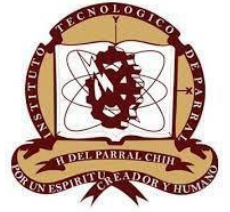




TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE PARRAL



TEMA: “Utilizar una librería para realizar traducciones y practicar la creación de interfaces gráficas de usuario.”.

CARRERA: Ingeniería Sistemas Computacionales.

MATERIA: Tópicos avanzados de programación.

DOCENTE: Ing. Edgar Omar Bañuelos Lozoya

ALUMNO:

Luis Enrique Gallardo Ramírez

Nos. De Control:

21410483

Fecha de entrega: 13/10/2023

Tiempo de desarrollo en horas: 4-5

Conclusiones: A pesar de las complicaciones y retos de la actividad, se llegó al resultado esperado, aunque carece de detalles como alineaciones, el funcionamiento es el planteado al principio en teoría. Se aprendió el uso de nuevas librerías como “Googletrans” para la traducción de textos.

Referencias utilizadas:

PySide6 y PyQt5: Estas son bibliotecas para la creación de aplicaciones de escritorio con GUI en Python.

PySide6: <https://doc.qt.io/qtforpython/>

PyQt5: <https://www.riverbankcomputing.com/static/Docs/PyQt5/>

googletrans: Esta biblioteca se utiliza para realizar traducciones a través de Google Translate:

Repositorio de googletrans: <https://github.com/ssut/py-googletrans>

Qt Alignments (Alineaciones en Qt): Para alinear widgets en un diseño de Qt, puedes consultar la documentación oficial de Qt. El alineamiento se realiza mediante la clase Qt.AlignmentFlag:

Documentación de Qt sobre alineaciones:
<https://doc.qt.io/qt-5/qt.html#AlignmentFlag-enum>

```

import sys
from PySide6.QtCore import Qt
from PyQt5.QtWidgets import QApplication, QMainWindow, QComboBox, QLineEdit,
QPushButton, QTextEdit, QWidget, QGridLayout
from googletrans import Translator

idiomas = {
    'español': 'es',
    'inglés': 'en',
    'portugués': 'pt',
    'frances': 'fr'
}

class AplicacionTraductor(QMainWindow):
    def __init__(self):
        super().__init__()
        self.setGeometry(100, 100, 600, 400)
        self.setWindowTitle('Traductor')
        self.setFixedSize(400, 500)

        #Idiomas a utilizar
        self.idiomas = ['español', 'inglés', 'portugués', 'frances']

        #Creación y adición de los idiomas al combo box
        self.combo_idioma_origen = QComboBox()
        self.combo_idioma_origen.addItem(self.idiomas)
        self.combo_idioma_origen.setCurrentIndex(0)
        self.combo_idioma_origen.setFixedSize(100, 20)

        self.combo_idioma_origen.currentIndexChanged.connect(self.ajustarComboOrigen
        )

        self.entrada_texto = QTextEdit()
        self.boton_traducir = QPushButton('Traducir')
        self.boton_traducir.setFixedSize(80, 50)

        self.boton_traducir.clicked.connect(self.traducir_texto)

        #Segundo combo box con los idiomas para mostrar ya traducido
        self.combo_idioma_destino = QComboBox()
        self.combo_idioma_destino.addItem(self.idiomas)
        self.combo_idioma_destino.setCurrentIndex(1)
        self.combo_idioma_destino.setFixedSize(100, 20)

        self.combo_idioma_destino.currentIndexChanged.connect(self.ajustarComboDesti
        no)

        #QLineedit en modo de lectura

```

```

self.salida_texto = QTextEdit()
self.salida_texto.setReadOnly(True)

#Layout con los componentes dentro
self.layout = QGridLayout()
self.layout.addWidget(self.combo_idioma_origen, 0, 0)
self.layout.addWidget(self.entrada_texto, 1, 0)
self.layout.addWidget(self.boton_traducir, 2, 0)
self.layout.addWidget(self.combo_idioma_destino, 3, 0)
self.layout.addWidget(self.salida_texto, 4, 0)
contenedor = QWidget()
contenedor.setLayout(self.layout)
self.setCentralWidget(contenedor)

#Función para traducir el texto
def traducir_texto(self):
    idioma_origen = idiomas[self.combo_idioma_origen.currentText()]
    idioma_destino = idiomas[self.combo_idioma_destino.currentText()]
    texto_a_traducir = self.entrada_texto.text()

    traductor = Translator()
    traduccion = traductor.translate(texto_a_traducir,
src=idioma_origen, dest=idioma_destino)
    self.salida_texto.setPlainText(traduccion.text)

#Metodos para evitar la igualdad entre combo box o idiomas
def ajustarComboOrigen(self):
    idioma_origen = self.combo_idioma_origen.currentText()
    idioma_destino = self.combo_idioma_destino.currentText()

    if idioma_origen == idioma_destino:
        nuevo_indice = (self.combo_idioma_destino.currentIndex() + 1) %
len(self.idiomas)
        self.combo_idioma_destino.setCurrentIndex(nuevo_indice)

def ajustarComboDestino(self):
    idioma_origen = self.combo_idioma_origen.currentText()
    idioma_destino = self.combo_idioma_destino.currentText()

    if idioma_origen == idioma_destino:
        nuevo_indice = (self.combo_idioma_destino.currentIndex() + 1) %
len(self.idiomas)
        self.combo_idioma_origen.setCurrentIndex(nuevo_indice)

if __name__ == '__main__':
    app = QApplication(sys.argv)
    aplicacion_traductor = AplicacionTraductor()
    aplicacion_traductor.show()

```

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