Actividad 09 (QScene)

RAFAEL ARTURO GUTIERREZ CRUZ

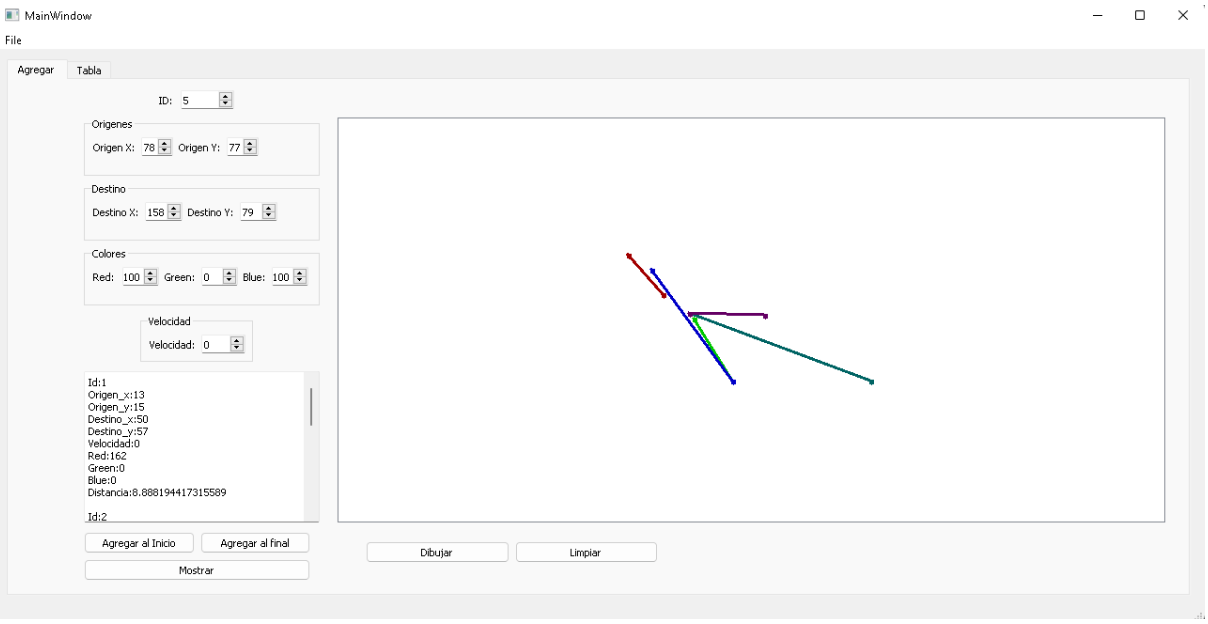
Seminario de Solucion de Problemas de Algoritmia

Lineamientos de evaluación

- [ ] El reporte está en formato Google Docs o PDF.
- [ ] El reporte sigue las pautas del Formato de Actividades .
- [ ] El reporte tiene desarrollada todas las pautas del Formato de Actividades.
- [ ] Se muestra captura de pantalla de lo que se pide en el punto 2.

Desarrollo

Captura de pantalla de al menos 5 partículas desplegadas en el QScene



ID: 5

Origenes
Origen X: 78 Origen Y: 77

Destino
Destino X: 158 Destino Y: 79

Colores
Red: 100 Green: 0 Blue: 100

Velocidad
Velocidad: 0

```
Id:1
Origen_x:13
Origen_y:15
Destino_x:50
Destino_y:57
Velocidad:0
Red:162
Green:0
Blue:0
Distancia:8.888194417315589

Id:2
```

Conclusiones

No tuve muchas complicaciones con la práctica, siendo que en el video de referencia es algo más diferente a los otros, tuve menos complicaciones a la hora de desarrollar esta práctica que en la anterior.

Referencias

MICHEL DAVALOS BOITES. (2020c, noviembre 5). *PySide2 - QScene (Qt for Python)(VI)* [Vídeo]. YouTube. Recuperado 23 de octubre de 2022, de https://www.youtube.com/watch?v=3jHTFzPpZY8

Código

algoritmos.py

```
import math

def distancia_euclidiana(x_1, y_1, x_2, y_2):

    valor1 = x_1 - y_1
    valor1**2

    valor2 = x_2 - y_2
    valor2**2

    return math.sqrt(valor1+valor2)
```

main.py

```
from PySide2.QtWidgets import QApplication
from mainwindow import MainWindow
from scipy.optimize import linprog
import sys

app =QApplication()

window = MainWindow()

window.show()

sys.exit(app.exec_())
```

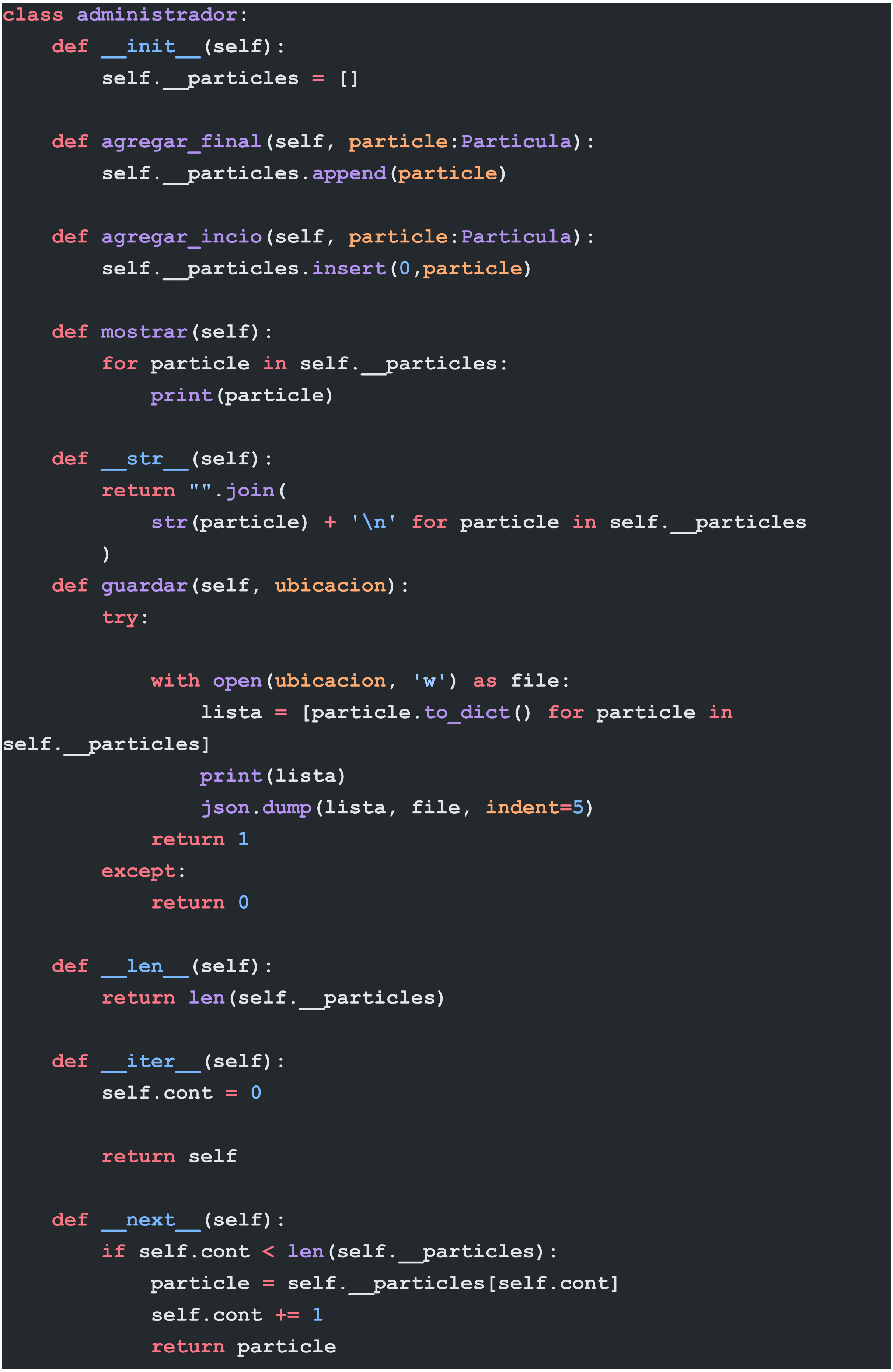
mainwindow.py

```
from base64 import decodebytes
from contextlib import redirect_stderr
from PySide2.QtWidgets import QMainWindow, QFileDialog, QMessageBox,
QTableWidgetItem, QGraphicsScene
from PySide2.QtCore import Slot
from PySide2.QtGui import QPen, QColor, QTransform
from ui_mainwindow import Ui_MainWindow
from particle_adminstrator import administrador
from particulas import Particula

class MainWindow(QMainWindow):
    def __init__(self):
        super(MainWindow, self).__init__()

        self.administrador = administrador()

        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)

self.ui.agregarFinal_pushButton.clicked.connect(self.click_agregar)

self.ui.AgragrInicio_pushButton.clicked.connect(self.click_agregar_inic
io)
        self.ui.Mostrar_pushButton.clicked.connect(self.click_mostrar)


self.ui.actionAbrir.triggered.connect(self.action_abrir_archivo)

self.ui.actionGuardar.triggered.connect(self.action_guardar_archivo)


self.ui.Mostrar_Tabla_pushButton_2.clicked.connect(self.mostrar_tabla)
        self.ui.buscar_pushButton.clicked.connect(self.Buscar)
        self.ui.Limpiar_pushButton_2.clicked.connect(self.Limpiar)
        self.ui.Dibujar_pushButton.clicked.connect(self.Dibujar)

        self.scene = QGraphicsScene()
        self.ui.Particulas_graphicsView.setScene(self.scene)

    @Slot()
    def Dibujar(self):
        # print('dibujando')
        Pen = QPen()
        Pen.setWidth(3)


        for particle in self.administrador:
            r = particle.Red
            g = particle.Green
            b = particle.Blue
            color = QColor(r, g, b)
            Pen.setColor(color)

self.scene.addEllipse(particle.OrigenX,particle.OrigenY,3,3, Pen)

self.scene.addEllipse(particle.DestinoX,particle.DestinoY,3,3, Pen)
            self.scene.addLine(particle.OrigenX, particle.OrigenY,
particle.DestinoX, particle.DestinoY, Pen)




    @Slot()
    def Limpiar(self):
        self.scene.clear()

    @Slot()
    def Buscar(self):
        Codigo_Buscado = self.ui.Buscar_lineEdit.text()


        encontrado = False
        for particle in self.administrador:
            if Codigo_Buscado == str(particle.Codigo):
                self.ui.Tabla.clear()
                self.ui.Tabla.setRowCount(1)

                Codigo_Widget = QTableWidgetItem(str(particle.Codigo))
                OrigenesX_Widget =
QTableWidgetItem(str(particle.OrigenX))
                OrigenesY_Widget =
QTableWidgetItem(str(particle.OrigenY))
                DestinoX_Widget =
QTableWidgetItem(str(particle.DestinoX))
                DestinoY_Widget =
QTableWidgetItem(str(particle.DestinoY))
                Velocidad_Widget =
QTableWidgetItem(str(particle.Velocidad))
                Red_Widget = QTableWidgetItem(str(particle.Red))
                Green_Widget = QTableWidgetItem(str(particle.Green))
                Blue_Widget = QTableWidgetItem(str(particle.Blue))
                Distancia_Widget =
QTableWidgetItem(str(particle.Distancia))

                self.ui.Tabla.setItem(0, 0, Codigo_Widget)
                self.ui.Tabla.setItem(0, 1, OrigenesX_Widget)
                self.ui.Tabla.setItem(0, 2, OrigenesY_Widget)
                self.ui.Tabla.setItem(0, 3, DestinoX_Widget)
                self.ui.Tabla.setItem(0, 4, DestinoY_Widget)
                self.ui.Tabla.setItem(0, 5, Velocidad_Widget)
                self.ui.Tabla.setItem(0, 6, Red_Widget)
                self.ui.Tabla.setItem(0, 7, Green_Widget)
                self.ui.Tabla.setItem(0, 8, Blue_Widget)
                self.ui.Tabla.setItem(0, 9, Distancia_Widget)

                encontrado = True
                return
        print(Codigo_Buscado)
        if not encontrado:
            QMessageBox.warning(self,"Atencion", f'Particula
"{Codigo_Buscado}"no encontrada')

    @Slot()
    def mostrar_tabla(self):
        self.ui.Tabla.setColumnCount(10)
        headers = ["ID" ,"Origen X" ,"Origen Y" ,"Destino X" ,"Destino
Y" ,"Velocidad" ,"Red" ,"Green" ,"Blue" ,"Distancia"]
        self.ui.Tabla.setHorizontalHeaderLabels(headers)

        self.ui.Tabla.setRowCount(len(self.administrador))
        self.ui.Tabla.setColumnWidth(0,50)
        self.ui.Tabla.setColumnWidth(9,200)


        row = 0
        for particle in self.administrador:
            Codigo_Widget = QTableWidgetItem(str(particle.Codigo))
            OrigenesX_Widget = QTableWidgetItem(str(particle.OrigenX))
            OrigenesY_Widget = QTableWidgetItem(str(particle.OrigenY))
            DestinoX_Widget = QTableWidgetItem(str(particle.DestinoX))
            DestinoY_Widget = QTableWidgetItem(str(particle.DestinoY))
            Velocidad_Widget =
QTableWidgetItem(str(particle.Velocidad))
            Red_Widget = QTableWidgetItem(str(particle.Red))
            Green_Widget = QTableWidgetItem(str(particle.Green))
            Blue_Widget = QTableWidgetItem(str(particle.Blue))
            Distancia_Widget =
QTableWidgetItem(str(particle.Distancia))
```

```
            self.ui.Tabla.setItem(row, 0, Codigo_Widget)
            self.ui.Tabla.setItem(row, 1, OrigenesX_Widget)
            self.ui.Tabla.setItem(row, 2, OrigenesY_Widget)
            self.ui.Tabla.setItem(row, 3, DestinoX_Widget)
            self.ui.Tabla.setItem(row, 4, DestinoY_Widget)
            self.ui.Tabla.setItem(row, 5, Velocidad_Widget)
            self.ui.Tabla.setItem(row, 6, Red_Widget)
            self.ui.Tabla.setItem(row, 7, Green_Widget)
            self.ui.Tabla.setItem(row, 8, Blue_Widget)
            self.ui.Tabla.setItem(row, 9, Distancia_Widget)

            row += 1

    @Slot()
    def action_abrir_archivo(self):
        # print('Abriendo')
        ubicacion = QFileDialog.getOpenFileName(self, 'Abrir', '.',
'JSON (*.json)')[0]

        if self.administrador.abrir(ubicacion):
            QMessageBox.information(self, "Exito","Archivo Cargado de:
" + ubicacion)
        else:
            QMessageBox.critical(self, "Error", "No se pudo cargar el
archivo")

    @Slot()
    def action_guardar_archivo(self):
        # print('Guardando')
        ubicacion = QFileDialog.getSaveFileName(self, 'Guardar', '.',
'JSON (*.json)')[0]
        print(ubicacion)
        if self.administrador.guardar(ubicacion):
            QMessageBox.information(self, "Exito","Archivo Guardado en:
" + ubicacion)
        else:
            QMessageBox.critical(self, "Error","No se pudo guardar el
archivo")

    @Slot()
    def click_mostrar(self):
        # self.administrador.mostrar()
        self.ui.salida.clear()
        self.ui.salida.insertPlainText(str(self.administrador))

    @Slot()
    def click_agregar_inicio(self):
        codigo = self.ui.ID_pinBox.value()
        OrigX = self.ui.OrigenX_spinBox.value()
        OrigY = self.ui.OrigenY_spinBox_2.value()
        desX = self.ui.DesX_pinBox.value()
        desY = self.ui.DesY_spinBox_2.value()
        velocidad = self.ui.Velocidad_spinBox_3.value()
        red = self.ui.Red_spinBox_4.value()
        green = self.ui.Green_spinBox_5.value()
        blue = self.ui.Blue_spinBox_6.value()

        Particle = Particula(id=codigo, origen_x=OrigX,
origen_y=OrigY,destino_x=desX, destino_y=desY, velocidad=velocidad,
red=red, green=green, blue=blue)
        self.administrador.agregar_incio(Particle)

    @Slot()
    def click_agregar(self):
        codigo = self.ui.ID_pinBox.value()
        OrigX = self.ui.OrigenX_spinBox.value()
        OrigY = self.ui.OrigenY_spinBox_2.value()
        desX = self.ui.DesX_pinBox.value()
        desY = self.ui.DesY_spinBox_2.value()
        velocidad = self.ui.Velocidad_spinBox_3.value()
        red = self.ui.Red_spinBox_4.value()
        green = self.ui.Green_spinBox_5.value()
        blue = self.ui.Blue_spinBox_6.value()

        Particle = Particula(id=codigo, origen_x=OrigX, origen_y=OrigY,
destino_x=desX, destino_y=desY, velocidad=velocidad, red=red,
green=green, blue=blue)
        self.administrador.agregar_final(Particle)
```

particle_administrador.py

```
from particulas import Particula
import json
```



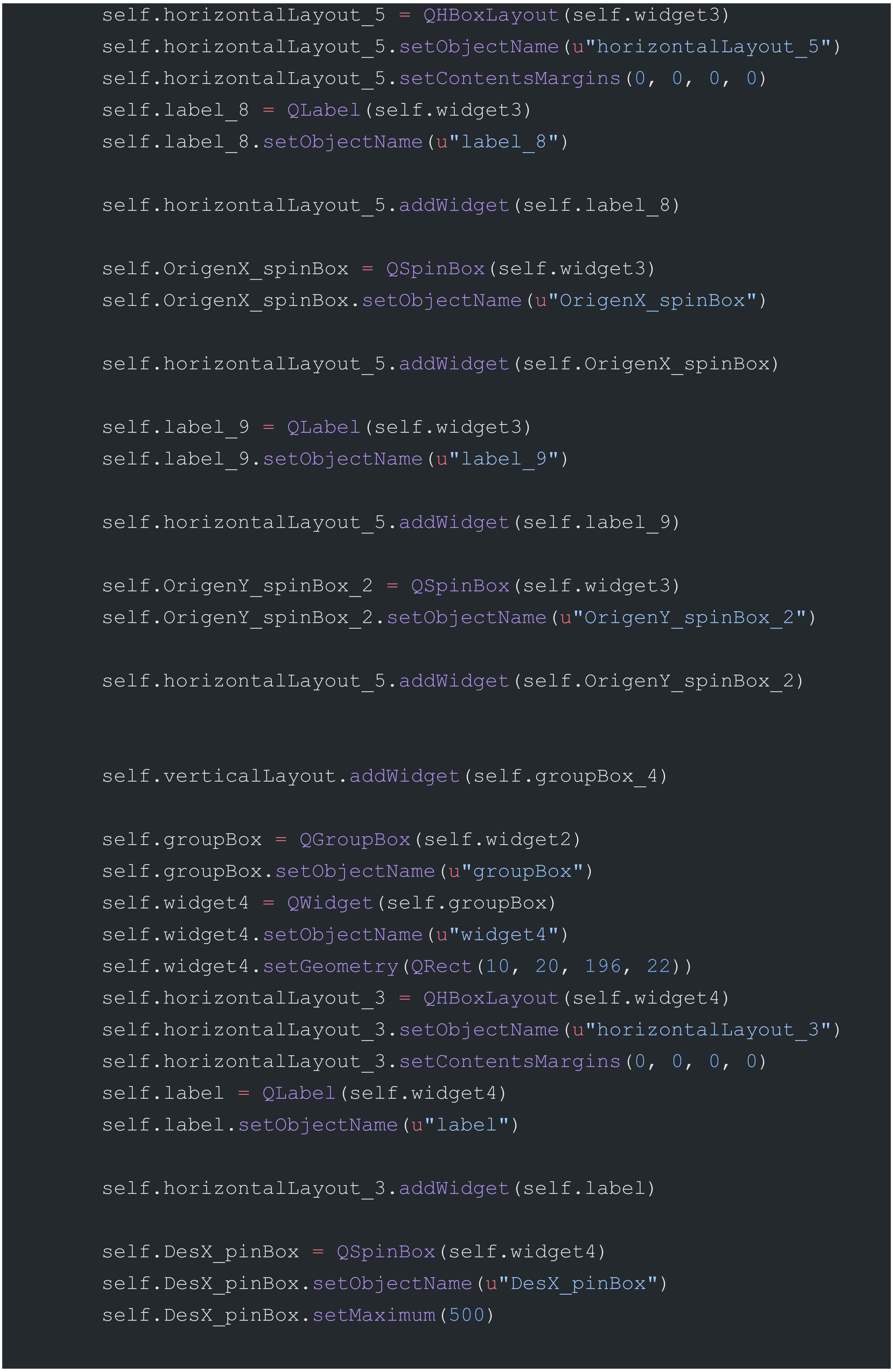
```
class administrador:
    def __init__(self):
        self.__particles = []

    def agregar_final(self, particle:Particula):
        self.__particles.append(particle)

    def agregar_incio(self, particle:Particula):
        self.__particles.insert(0,particle)

    def mostrar(self):
        for particle in self.__particles:
            print(particle)

    def __str__(self):
        return "".join(
            str(particle) + '\n' for particle in self.__particles
        )
    def guardar(self, ubicacion):
        try:

            with open(ubicacion, 'w') as file:
                lista = [particle.to_dict() for particle in
self.__particles]
                print(lista)
                json.dump(lista, file, indent=5)
            return 1
        except:
            return 0

    def __len__(self):
        return len(self.__particles)

    def __iter__(self):
        self.cont = 0

        return self

    def __next__(self):
        if self.cont < len(self.__particles):
            particle = self.__particles[self.cont]
            self.cont += 1
            return particle
        else:
            raise StopIteration

    def abrir(self, ubicacion):
        try:
            with open(ubicacion, 'r') as file:
                lista = json.load(file)
                self.__particles = [Particula(**particle)for particle
in lista]
            return 1
        except:
            return 0
```

particulas.py

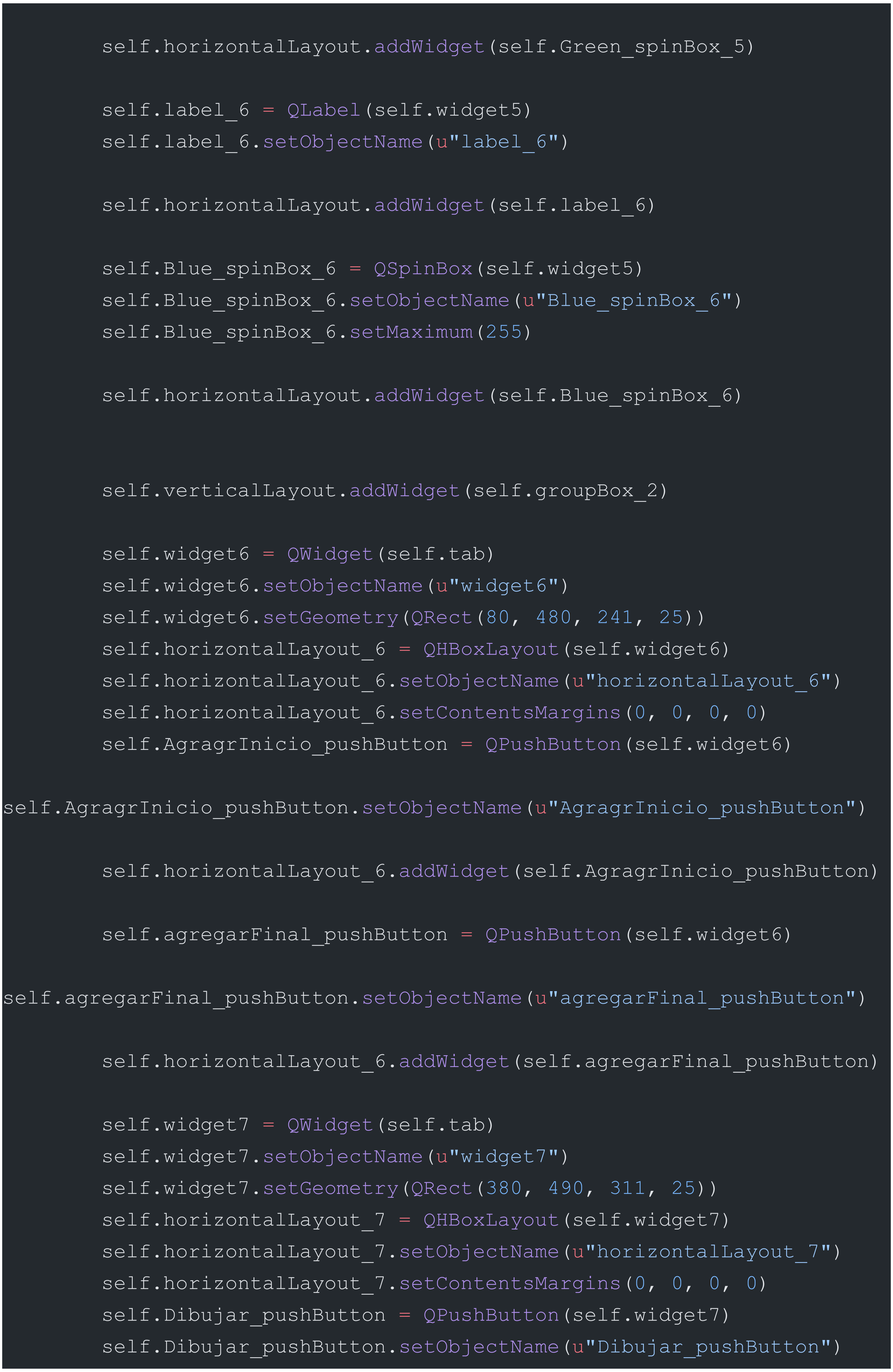
```
import json
from algoritmos import distancia_euclidiana
class Particula:
    def __init__(self, id=0, origen_x=0,
                    origen_y=0, destino_x=0,
                    destino_y=0, velocidad=0,
                    red=0, green=0, blue=0):
        self.__id = id
        self.__origen_x = origen_x
        self.__origen_y = origen_y
        self.__destino_x = destino_x
        self.__destino_y = destino_y
        self.__velocidad = velocidad
        self.__red = red
        self.__green = green
        self.__blue = blue
        self.__distancia = distancia_euclidiana(destino_x, origen_x,
destino_y, origen_y)

    def __str__(self):
        return(
            'Id:' + str(self.__id) + '\n'
            'Origen_x:' + str(self.__origen_x) + '\n' +
            'Origen_y:' + str(self.__origen_y) + '\n' +
            'Destino_x:' + str(self.__destino_x) + '\n' +
            'Destino_y:' + str(self.__destino_y) + '\n' +
            'Velocidad:' + str(self.__velocidad) + '\n' +
            'Red:' + str(self.__red) + '\n' +
            'Green:' + str(self.__green) + '\n' +
            'Blue:' + str(self.__blue) + '\n' +
            'Distancia:' + str(self.__distancia) + '\n'
        )
    @property
    def Codigo(self):
        return self.__id
    @property
    def OrigenX(self):
        return self.__origen_x
    @property
    def OrigenY(self):
        return self.__origen_y
    @property
    def DestinoX(self):
        return self.__destino_x
    @property
    def DestinoY(self):
        return self.__destino_y
    @property
    def Velocidad(self):
        return self.__velocidad
    @property
    def Red(self):
        return self.__red
    @property
    def Green(self):
        return self.__green
    @property
    def Blue(self):
        return self.__blue
    @property
    def Distancia(self):
        return self.__distancia

    def to_dict(self):
        return {
            "id":self.__id,
            "origen_x":self.__origen_x,
            "origen_y":self.__origen_y,
            "destino_x":self.__destino_x,
            "destino_y":self.__destino_y,
            "velocidad":self.__velocidad,
            "red":self.__red,
            "green":self.__green,
            "blue":self.__blue
        }

```

ui_mainwindow.py

```
from PySide2.QtCore import *
from PySide2.QtGui import *
from PySide2.QtWidgets import *


class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        if not MainWindow.objectName():
            MainWindow.setObjectName(u"MainWindow")
        MainWindow.resize(1285, 630)
        self.actionAbrir = QAction(MainWindow)
        self.actionAbrir.setObjectName(u"actionAbrir")
        self.actionGuardar = QAction(MainWindow)
        self.actionGuardar.setObjectName(u"actionGuardar")
        self.centralwidget = QWidget(MainWindow)
        self.centralwidget.setObjectName(u"centralwidget")
        self.tabWidget = QTabWidget(self.centralwidget)
        self.tabWidget.setObjectName(u"tabWidget")
        self.tabWidget.setGeometry(QRect(10, 10, 1261, 571))
        self.tab = QWidget()
        self.tab.setObjectName(u"tab")
        self.salida = QPlainTextEdit(self.tab)
        self.salida.setObjectName(u"salida")
        self.salida.setGeometry(QRect(80, 310, 251, 161))
        self.Mostrar_pushButton = QPushButton(self.tab)
        self.Mostrar_pushButton.setObjectName(u"Mostrar_pushButton")
        self.Mostrar_pushButton.setGeometry(QRect(80, 510, 241, 23))
        self.groupBox_3 = QGroupBox(self.tab)
        self.groupBox_3.setObjectName(u"groupBox_3")
        self.groupBox_3.setGeometry(QRect(140, 250, 120, 51))
        self.widget = QWidget(self.groupBox_3)
        self.widget.setObjectName(u"widget")
        self.widget.setGeometry(QRect(10, 20, 102, 22))
        self.horizontalLayout_4 = QHBoxLayout(self.widget)
        self.horizontalLayout_4.setObjectName(u"horizontalLayout_4")
        self.horizontalLayout_4.setContentsMargins(0, 0, 0, 0)
        self.label_3 = QLabel(self.widget)
        self.label_3.setObjectName(u"label_3")

        self.horizontalLayout_4.addWidget(self.label_3)

        self.Velocidad_spinBox_3 = QSpinBox(self.widget)
        self.Velocidad_spinBox_3.setObjectName(u"Velocidad_spinBox_3")
        self.Velocidad_spinBox_3.setMaximum(1000)

        self.horizontalLayout_4.addWidget(self.Velocidad_spinBox_3)

        self.Particulas_graphicsView = QGraphicsView(self.tab)

self.Particulas_graphicsView.setObjectName(u"Particulas_graphicsView")
        self.Particulas_graphicsView.setGeometry(QRect(350, 40, 881,
431))
        self.widget1 = QWidget(self.tab)
        self.widget1.setObjectName(u"widget1")
        self.widget1.setGeometry(QRect(160, 10, 80, 22))
        self.horizontalLayout_2 = QHBoxLayout(self.widget1)
        self.horizontalLayout_2.setObjectName(u"horizontalLayout_2")
        self.horizontalLayout_2.setContentsMargins(0, 0, 0, 0)
        self.label_7 = QLabel(self.widget1)
        self.label_7.setObjectName(u"label_7")

        self.horizontalLayout_2.addWidget(self.label_7)

        self.ID_pinBox = QSpinBox(self.widget1)
        self.ID_pinBox.setObjectName(u"ID_pinBox")
        self.ID_pinBox.setMaximum(500000)

        self.horizontalLayout_2.addWidget(self.ID_pinBox)

        self.widget2 = QWidget(self.tab)
        self.widget2.setObjectName(u"widget2")
        self.widget2.setGeometry(QRect(80, 40, 251, 201))
        self.verticalLayout = QVBoxLayout(self.widget2)
        self.verticalLayout.setObjectName(u"verticalLayout")
        self.verticalLayout.setContentsMargins(0, 0, 0, 0)
        self.groupBox_4 = QGroupBox(self.widget2)
        self.groupBox_4.setObjectName(u"groupBox_4")
        self.widget3 = QWidget(self.groupBox_4)
        self.widget3.setObjectName(u"widget3")
        self.widget3.setGeometry(QRect(10, 20, 176, 22))
```



```
        self.horizontalLayout_5 = QHBoxLayout(self.widget3)
        self.horizontalLayout_5.setObjectName(u"horizontalLayout_5")
        self.horizontalLayout_5.setContentsMargins(0, 0, 0, 0)
        self.label_8 = QLabel(self.widget3)
        self.label_8.setObjectName(u"label_8")

        self.horizontalLayout_5.addWidget(self.label_8)

        self.OrigenX_spinBox = QSpinBox(self.widget3)
        self.OrigenX_spinBox.setObjectName(u"OrigenX_spinBox")

        self.horizontalLayout_5.addWidget(self.OrigenX_spinBox)

        self.label_9 = QLabel(self.widget3)
        self.label_9.setObjectName(u"label_9")

        self.horizontalLayout_5.addWidget(self.label_9)

        self.OrigenY_spinBox_2 = QSpinBox(self.widget3)
        self.OrigenY_spinBox_2.setObjectName(u"OrigenY_spinBox_2")

        self.horizontalLayout_5.addWidget(self.OrigenY_spinBox_2)


        self.verticalLayout.addWidget(self.groupBox_4)

        self.groupBox = QGroupBox(self.widget2)
        self.groupBox.setObjectName(u"groupBox")
        self.widget4 = QWidget(self.groupBox)
        self.widget4.setObjectName(u"widget4")
        self.widget4.setGeometry(QRect(10, 20, 196, 22))
        self.horizontalLayout_3 = QHBoxLayout(self.widget4)
        self.horizontalLayout_3.setObjectName(u"horizontalLayout_3")
        self.horizontalLayout_3.setContentsMargins(0, 0, 0, 0)
        self.label = QLabel(self.widget4)
        self.label.setObjectName(u"label")

        self.horizontalLayout_3.addWidget(self.label)

        self.DesX_pinBox = QSpinBox(self.widget4)
        self.DesX_pinBox.setObjectName(u"DesX_pinBox")
        self.DesX_pinBox.setMaximum(500)
        self.horizontalLayout_3.addWidget(self.DesX_pinBox)

        self.label_2 = QLabel(self.widget4)
        self.label_2.setObjectName(u"label_2")

        self.horizontalLayout_3.addWidget(self.label_2)

        self.DesY_spinBox_2 = QSpinBox(self.widget4)
        self.DesY_spinBox_2.setObjectName(u"DesY_spinBox_2")
        self.DesY_spinBox_2.setMaximum(500)

        self.horizontalLayout_3.addWidget(self.DesY_spinBox_2)


        self.verticalLayout.addWidget(self.groupBox)

        self.groupBox_2 = QGroupBox(self.widget2)
        self.groupBox_2.setObjectName(u"groupBox_2")
        self.widget5 = QWidget(self.groupBox_2)
        self.widget5.setObjectName(u"widget5")
        self.widget5.setGeometry(QRect(10, 20, 229, 22))
        self.horizontalLayout = QHBoxLayout(self.widget5)
        self.horizontalLayout.setObjectName(u"horizontalLayout")
        self.horizontalLayout.setContentsMargins(0, 0, 0, 0)
        self.label_4 = QLabel(self.widget5)
        self.label_4.setObjectName(u"label_4")

        self.horizontalLayout.addWidget(self.label_4)

        self.Red_spinBox_4 = QSpinBox(self.widget5)
        self.Red_spinBox_4.setObjectName(u"Red_spinBox_4")
        self.Red_spinBox_4.setMaximum(255)

        self.horizontalLayout.addWidget(self.Red_spinBox_4)

        self.label_5 = QLabel(self.widget5)
        self.label_5.setObjectName(u"label_5")

        self.horizontalLayout.addWidget(self.label_5)

        self.Green_spinBox_5 = QSpinBox(self.widget5)
        self.Green_spinBox_5.setObjectName(u"Green_spinBox_5")
        self.Green_spinBox_5.setMaximum(255)
```



```
        self.horizontalLayout.addWidget(self.Green_spinBox_5)

        self.label_6 = QLabel(self.widget5)
        self.label_6.setObjectName(u"label_6")

        self.horizontalLayout.addWidget(self.label_6)

        self.Blue_spinBox_6 = QSpinBox(self.widget5)
        self.Blue_spinBox_6.setObjectName(u"Blue_spinBox_6")
        self.Blue_spinBox_6.setMaximum(255)

        self.horizontalLayout.addWidget(self.Blue_spinBox_6)


        self.verticalLayout.addWidget(self.groupBox_2)

        self.widget6 = QWidget(self.tab)
        self.widget6.setObjectName(u"widget6")
        self.widget6.setGeometry(QRect(80, 480, 241, 25))
        self.horizontalLayout_6 = QHBoxLayout(self.widget6)
        self.horizontalLayout_6.setObjectName(u"horizontalLayout_6")
        self.horizontalLayout_6.setContentsMargins(0, 0, 0, 0)
        self.AgragrInicio_pushButton = QPushButton(self.widget6)

self.AgragrInicio_pushButton.setObjectName(u"AgragrInicio_pushButton")

        self.horizontalLayout_6.addWidget(self.AgragrInicio_pushButton)

        self.agregarFinal_pushButton = QPushButton(self.widget6)

self.agregarFinal_pushButton.setObjectName(u"agregarFinal_pushButton")

        self.horizontalLayout_6.addWidget(self.agregarFinal_pushButton)

        self.widget7 = QWidget(self.tab)
        self.widget7.setObjectName(u"widget7")
        self.widget7.setGeometry(QRect(380, 490, 311, 25))
        self.horizontalLayout_7 = QHBoxLayout(self.widget7)
        self.horizontalLayout_7.setObjectName(u"horizontalLayout_7")
        self.horizontalLayout_7.setContentsMargins(0, 0, 0, 0)
        self.Dibujar_pushButton = QPushButton(self.widget7)
        self.Dibujar_pushButton.setObjectName(u"Dibujar_pushButton")
        self.horizontalLayout_7.addWidget(self.Dibujar_pushButton)

        self.Limpiar_pushButton_2 = QPushButton(self.widget7)

        self.Limpiar_pushButton_2.setObjectName(u"Limpiar_pushButton_2")

        self.horizontalLayout_7.addWidget(self.Limpiar_pushButton_2)

        self.tabWidget.addTab(self.tab, "")
        self.tab_2 = QWidget()
        self.tab_2.setObjectName(u"tab_2")
        self.gridLayout = QGridLayout(self.tab_2)
        self.gridLayout.setObjectName(u"gridLayout")
        self.Tabla = QTableWidget(self.tab_2)
        self.Tabla.setObjectName(u"Tabla")

        self.gridLayout.addWidget(self.Tabla, 0, 0, 1, 3)

        self.Buscar_lineEdit = QLineEdit(self.tab_2)
        self.Buscar_lineEdit.setObjectName(u"Buscar_lineEdit")

        self.gridLayout.addWidget(self.Buscar_lineEdit, 1, 0, 1, 1)

        self.buscar_pushButton = QPushButton(self.tab_2)
        self.buscar_pushButton.setObjectName(u"buscar_pushButton")

        self.gridLayout.addWidget(self.buscar_pushButton, 1, 1, 1, 1)

        self.Mostrar_Tabla_pushButton_2 = QPushButton(self.tab_2)

        self.Mostrar_Tabla_pushButton_2.setObjectName(u"Mostrar_Tabla_pushButton_2")

        self.gridLayout.addWidget(self.Mostrar_Tabla_pushButton_2, 1, 2, 1, 1)

        self.tabWidget.addTab(self.tab_2, "")
        MainWindow.setCentralWidget(self.centralwidget)
        self.menubar = QMenuBar(MainWindow)
        self.menubar.setObjectName(u"menubar")
        self.menubar.setGeometry(QRect(0, 0, 1285, 21))
        self.menuFile = QMenu(self.menubar)
        self.menuFile.setObjectName(u"menuFile")
        MainWindow.setMenuBar(self.menubar)
        self.statusbar = QStatusBar(MainWindow)
        self.statusbar.setObjectName(u"statusbar")
        MainWindow.setStatusBar(self.statusbar)

        self.menubar.addAction(self.menuFile.menuAction())
        self.menuFile.addAction(self.actionAbrir)
        self.menuFile.addAction(self.actionGuardar)

        self.retranslateUi(MainWindow)

        self.tabWidget.setCurrentIndex(0)


        QMetaObject.connectSlotsByName(MainWindow)
    # setupUi

    def retranslateUi(self, MainWindow):

MainWindow.setWindowTitle(QCoreApplication.translate("MainWindow",
u"MainWindow", None))

self.actionAbrir.setText(QCoreApplication.translate("MainWindow",
u"Abrir", None))
#if QT_CONFIG(shortcut)

self.actionAbrir.setShortcut(QCoreApplication.translate("MainWindow",
u"Ctrl+O", None))
#endif // QT_CONFIG(shortcut)

self.actionGuardar.setText(QCoreApplication.translate("MainWindow",
u"Guardar", None))
#if QT_CONFIG(shortcut)

self.actionGuardar.setShortcut(QCoreApplication.translate("MainWindow",
u"Ctrl+S", None))
#endif // QT_CONFIG(shortcut)

self.Mostrar_pushButton.setText(QCoreApplication.translate("MainWindow"
, u"Mostrar", None))
```

```
self.groupBox_3.setTitle(QCoreApplication.translate("MainWindow",
u"Velocidad", None))
        self.label_3.setText(QCoreApplication.translate("MainWindow",
u"Velocidad:", None))
        self.label_7.setText(QCoreApplication.translate("MainWindow",
u"ID:", None))

self.groupBox_4.setTitle(QCoreApplication.translate("MainWindow",
u"Origenes", None))
        self.label_8.setText(QCoreApplication.translate("MainWindow",
u"Origen X:", None))
        self.label_9.setText(QCoreApplication.translate("MainWindow",
u"Origen Y:", None))
        self.groupBox.setTitle(QCoreApplication.translate("MainWindow",
u"Destino", None))
        self.label.setText(QCoreApplication.translate("MainWindow",
u"Destino X:", None))
        self.label_2.setText(QCoreApplication.translate("MainWindow",
u"Destino Y:", None))

self.groupBox_2.setTitle(QCoreApplication.translate("MainWindow",
u"Colores", None))
        self.label_4.setText(QCoreApplication.translate("MainWindow",
u"Red:", None))
        self.label_5.setText(QCoreApplication.translate("MainWindow",
u"Green:", None))
        self.label_6.setText(QCoreApplication.translate("MainWindow",
u"Blue:", None))

self.AgragrInicio_pushButton.setText(QCoreApplication.translate("MainWi
ndow", u"Agregar al Inicio", None))

self.agregarFinal_pushButton.setText(QCoreApplication.translate("MainWi
ndow", u"Agregar al final", None))

self.Dibujar_pushButton.setText(QCoreApplication.translate("MainWindow"
, u"Dibujar", None))

self.Limpiar_pushButton_2.setText(QCoreApplication.translate("MainWindo
w", u"Limpiar", None))
        self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab),
QCoreApplication.translate("MainWindow", u"Agregar", None))
```

```
self.Buscar_lineEdit.setPlaceholderText(QCoreApplication.translate("MainWindow", u"ID", None))

self.buscar_pushButton.setText(QCoreApplication.translate("MainWindow", u"Buscar", None))

self.Mostrar_Tabla_pushButton_2.setText(QCoreApplication.translate("MainWindow", u"Mostra", None))
        self.tabWidget.setTabText(self.tabWidget.indexOf(self.tab_2), QCoreApplication.translate("MainWindow", u"Tabla", None))
        self.menuFile.setTitle(QCoreApplication.translate("MainWindow", u"File", None))
    # retranslateUi
```