**APPENDIX**

from PyQt5 import QtCore, QtGui, QtWidgets

from software import \*

from PyQt5.QtGui import QPixmap, QMovie, QPainter

class Ui\_Dialog(object):

def messagebox(self, title, message):

mess = QtWidgets.QMessageBox()

mess.setWindowTitle(title)

mess.setText(message)

mess.setStandardButtons(QtWidgets.QMessageBox.Ok)

mess.exec\_()

def main(self):

self.a = True

if self.a:

self.window = QtWidgets.QMainWindow()

self.ui = Ui\_MainWindow()

self.ui.setupUi(self.window)

self.window.show()

else:

print('cant move to next page')

return self.a

def setupUi(self, Dialog):

Dialog.setObjectName("Dialog")

Dialog.resize(542, 292)

self.lblback = QtWidgets.QLabel(Dialog)

self.lblback.setGeometry(QtCore.QRect(0, 0, 541, 291))

self.lblback.setText("")

self.lblback.setObjectName("lblback")

Pixmap = QPixmap('image\soft2.jpg')

self.lblback.setPixmap(Pixmap)

self.lblback.setScaledContents(True)

#self.gif = QMovie("./image/stockgif1.gif")

#self.lblback.setMovie(self.gif)

#self.gif.start()

#self.lblback.setScaledContents(True)

self.lbltitle = QtWidgets.QLabel(Dialog)

self.lbltitle.setGeometry(QtCore.QRect(0, 90, 541, 71))

font = QtGui.QFont()

font.setFamily("Bauhaus 93")

font.setPointSize(20)

self.lbltitle.setFont(font)

self.lbltitle.setAlignment(QtCore.Qt.AlignCenter)

self.lbltitle.setObjectName("lbltitle")

self.lbltitle.setStyleSheet("color: rgb(255, 255, 255);\n")

self.commandLinkButton = QtWidgets.QCommandLinkButton(Dialog)

self.commandLinkButton.setGeometry(QtCore.QRect(420, 240, 111, 41))

font = QtGui.QFont()

font.setFamily("Segoe UI")

font.setBold(True)

font.setWeight(75)

self.commandLinkButton.setFont(font)

self.commandLinkButton.setObjectName("commandLinkButton")

self.commandLinkButton.clicked.connect(self.main)

self.commandLinkButton.setStyleSheet("color: rgb(255, 255, 255);\n")

self.label = QtWidgets.QLabel(Dialog)

self.label.setGeometry(QtCore.QRect(120, 150, 300, 21))

font = QtGui.QFont()

font.setPointSize(10)

font.setBold(False)

font.setWeight(50)

self.label.setFont(font)

self.label.setAlignment(QtCore.Qt.AlignCenter)

self.label.setObjectName("label")

self.label.setStyleSheet("color: rgb(255, 255, 255);\n")

self.retranslateUi(Dialog)

QtCore.QMetaObject.connectSlotsByName(Dialog)

def retranslateUi(self, Dialog):

\_translate = QtCore.QCoreApplication.translate

Dialog.setWindowTitle(\_translate("Dialog", "Home"))

self.lbltitle.setText(\_translate("Dialog", "SOFTWARE COST ESTIMATION"))

self.commandLinkButton.setText(\_translate("Dialog", "Continue"))

self.label.setText(\_translate("Dialog", ""))

if \_\_name\_\_ == "\_\_main\_\_":

import sys

app = QtWidgets.QApplication(sys.argv)

Dialog = QtWidgets.QDialog()

ui = Ui\_Dialog()

ui.setupUi(Dialog)

Dialog.show()

sys.exit(app.exec\_())

# -\*- coding: utf-8 -\*-

# Form implementation generated from reading ui file './pyQts/software.ui'

#

# Created by: PyQt5 UI code generator 5.15.0

#

# WARNING: Any manual changes made to this file will be lost when pyuic5 is

# run again. Do not edit this file unless you know what you are doing.

from PyQt5 import QtCore, QtGui, QtWidgets

import pickle

import soft\_func

import pandas as pd

from PyQt5.QtGui import QPixmap, QMovie, QPainter

class Ui\_MainWindow(object):

def messagebox(self, title, message):

mess = QtWidgets.QMessageBox()

mess.setWindowTitle(title)

mess.setText(message)

mess.setStandardButtons(QtWidgets.QMessageBox.Ok)

mess.exec\_()

def predictor(self):

ei = soft\_func.input\_data(self.ei)

ilf = soft\_func.input\_data2(self.ilf)

eif = soft\_func.input\_data3(self.eif)

eo = soft\_func.input\_data4(self.eo)

eq = soft\_func.input\_data5(self.eq)

teamexp = soft\_func.input\_data6(self.teamexp)

entity = soft\_func.input\_data7(self.entity)

transaction = soft\_func.input\_data8(self.transaction)

length = soft\_func.input\_data9(self.length)

managerexp = soft\_func.input\_data10(self.managerexp)

ILFs = ilf\*10

EIFs = eif\*7

EIs = ei\*4

EOs = eo\*5

EQs = eq\*4

users\_points = EIs + ILFs + EIFs + EOs + EQs

func\_point = users\_points \* 0.65 + (0.01 \* (14 \* 3))

table = pd.DataFrame([[teamexp, managerexp, length, transaction, entity, func\_point]], columns =

['TeamExp', 'ManagerExp','Length', 'Transactions', 'Entities', 'PointsAdjust'])

model = pickle.load(open("model\soft\_model.sav", "rb"))

strmodel = str(model)

if strmodel.startswith('DecisionTreeRegressor'):

self.mod = True

prediction = model.predict(table)

proposed = self.lineEdit.text()

proposed = int(proposed)

for a in prediction:

a = a\*100

if a < proposed:

#self.messagebox("softwareCOST", " The function point is calculated as %s... on mid-level classification\n\n Combining this with the other details provided,\

# \n\n This is our advice: \n\nYou can go ahead with the project, the offer is a good one."%func\_point)

self.messagebox("software cost", " The amount offered for the job meets the specifications. . \n\n Recommendation: Development team can proceed with project ")

elif a > proposed:

#self.messagebox("softwareCOST", "The function point is calculated as %s... on mid-level classification \n\n Combining this with the other details provided,\

# \n\n Our candid advice is that you Consider requesting an \n increase in payment for this project."%func\_point)

self.messagebox("software cost", " This project cannot be completed because the amount offered is too low.. \n\n Recommendation: Project Manager should meet with Clients for further negotiations")

else:

self.mod = False

print('error')

return self.mod

def setupUi(self, MainWindow):

MainWindow.setObjectName("MainWindow")

MainWindow.resize(1365, 759)

self.centralwidget = QtWidgets.QWidget(MainWindow)

self.centralwidget.setObjectName("centralwidget")

self.label\_9 = QtWidgets.QLabel(self.centralwidget)

self.label\_9.setGeometry(QtCore.QRect(-20, -5, 1391, 91))

self.label\_9.setStyleSheet("background-color: rgb(0, 0, 17);\n"

"color: rgb(0, 0, 15);")

self.label\_9.setText("")

self.label\_9.setObjectName("label\_9")

self.label\_13 = QtWidgets.QLabel(self.centralwidget)

self.label\_13.setGeometry(QtCore.QRect(30, 20, 231, 51))

font = QtGui.QFont()

font.setFamily("Cooper Black")

font.setPointSize(20)

font.setBold(False)

font.setItalic(False)

font.setWeight(50)

self.label\_13.setFont(font)

self.label\_13.setStyleSheet("font: 20pt \"Cooper Black\";\n"

"color: rgb(255, 255, 0);\n"

"")

self.label\_13.setObjectName("label\_13")

self.lbl\_back = QtWidgets.QLabel(self.centralwidget)

self.lbl\_back.setGeometry(QtCore.QRect(5, 90, 881, 531))

self.lbl\_back.setObjectName("lbl\_back")

#self.lbl\_back.setStyleSheet("background-color: rgb(255, 255, 255);")

Pixmap = QPixmap('image\potrate.jpeg')

self.lbl\_back.setPixmap(Pixmap)

self.lbl\_back.setScaledContents(True)

self.label\_15 = QtWidgets.QLabel(self.centralwidget)

self.label\_15.setGeometry(QtCore.QRect(-10, 625, 1381, 121))

self.label\_15.setStyleSheet("background-color: rgb(0, 0, 17);")

self.label\_15.setObjectName("label\_15")

self.label = QtWidgets.QLabel(self.centralwidget)

self.label.setGeometry(QtCore.QRect(0, 90, 881, 211))

self.label.setObjectName("label")

self.label\_2 = QtWidgets.QLabel(self.centralwidget)

self.label\_2.setGeometry(QtCore.QRect(880, 90, 481, 531))

self.label\_2.setObjectName("label\_2")

#self.label\_2.setStyleSheet("background-image: url(image\imgs2.jpg);")

Pixmap1 = QPixmap('image\stkimg.png')

self.label\_2.setPixmap(Pixmap1)

self.label\_2.setScaledContents(True)

self.ei = QtWidgets.QLineEdit(self.centralwidget)

self.ei.setGeometry(QtCore.QRect(310, 220, 231, 30))

self.ei.setObjectName("ei")

self.ei.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.ilf = QtWidgets.QLineEdit(self.centralwidget)

self.ilf.setGeometry(QtCore.QRect(80, 190, 211, 30))

self.ilf.setObjectName("ilf")

self.ilf.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.eif = QtWidgets.QLineEdit(self.centralwidget)

self.eif.setGeometry(QtCore.QRect(570, 190, 211, 30))

self.eif.setObjectName("eif")

self.eif.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.eo = QtWidgets.QLineEdit(self.centralwidget)

self.eo.setGeometry(QtCore.QRect(80, 260, 211, 30))

self.eo.setObjectName("eo")

self.eo.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.eq = QtWidgets.QLineEdit(self.centralwidget)

self.eq.setGeometry(QtCore.QRect(570, 260, 211, 30))

self.eq.setObjectName("eq")

self.eq.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.teamexp = QtWidgets.QLineEdit(self.centralwidget)

self.teamexp.setGeometry(QtCore.QRect(80, 460, 211, 30))

self.teamexp.setObjectName("teamexp")

self.teamexp.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.entity = QtWidgets.QLineEdit(self.centralwidget)

self.entity.setGeometry(QtCore.QRect(570, 530, 211, 30))

self.entity.setObjectName("entity")

self.entity.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.transaction = QtWidgets.QLineEdit(self.centralwidget)

self.transaction.setGeometry(QtCore.QRect(80, 530, 211, 30))

self.transaction.setObjectName("transaction")

self.transaction.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.length = QtWidgets.QLineEdit(self.centralwidget)

self.length.setGeometry(QtCore.QRect(570, 460, 211, 30))

self.length.setObjectName("length")

self.length.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.managerexp = QtWidgets.QLineEdit(self.centralwidget)

self.managerexp.setGeometry(QtCore.QRect(310, 490, 231, 30))

self.managerexp.setObjectName("managerexp")

self.managerexp.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.result\_Text = QtWidgets.QLabel(self.centralwidget)

self.result\_Text.setGeometry(QtCore.QRect(920, 190, 371, 361))

self.result\_Text.setObjectName("result\_Text")

font = QtGui.QFont()

font.setFamily("Bauhaus 93")

font.setPointSize(6)

self.result\_Text.setFont(font)

self.result\_Text.setStyleSheet("color: rgb(255, 255, 255);\n")

self.submit = QtWidgets.QPushButton(self.centralwidget)

self.submit.setGeometry(QtCore.QRect(380, 350, 111, 51))

font = QtGui.QFont()

font.setFamily("Bauhaus 93")

font.setPointSize(12)

self.submit.setFont(font)

self.submit.setObjectName("submit")

self.submit.clicked.connect(self.predictor)

self.submit.setStyleSheet("background-color: rgb(137, 207, 240);")

self.lineEdit = QtWidgets.QLineEdit(self.centralwidget)

self.lineEdit.setGeometry(QtCore.QRect(310, 120, 231, 30))

self.lineEdit.setObjectName("lineEdit")

self.lineEdit.setStyleSheet("border :1px solid blue;"

"border-top-left-radius :7px;"

" border-top-right-radius :7px; "

"border-bottom-left-radius : 7px; "

"border-bottom-right-radius : 7px")

self.label\_9.raise\_()

self.label\_13.raise\_()

self.lbl\_back.raise\_()

self.label\_15.raise\_()

self.label.raise\_()

self.label\_2.raise\_()

self.ei.raise\_()

self.ilf.raise\_()

self.eif.raise\_()

self.eo.raise\_()

self.eq.raise\_()

self.teamexp.raise\_()

self.entity.raise\_()

self.transaction.raise\_()

self.length.raise\_()

self.managerexp.raise\_()

self.result\_Text.raise\_()

self.submit.raise\_()

self.lineEdit.raise\_()

MainWindow.setCentralWidget(self.centralwidget)

self.statusbar = QtWidgets.QStatusBar(MainWindow)

self.statusbar.setObjectName("statusbar")

MainWindow.setStatusBar(self.statusbar)

self.actionSignup = QtWidgets.QAction(MainWindow)

self.actionSignup.setObjectName("actionSignup")

self.retranslateUi(MainWindow)

QtCore.QMetaObject.connectSlotsByName(MainWindow)

def retranslateUi(self, MainWindow):

\_translate = QtCore.QCoreApplication.translate

MainWindow.setWindowTitle(\_translate("MainWindow", "MainWindow"))

self.label\_13.setText(\_translate("MainWindow", "Software Cost"))

self.lbl\_back.setText(\_translate("MainWindow", ""))

self.label\_15.setText(\_translate("MainWindow", ""))

self.label.setText(\_translate("MainWindow", ""))

self.ei.setPlaceholderText(\_translate("MainWindow", "Number of User Inputs"))

self.ilf.setPlaceholderText(\_translate("MainWindow", "Number of User Files"))

self.eif.setPlaceholderText(\_translate("MainWindow", "Number of External Interfaces"))

self.eo.setPlaceholderText(\_translate("MainWindow", "Number of User Outputs"))

self.eq.setPlaceholderText(\_translate("MainWindow", "Number of User Enquiries"))

self.teamexp.setPlaceholderText(\_translate("MainWindow", "Team Experience in years"))

self.entity.setPlaceholderText(\_translate("MainWindow", "Total number of Entities"))

self.transaction.setPlaceholderText(\_translate("MainWindow", "Total number of Logical Transactions"))

self.length.setPlaceholderText(\_translate("MainWindow", "Proposed length of production in months"))

self.managerexp.setPlaceholderText(\_translate("MainWindow", "Manager Experience in years"))

self.submit.setText(\_translate("MainWindow", "CHECK"))

self.lineEdit.setPlaceholderText(\_translate("MainWindow", "Proposed amount offered for this software"))

self.actionSignup.setText(\_translate("MainWindow", "Signup"))

self.result\_Text.setText(\_translate("MainWindow", ""))

if \_\_name\_\_ == "\_\_main\_\_":

import sys

app = QtWidgets.QApplication(sys.argv)

MainWindow = QtWidgets.QMainWindow()

ui = Ui\_MainWindow()

ui.setupUi(MainWindow)

MainWindow.show()

sys.exit(app.exec\_())

# -\*- coding: utf-8 -\*-

"""

Created on Thu Jul 8 15:13:02 2021

@author: Touching tap

"""

def input\_data(ei):

age = int(ei.text())

return age

def input\_data2(ilf):

x = int(ilf.text())

return x

def input\_data3(eif):

st = int(eif.text())

return st

def input\_data4(eo):

el = int(eo.text())

return el

def input\_data5(eq):

sec = int(eq.text())

return sec

def input\_data6(teamexp):

rom = int(teamexp.text())

return rom

def input\_data7(entity):

sup = int(entity.text())

return sup

def input\_data8(transaction):

hlt = int(transaction.text())

return hlt

def input\_data9(length):

fam = int(length.text())

return fam

def input\_data10(managerexp):

fam = int(managerexp.text())

return fam

**Justification**

Python: Python programming language is the language chosen to build the system because it has a lot of libraries for data science and machine learning. It has libraries such as Pandas, Numpy, Scikit-learn and alot of other libraries that are built basically for this purpose.

PyQt is a toolkit for graphical user interface (GUI) widgets. It is extracted from the library of Qt. PyQt is the product of the combination of the Python language and the Qt library. PyQt is chosen over tkinter because it comes with many powerful and advanced widgets than tkinter. PyQt has a modern look and a good UI than tkinter and it also has a Designer package which tkinter doesnt have.

Spyder and Jupyter: Jupyter and Spyder are preferred because they already have most of the data science libraries installed in the anaconda environment. So it saves us the stress of having to install libraries which will be necessary in the use of other environments..

**ARCHITECTURE**

MAIN PAGE

Proposed Length of Prod.

Team Experience

Number of Entities

Manager Experience

No. of Logical Transactions

MODEL

Function Point

Number of User Inputs

Number of User Files

Number of External Interfaces

Number of User Outputs

Number of User Enquiries

RESULT

PROPOSED AMOUNT