

# » UI: User Interface

---

Cruz Collazo Wendy Paola.

*Seminario de solución de Problemas de Algoritmia.*

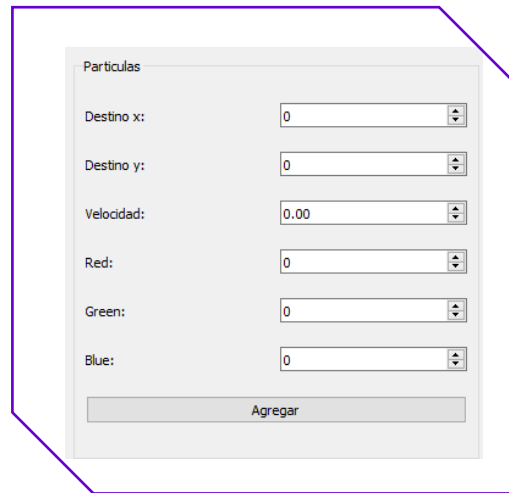
## **Lineamientos de Evaluación:**

- 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 la captura de pantalla de la interfaz de usuario corriendo desde **Python** con los widgets para representar la información de una Partícula (como se definió en el punto 1 de las instrucciones).



## Desarrollo

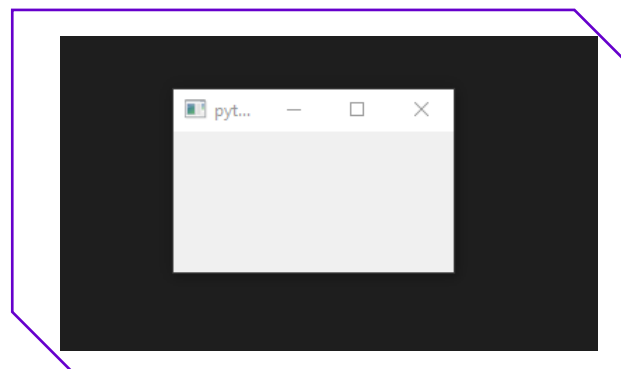
Comencé en Designer creando la Interfaz guiándome con el video, para eso use el “Label” para poner los datos que iba a pedir, después use “spin box” para los datos de entrada del usuario, y use un “doublé spin box” para lo valores de velocidad. Una vez teniendo esto tuve que juntarlos para que se viera mas ordenado y utilice el “combo box” para poder hacer esto. Alinee tanto el combo como el main para que me quedara todo alineado y el recuadro.



Vista desde el QT  
Designer

Teniendo la interface aquí, me pase a visual studio donde ya tenía el main.c. y ya que el código que tenía no me servía del todo para poder mostrar mi interface de Qt Designer. Primero tuve que crear una clase del el Mainwindow para poder meter la interfaz en el código que previamente ya tenía.

Esta clase me ayuda poder controlar la vista que hice en Qt.

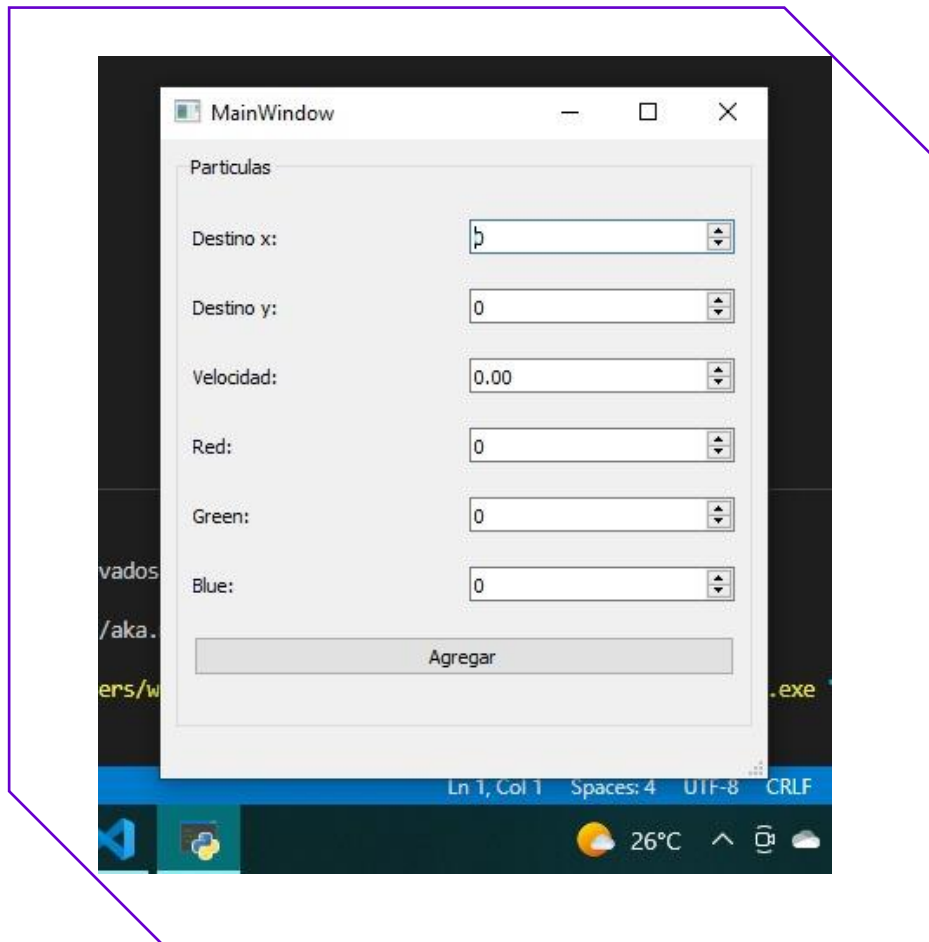


Antes de Meter  
la Interface

Para poder meter la interface primero tenia que convertir el archivo .ui a un archivo .py así que usé el comando “pyside2-uic Mainwindow.ui > ui\_mainwindow.py” para poder tener el código en .py en esta clase no moví absolutamente nada. Y en caso de modificar la interface el QT se tendrá que volver a usar el comando para “actualizar”.

Regresando a la clase que se había creado importamos el archivo que convertimos. Y dentro de la definición de esta clase, le agregamos un método ‘setUpUi’ la cual no ayuda a meterle los componentes de la vista.

Ahora faltaría crear un método para el botón de “Aceptar” y este se crearía el método en la clase. Importamos la Palabra Slot que esta nos va a ayudar al momento de que el usuario de aceptar lo detecte y ya solo la tuvimos que conectar para que el slot se conectara con la función del clic.



## ● *Conclusión:*

En esta actividad si batallé un poquito al momento de querer configurar el visual para que no me marcara error en el “PySide” ya que a mí no me aparecía como en el video, pero ya después de varios intentos pude configurarlo.

Al momento de querer correr cada que hacia una clase o modificaba algo no me abría la pantallita de la interface, pero después de que terminé que ver el video y hacer los pasos me tuve que pasar a mi lap para ver si ahí me dejaba ver y pues en la lap si pude.

Fue algo nuevo para mi ya que nunca había creado una interface y mucho menos con código.

## ● *Referencias:*

- Davalos Boites, Michel- Introducción (QT For Python) (I).  
<https://www.youtube.com/watch?v=T0qJdF1fMqo>

## ● Código

### main.py

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

#Aplicacion de Qt
app = QApplication()
#Se crea un Boton con la palabra Hola
window = MainWindow()
#Se hace visible el boton
window.show()
#Qt loop
sys.exit(app.exec_())
```

### Mainwindow.p

```
from PySide2.QtWidgets import QMainWindow
from PySide2.QtCore import Slot
from ui_mainwindow import Ui_MainWindow

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

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

    @Slot()
    def click_agregar(self):
        print ('Click')
```

## 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.setObjectName():
            MainWindow.setObjectName(u"MainWindow")
            MainWindow.resize(350, 368)
            self.centralwidget = QWidget(MainWindow)
            self.centralwidget.setObjectName(u"centralwidget")
            self.gridLayout_2 = QGridLayout(self.centralwidget)
            self.gridLayout_2.setObjectName(u"gridLayout_2")
            self.groupBox = QGroupBox(self.centralwidget)
            self.groupBox.setObjectName(u"groupBox")
            self.gridLayout = QGridLayout(self.groupBox)
            self.gridLayout.setObjectName(u"gridLayout")
            self.label = QLabel(self.groupBox)
            self.label.setObjectName(u"label")

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

            self.spinBox = QSpinBox(self.groupBox)
            self.spinBox.setObjectName(u"spinBox")

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

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

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

self.spinBox_2 = QSpinBox(self.groupBox)
self.spinBox_2.setObjectName(u"spinBox_2")

self.gridLayout.addWidget(self.spinBox_2, 1, 1, 1, 1)

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

self.gridLayout.addWidget(self.label_3, 2, 0, 1, 1)

self.doubleSpinBox = QDoubleSpinBox(self.groupBox)
self.doubleSpinBox.setObjectName(u"doubleSpinBox")

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

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

self.gridLayout.addWidget(self.label_4, 3, 0, 1, 1)

self.spinBox_3 = QSpinBox(self.groupBox)
self.spinBox_3.setObjectName(u"spinBox_3")

self.gridLayout.addWidget(self.spinBox_3, 3, 1, 1, 1)

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

self.gridLayout.addWidget(self.label_5, 4, 0, 1, 1)

self.spinBox_4 = QSpinBox(self.groupBox)
self.spinBox_4.setObjectName(u"spinBox_4")

self.gridLayout.addWidget(self.spinBox_4, 4, 1, 1, 1)

self.label_6 = QLabel(self.groupBox)
```

```

self.label_6.setObjectName(u"label_6")

self.gridLayout.addWidget(self.label_6, 5, 0, 1, 1)

self.spinBox_5 = QSpinBox(self.groupBox)
self.spinBox_5.setObjectName(u"spinBox_5")

self.gridLayout.addWidget(self.spinBox_5, 5, 1, 1, 1)

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

self.gridLayout.addWidget(self.pushButton, 6, 0, 1, 2)

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, 350, 21))
MainWindow.setMenuBar(self.menubar)
self.statusbar = QStatusBar(MainWindow)
self.statusbar.setObjectName(u"statusbar")
MainWindow.setStatusBar(self.statusbar)

self.retranslateUi(MainWindow)

QMetaObject.connectSlotsByName(MainWindow)
# setupUi

def retranslateUi(self, MainWindow):

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

self.groupBox.setTitle(QCoreApplication.translate("MainWindow",
u"Particulas", None))
    self.label.setText(QCoreApplication.translate("MainWindow",
u"Destino x:", None))
    self.label_2.setText(QCoreApplication.translate("MainWindow",
u"Destino y:", None))

```



```
        self.label_3.setText(QCoreApplication.translate("MainWindow",  
u"Velocidad:", 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.pushButton.setText(QCoreApplication.translate("MainWindow",  
u"Agregar", None))  
    # retranslateUi
```