Actividad 07 // (QFileDialog)

Sámano Juárez Juan Jesus.

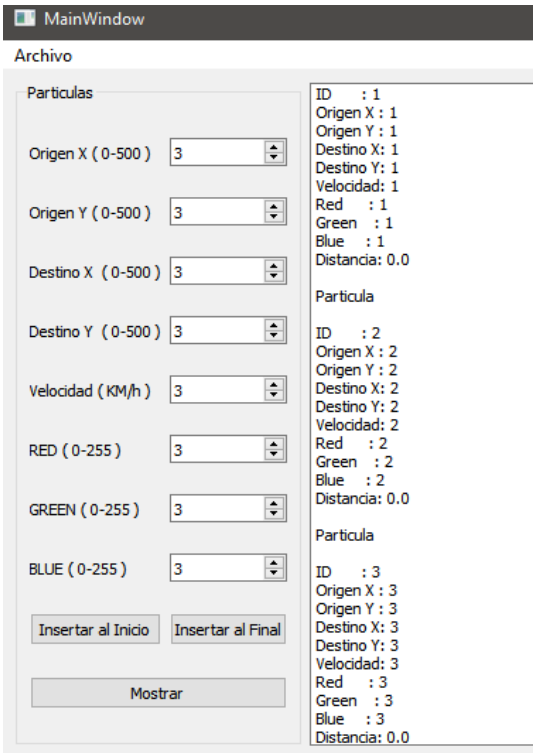
Seminario de Solución de Problemas de Algoritmia.

Lineamiento 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 la captura de pantalla de las partículas con el método mostrar() previo a generar el respaldo.
- [ ] Se muestran capturas de pantallas de los pasos que se realizan en la interfaz para generar el respaldo.
- [ ] Se muestra el contenido del archivo *.json.*
- [ ] Se muestran capturas de pantallas de los pasos que se realizan en la interfaz para abrir el archivo de respaldo *.json*.
- [ ] Se muestra la captura de pantalla de las partículas con el método mostrar() después de abrir el respaldo.

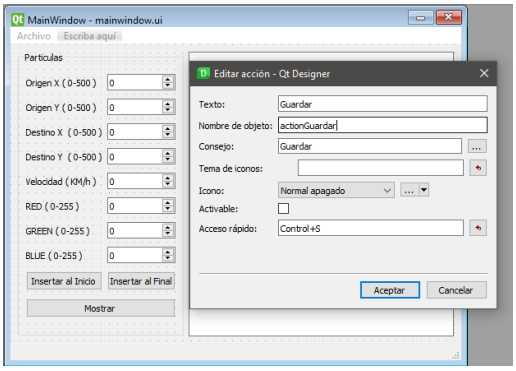
Desarrollo.

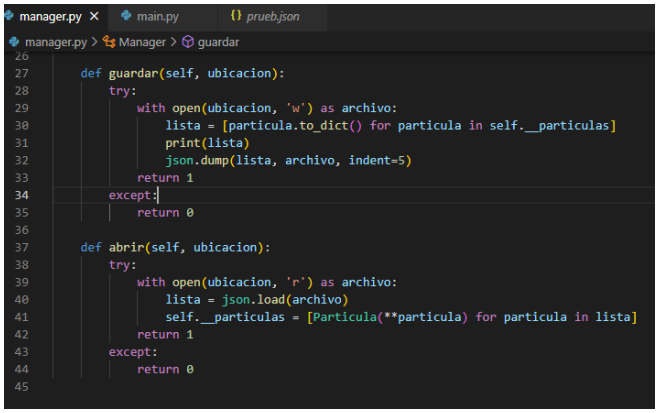
Se muestra la captura de pantalla de las partículas con el método `mostrar()` previo a generar el respaldo.



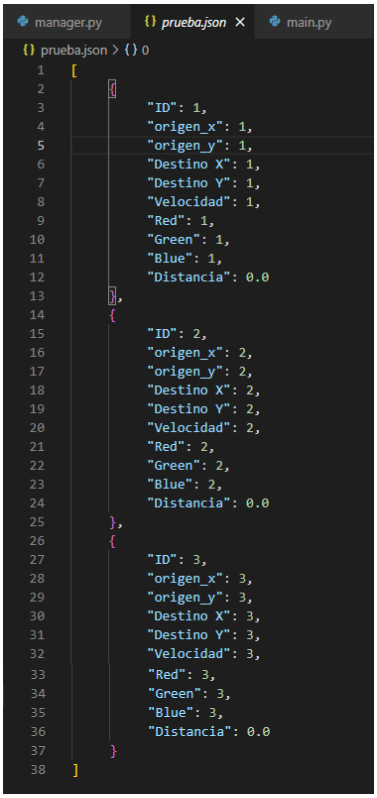

```
ThinkPad T440@DESKTOP-MA4RIIA MINGW64 ~/Desktop/UDGING/4to/SEM-ALOGIRTMIA/ac7 (master)
$ "C:/Users/ThinkPad T440/AppData/Local/Programs/Python/Python310/python.exe" "c:/Users/ThinkPad T440/Desktop/UDGING/4to/SEM-ALOGIRTMIA/ac7/
main.py"
C:/Users/ThinkPad T440/Desktop/UDGING/4to/SEM-ALOGIRTMIA/ac7/prueba.json
[{'ID': 1, 'origen_x': 1, 'origen_y': 1, 'Destino X': 1, 'Destino Y': 1, 'Velocidad': 1, 'Red': 1, 'Green': 1, 'Blue': 1, 'Distancia': 0.0},
 {'ID': 2, 'origen_x': 2, 'origen_y': 2, 'Destino X': 2, 'Destino Y': 2, 'Velocidad': 2, 'Red': 2, 'Green': 2, 'Blue': 2, 'Distancia': 0.0},
 {'ID': 3, 'origen_x': 3, 'origen_y': 3, 'Destino X': 3, 'Destino Y': 3, 'Velocidad': 3, 'Red': 3, 'Green': 3, 'Blue': 3, 'Distancia': 0.0}]
```

Se muestran capturas de pantallas de los pasos que se realizan en la interfaz para generar el respaldo.

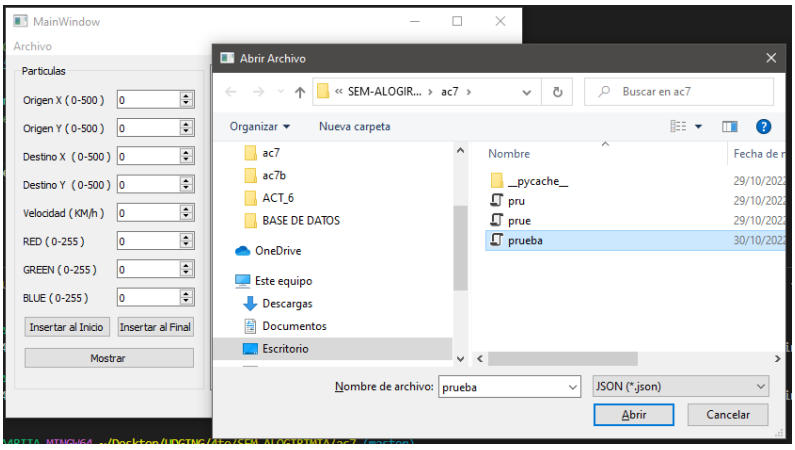


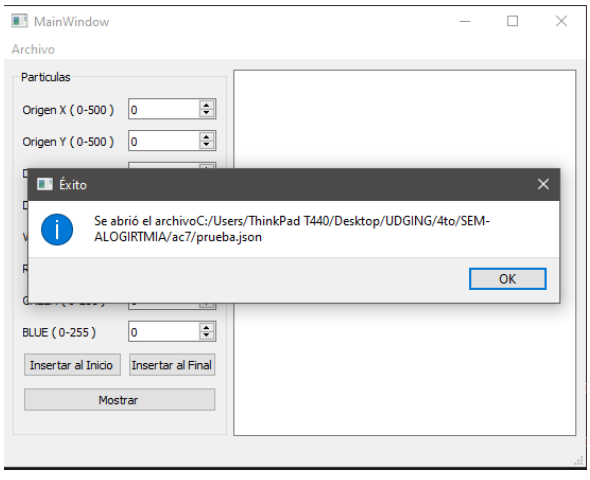


Se muestra el contenido del archivo *.json.*

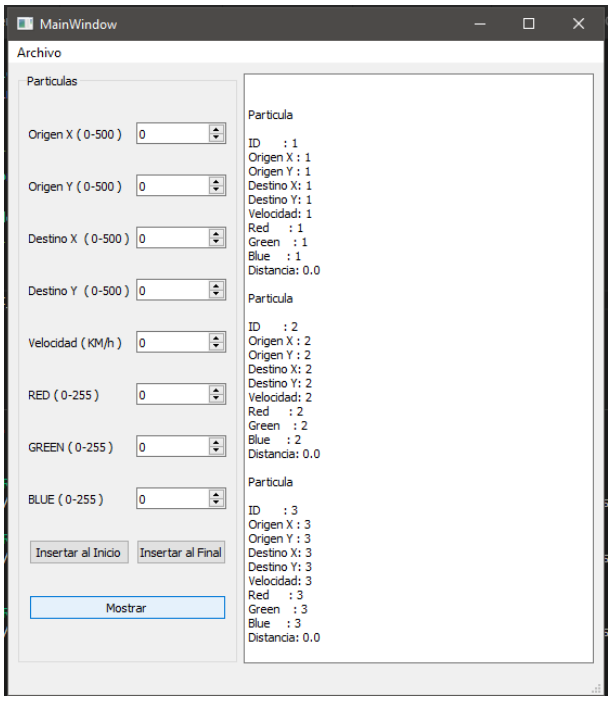


Se muestran capturas de pantallas de los pasos que se realizan en la interfaz para abrir el archivo de respaldo *.json*.





Se muestra la captura de pantalla de las partículas con el método `mostrar()` después de abrir el respaldo



Conclusiones.

Este trabajo el principio parecía fácil, y lo fue, pero solo al principio, desde el inicio todo lo pude realizar con mucha facilidad, hasta el que llegué al punto de abrir un archivo ".json" desde la interfaz creada en Qt, se me complicó bastante, y tuve varios errores, hasta que pude resolverlo, el error estaba en las kell, donde no coincidían algunas, ya que las renombre en to_dict fue hasta que funcionó.

Referencias.

Michel Davalos Boites.[MICHEL DAVALOS BOITES](28/10/2022) PySide2 - QFileDialog(Qt for Python)(IV)[Archivo de video]. https://www.youtube.com/watch?v=HRY8QvXmcDM&ab_channel=MICHELDAVALOSBOITES

Código.

Main.py

```
from PySide2.QtWidgets import QPushButton, QApplication
from mainwindow import MainWindow
import sys

#Aplicación de QT
app = QApplication()
#Crear objeto
window = MainWindow()
#Hacer visible el elemento Botón
window.show()
#Qt loop
sys.exit(app.exec_())
```

mainwindow.py

```
from math import fabs
from PySide2.QtWidgets import QMainWindow, QFileDialog, QMessageBox
from PySide2.QtCore import Slot
from ui_mainwindow import Ui_MainWindow
from manager import Manager
from particula import Particula

class MainWindow(QMainWindow):
    def __init__(self):
        super(MainWindow, self).__init__()
        self.manager = Manager()
        self.ui = Ui_MainWindow()
        self.ui.setupUi(self)
        self.id = 0

        #Cuando el botón pushbutton es presionado, ejecuta la función
click_agregar
        # self.ui.mostrar.clicked.connect(self.click_mostrar)
        self.ui.insertar_inicio.clicked.connect(self.click_insertar_inicio)
        self.ui.insertar_final.clicked.connect(self.click_insertar_final)
        self.ui.mostrar.clicked.connect(self.click_mostrar)
        self.ui.actionAbrir.triggered.connect(self.action_abrir_archivo)
        self.ui.actionGuardar.triggered.connect(self.action_guardar_archivo)

    #Funcion que es llamada por x razón que imprime Click en Terminal.
    @Slot()
```

```python
    # def click_mostrar(self):
    #     a

    @Slot()
    def action_abrir_archivo(self):
        #print("Abrir_archivo")
        ubicacion = QFileDialog.getOpenFileName(
            self,
            'Abrir Archivo',
            '.',
            'JSON (*.json)'
        ) [0]
        if self.manager.abrir(ubicacion):
            QMessageBox.information(
                self,
                "Éxito",
                "Se abrió el archivo" + ubicacion
            )
        else:
            QMessageBox.critical(
                self,
                "Error",
                "Error al abrir el archivo" + ubicacion
            )


    @Slot()
    def action_guardar_archivo(self):
        #print("guardar_archivo")
        ubicacion = QFileDialog.getSaveFileName(
            self,
            'Guardar Archivo',
            '.',
            'JSON (*.json)'
        )[0]
        print(ubicacion)
        if self.manager.guardar(ubicacion):
            QMessageBox.information(
                self,
                "Exito",
                "Se pudo crear el archivi" + ubicacion
            )
        else:
            QMessageBox.critical(
                self,
```

```
                "Error",
                "No se pudo crear el archivo" + ubicacion
            )


    def click_insertar_inicio(self):
        self.id += 1
        aux = Particula(self.id, self.ui.ox.value(), self.ui.oy.value(),
self.ui.dx.value(), self.ui.dy.value(), self.ui.velocidad.value(),
self.ui.red.value(), self.ui.green.value(), self.ui.blue.value())
        self.manager.agregarInicio(aux)
        self.click_mostrar()

    def click_insertar_final(self):
        self.id += 1
        aux = Particula(self.id , self.ui.ox.value(), self.ui.oy.value(),
self.ui.dx.value(), self.ui.dy.value(), self.ui.velocidad.value(),
self.ui.red.value(), self.ui.green.value(), self.ui.blue.value())
        self.manager.agregarFinal(aux)
        self.click_mostrar()

    def click_mostrar(self):
        self.ui.lista_particulas.clear()
        self.ui.lista_particulas.insertPlainText(str(self.manager))
```

manager.py

```
import imp
from turtle import st
from particula import Particula
import json


class Manager:

    def __init__(self):
        self.__particulas = []

    def agregarInicio(self, particula: Particula):
        self.__particulas.insert(0, particula)

    def agregarFinal(self, particula: Particula):
        self.__particulas.append(particula)
```

```
    def imprimir(self):
        for particula in self.__particulas:
            print(particula)

    def __str__(self):
        return "".join(
            str(particula) for particula in self.__particulas
        )

    def guardar(self, ubicacion):
        try:
            with open(ubicacion, 'w') as archivo:
                lista = [particula.to_dict() for particula in
self.__particulas]
                print(lista)
                json.dump(lista, archivo, indent=5)
            return 1
        except:
            return 0

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

particula.py

```
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(origen_x, origen_y,
destino_x, destino_y)

    def __str__(self):
        return ('\n\nParticula\n' +
                '\nID       : ' + str(self.__id) +
                '\nOrigen X : ' + str(self.__origen_x) +
                '\nOrigen Y : ' + str(self.__origen_y) +
                '\nDestino X: ' + str(self.__destino_x) +
                '\nDestino Y: ' + str(self.__destino_y) +
                '\nVelocidad: ' + str(self.__velocidad) +
                '\nRed      : ' + str(self.__red) +
                '\nGreen    : ' + str(self.__green) +
                '\nBlue     : ' + str(self.__blue) +
                '\nDistancia: ' + str(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

```
# -*- coding: utf-8 -*-

################################################################################
####
## Form generated from reading UI file 'mainwindow.ui'
##
## Created by: Qt User Interface Compiler version 5.15.2
##
```

```
## WARNING! All changes made in this file will be lost when recompiling UI
file!
################################################################################
####

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(549, 405)
        self.actionGuardar = QAction(MainWindow)
        self.actionGuardar.setObjectName(u"actionGuardar")
        self.actionAbrir = QAction(MainWindow)
        self.actionAbrir.setObjectName(u"actionAbrir")
        self.centralwidget = QWidget(MainWindow)
        self.centralwidget.setObjectName(u"centralwidget")
        self.gridLayout_2 = QGridLayout(self.centralwidget)
        self.gridLayout_2.setObjectName(u"gridLayout_2")
        self.lista_particulas = QPlainTextEdit(self.centralwidget)
        self.lista_particulas.setObjectName(u"lista_particulas")

        self.gridLayout_2.addWidget(self.lista_particulas, 0, 1, 1, 1)

        self.groupBox = QGroupBox(self.centralwidget)
        self.groupBox.setObjectName(u"groupBox")
        self.gridLayout = QGridLayout(self.groupBox)
        self.gridLayout.setObjectName(u"gridLayout")
        self.green = QSpinBox(self.groupBox)
        self.green.setObjectName(u"green")
        self.green.setMaximum(255)

        self.gridLayout.addWidget(self.green, 6, 1, 1, 1)

        self.dy = QSpinBox(self.groupBox)
        self.dy.setObjectName(u"dy")
        self.dy.setMaximum(500)

        self.gridLayout.addWidget(self.dy, 3, 1, 1, 1)

        self.label_7 = QLabel(self.groupBox)
```

```
        self.label_7.setObjectName(u"label_7")

        self.gridLayout.addWidget(self.label_7, 1, 0, 1, 1)

        self.blue = QSpinBox(self.groupBox)
        self.blue.setObjectName(u"blue")
        self.blue.setMaximum(255)

        self.gridLayout.addWidget(self.blue, 7, 1, 1, 1)

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

        self.gridLayout.addWidget(self.label_5, 6, 0, 1, 1)

        self.insertar_final = QPushButton(self.groupBox)
        self.insertar_final.setObjectName(u"insertar_final")

        self.gridLayout.addWidget(self.insertar_final, 8, 1, 1, 1)

        self.label_3 = QLabel(self.groupBox)
        self.label_3.setObjectName(u"label_3")

        self.gridLayout.addWidget(self.label_3, 4, 0, 1, 1)

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

        self.gridLayout.addWidget(self.label_2, 3, 0, 1, 1)

        self.red = QSpinBox(self.groupBox)
        self.red.setObjectName(u"red")
        self.red.setMaximum(255)

        self.gridLayout.addWidget(self.red, 5, 1, 1, 1)

        self.mostrar = QPushButton(self.groupBox)
        self.mostrar.setObjectName(u"mostrar")

        self.gridLayout.addWidget(self.mostrar, 9, 0, 1, 2)

        self.label_4 = QLabel(self.groupBox)
        self.label_4.setObjectName(u"label_4")

        self.gridLayout.addWidget(self.label_4, 5, 0, 1, 1)
```

```
        self.label_6 = QLabel(self.groupBox)
        self.label_6.setObjectName(u"label_6")

        self.gridLayout.addWidget(self.label_6, 7, 0, 1, 1)

        self.velocidad = QSpinBox(self.groupBox)
        self.velocidad.setObjectName(u"velocidad")
        self.velocidad.setMaximum(999)

        self.gridLayout.addWidget(self.velocidad, 4, 1, 1, 1)

        self.label = QLabel(self.groupBox)
        self.label.setObjectName(u"label")

        self.gridLayout.addWidget(self.label, 2, 0, 1, 1)

        self.dx = QSpinBox(self.groupBox)
        self.dx.setObjectName(u"dx")
        self.dx.setMaximum(500)

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

        self.insertar_inicio = QPushButton(self.groupBox)
        self.insertar_inicio.setObjectName(u"insertar_inicio")

        self.gridLayout.addWidget(self.insertar_inicio, 8, 0, 1, 1)

        self.label_8 = QLabel(self.groupBox)
        self.label_8.setObjectName(u"label_8")

        self.gridLayout.addWidget(self.label_8, 0, 0, 1, 1)

        self.oy = QSpinBox(self.groupBox)
        self.oy.setObjectName(u"oy")
        self.oy.setMaximum(500)

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

        self.ox = QSpinBox(self.groupBox)
        self.ox.setObjectName(u"ox")
        self.ox.setMaximum(500)

        self.gridLayout.addWidget(self.ox, 0, 1, 1, 1)
```

```
        self.gridLayout_2.addWidget(self.groupBox, 0, 0, 1, 1)

        MainWindow.setCentralWidget(self.centralwidget)
        self.menubar = QMenuBar(MainWindow)
        self.menubar.setObjectName(u"menubar")
        self.menubar.setGeometry(QRect(0, 0, 549, 21))
        self.menuArchivo = QMenu(self.menubar)
        self.menuArchivo.setObjectName(u"menuArchivo")
        MainWindow.setMenuBar(self.menubar)
        self.statusbar = QStatusBar(MainWindow)
        self.statusbar.setObjectName(u"statusbar")
        MainWindow.setStatusBar(self.statusbar)

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

        self.retranslateUi(MainWindow)

        QMetaObject.connectSlotsByName(MainWindow)
    # setupUi

    def retranslateUi(self, MainWindow):
        MainWindow.setWindowTitle(QCoreApplication.translate("MainWindow",
u"MainWindow", None))
        self.actionGuardar.setText(QCoreApplication.translate("MainWindow",
u"Guardar", None))
#if QT_CONFIG(shortcut)
        self.actionGuardar.setShortcut(QCoreApplication.translate("MainWindo
w", u"Ctrl+S", None))
#endif // QT_CONFIG(shortcut)
        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.groupBox.setTitle(QCoreApplication.translate("MainWindow",
u"Particulas", None))
        self.label_7.setText(QCoreApplication.translate("MainWindow",
u"Origen Y ( 0-500 )", None))
        self.label_5.setText(QCoreApplication.translate("MainWindow",
u"GREEN ( 0-255 )", None))
        self.insertar_final.setText(QCoreApplication.translate("MainWindow",
u"Insertar al Final", None))
```

```
        self.label_3.setText(QCoreApplication.translate("MainWindow",
u"Velocidad ( KM/h )", None))
        self.label_2.setText(QCoreApplication.translate("MainWindow",
u"Destino Y  ( 0-500 )", None))
        self.mostrar.setText(QCoreApplication.translate("MainWindow",
u"Mostrar", None))
        self.label_4.setText(QCoreApplication.translate("MainWindow", u"RED
( 0-255 )", None))
        self.label_6.setText(QCoreApplication.translate("MainWindow", u"BLUE
( 0-255 )", None))
        self.label.setText(QCoreApplication.translate("MainWindow",
u"Destino X  ( 0-500 )", None))
        self.insertar_inicio.setText(QCoreApplication.translate("MainWindow"
, u"Insertar al Inicio", None))
        self.label_8.setText(QCoreApplication.translate("MainWindow",
u"Origen X ( 0-500 )", None))
        self.menuArchivo.setTitle(QCoreApplication.translate("MainWindow",
u"Archivo", None))
    # retranslateUi
```

prueba.json

```
[
    {
        "id": 1,
        "origen_x": 1,
        "origen_y": 1,
        "destino_x": 1,
        "destino_y": 1,
        "velocidad": 1,
        "red": 1,
        "green": 1,
        "blue": 1
    },
    {
        "id": 2,
        "origen_x": 2,
        "origen_y": 2,
        "destino_x": 2,
        "destino_y": 2,
        "velocidad": 2,
        "red": 2,
        "green": 2,
        "blue": 2
    },
```

```
    {
        "id": 3,
        "origen_x": 3,
        "origen_y": 3,
        "destino_x": 3,
        "destino_y": 3,
        "velocidad": 3,
        "red": 3,
        "green": 3,
        "blue": 3
    }
]
```