearings	0,178				1,47		1,65		2,39	0,38
	0,357				1,40		1,57		2,28	0.40
	0,714				1,30		1,46		2,11	0,43
	1,07				1,23		1,38		2,00	0,46
	1,43	1	0	0,44	1,19	1	1,34	0,72	1.93	0.47
	2,14				1,12		1,26		1.82	0.50
	3,57				1,02		1,14		1,66	0,55
	5,35				1		1,12		1,63	0,56
	7,14				1,00		1,12		1.63	0,56

Question (3): [10 Marks]

The sketch shows the layout of double stages reducer. The first stage is bevel gears stage while the second is spur gears stage. **You are required to make a well proportional construction drawing for the output shaft only.** This drawing mainly includes output shaft, bearings, methods of bearing fixations, spur gear, frame, etc.

Note:

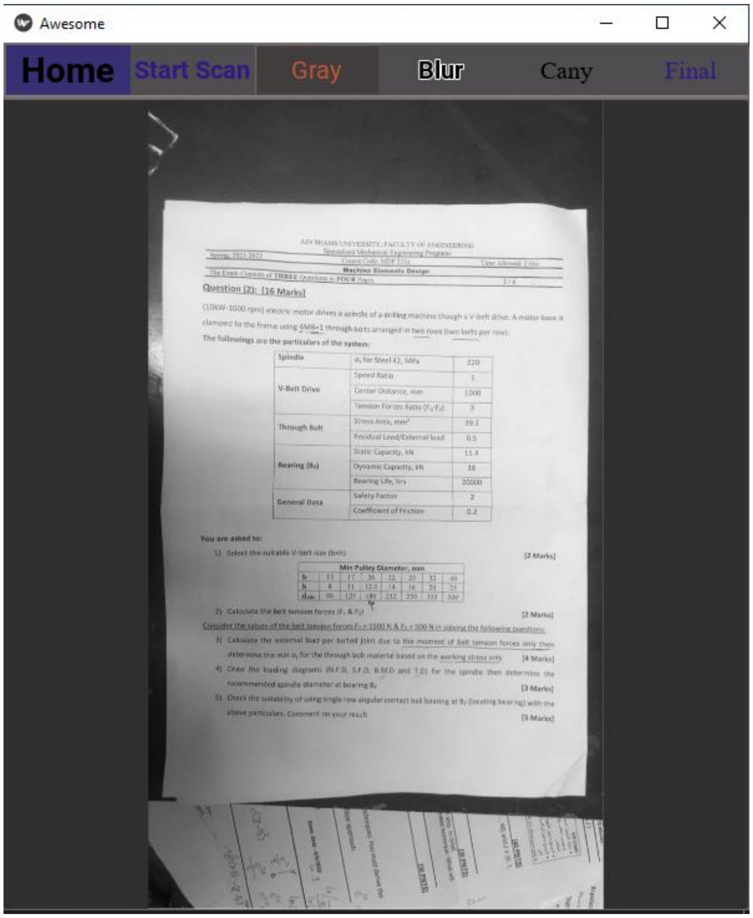
- The minimum output shaft diameter is 25 mm.
- V-belt pulley of 150 mm diameter with one belt (20×12.5 mm size) should be indicated (not exist in the sketch).
- Type of bearings must be mentioned.
- You are allowed to make any reasonable modifications.

END of Exam, Good Luck

Examination Committee: Dr. Tamer Elnady, Dr. Ayman Abd El_Wahab

Exam. Date : 6th of June, 2022

- **Fifth Test Sample**



AIN SHAMS UNIVERSITY, FACULTY OF ENGINEERING
Specialized Mechanical Engineering Program
Spring: 2021-2022
Course Code: MDP 311s
Time allowed: 3 Hrs
Machine Elements Design
The Exam Consists of THREE Questions in FOUR Pages
2/4

Question (2): [16 Marks]

(10kW-1000 rpm) electric motor drives a spindle of a drilling machine though a V-belt drive. A motor base is clamped to the frame using 4M8×1 through bolts arranged in two rows (two bolts per row).

The followings are the particulars of the system:

Spindle	σy for Steel 42, MPa	220
V-Belt Drive	Speed Ratio	1
	Center Distance, mm	1000
	Tension Forces Ratio (F1/F2)	3
Through Bolt	Stress Area, mm²	39.2
	Residual Load/External load	0.5
Bearing (B2)	Static Capacity, kN	11.4
	Dynamic Capacity, kN	18
	Bearing Life, hrs	20000
General Data	Safety Factor	2
	Coefficient of Friction	0.2

You are asked to:

1) Select the suitable V-belt size (b×h) [2 Marks]

Min Pulley Diameter, mm							
b	13	17	20	22	25	32	40
h	8	11	12.5	14	16	20	25
dmin	90	125	180	212	250	355	500

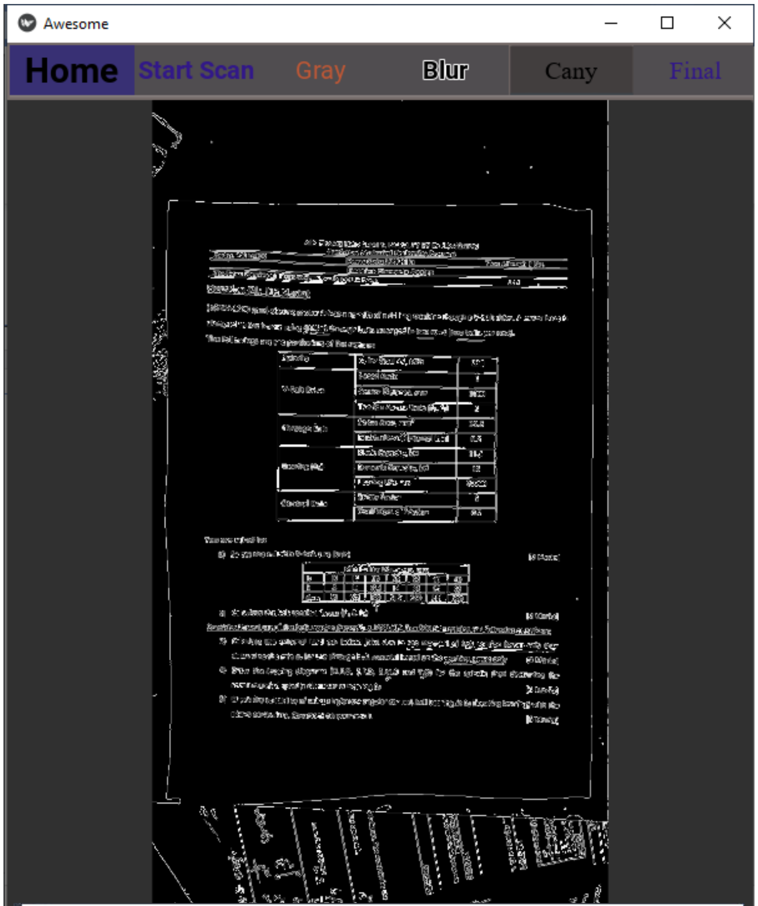
2) Calculate the belt tension forces (F1 & F2) [2 Marks]

Consider the values of the belt tension forces F1 = 1500 N & F2 = 500 N in solving the following questions.

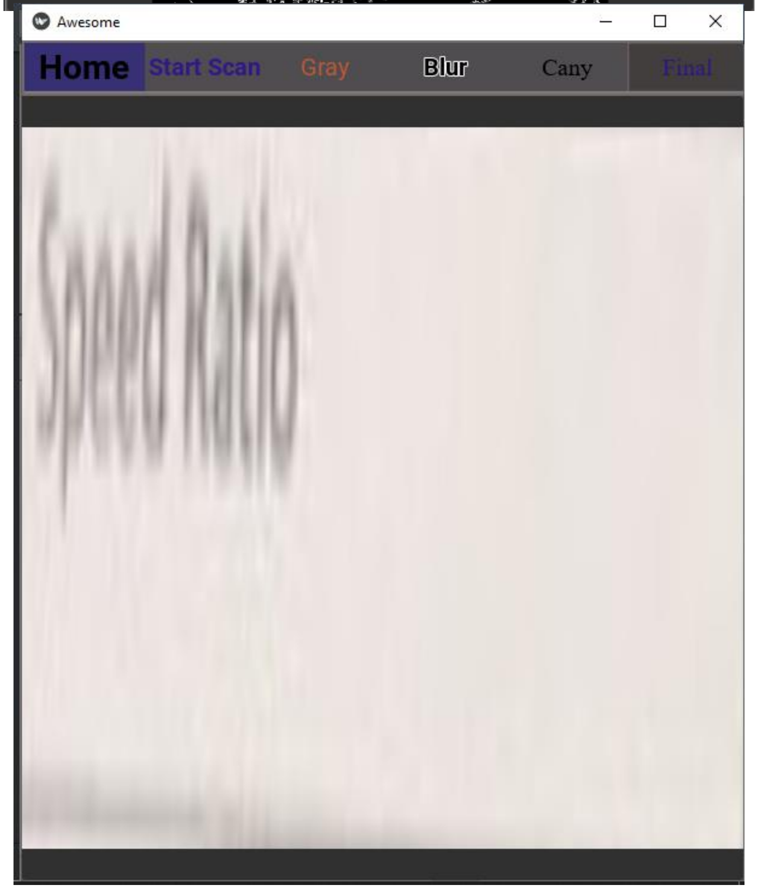
3) Calculate the external load per bolted joint due to the moment of belt tension forces only then determine the min σy for the through bolt material based on the working stress only [4 Marks]

4) Draw the loading diagrams (N.F.D, S.F.D, B.M.D and T.D) for the spindle then determine the recommended spindle diameter at bearing B2 [3 Marks]

5) Check the suitability of using single row angular contact ball bearing at B2 (locating bearing) with the above particulars. Comment on your result [5 Marks]



- **Failed due to high illumination and low-quality camera and positioning and another document interfered in the lower edge**



Comments

Overall, the app scored roughly around 80% (detected 8 out of 10 test samples and converted them into PDF) sometimes it detects all the 10 Documents Correctly but needing correctly positioning the camera above the document,

The app was tested on different illumination Conditions as well as inhomogeneities and it acted quite good and was able to scan right most of the time, most probably due to the camera quality is one of the reasons to fail in Automatic detection and scanning the desired region, but other Times when the Camera positioned Correctly above the Document it scans it perfectly

Appendix

Drive Link For Project Code (main.py / main.kv) and 10 Test Results and a Demo Video:

https://drive.google.com/drive/folders/1p9HagUBbRMtNxUOeTOTaMhlhaMNgfbyF?usp=sharing

Kivy Tutorial Course:

Intro To Kivy – Installing Kivy on Windows – Python Kivy GUI Tutorial #1 – KivyCoder.com

Buildozer For debugging the python and kivy files into one APK

Welcome to Buildozer’s documentation! — Buildozer 0.11 documentation

Used Software in The Project:

- PyCharm
- Kivy
- Android Studio (For visualizing to the logcat for errors)
- Windows Subsystem Linux
- Buildozer

Modules and Libraries Used:

- Python3
- Kivy ==2.1.0
- Numpy
- Opencv