

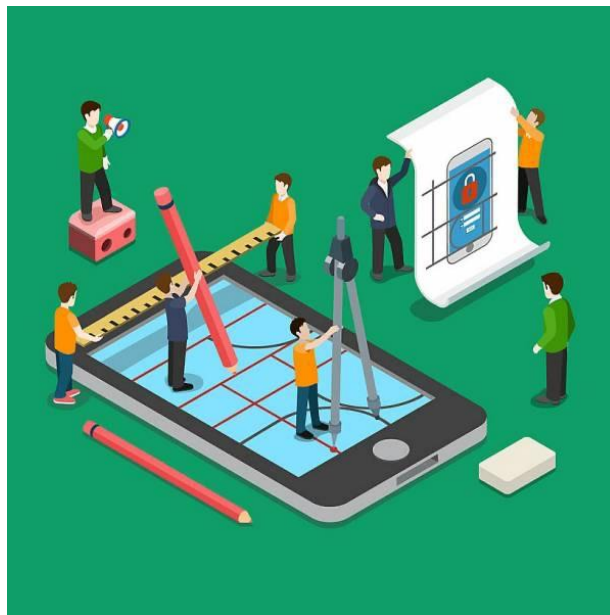


DESENROTLLAMENT D'INTERFÍCIES

2N DAM

CURS 2023-2024

**UD2: Creació d'interfícies gràfiques amb Python
utilitzant PySide.**



Instruccions inicials

Les pràctiques es lliuraran seguint la següent pauta: El nom de l'activitat o pràctica a enviar serà el nombre de la pràctica (per exemple: R1, seguit de guió baix i del vostre nom i cognoms). Per exemple: **R1_AntonioGómez**.

L'activitat s'ha de lliurar en la data establerta i no s'acceptaran els lliuraments fora del termini establert.

Les pràctiques han de ser originals i no s'admeten còpies de companys/es, Internet..

Per a les pràctiques farem ús del programari indicat per professorat i especificat a les pràctiques.

Per a realitzar aquest repte hem d'utilitzar Visual Studio Code i no farem ús de cap IDE específic.

Objectius de l'activitat

Crear interfícies gràfiques amb PySide i Qt.

Aprendre i familiaritzar-se amb els principals components de PySide6.

Crear aplicacions amb menús, toolbars i funcionalitats concretes amb PySide6.

Temporalització

L'activitat està prevista en una estimació de tres sessions lectives de 55 minuts cada sessió.

CREACIÓ D'INTERFÍCIES GRÀFIQUES AMB PYSIDE6:

REpte 1: EDITOR DE TEXT AMB PYSIDE6

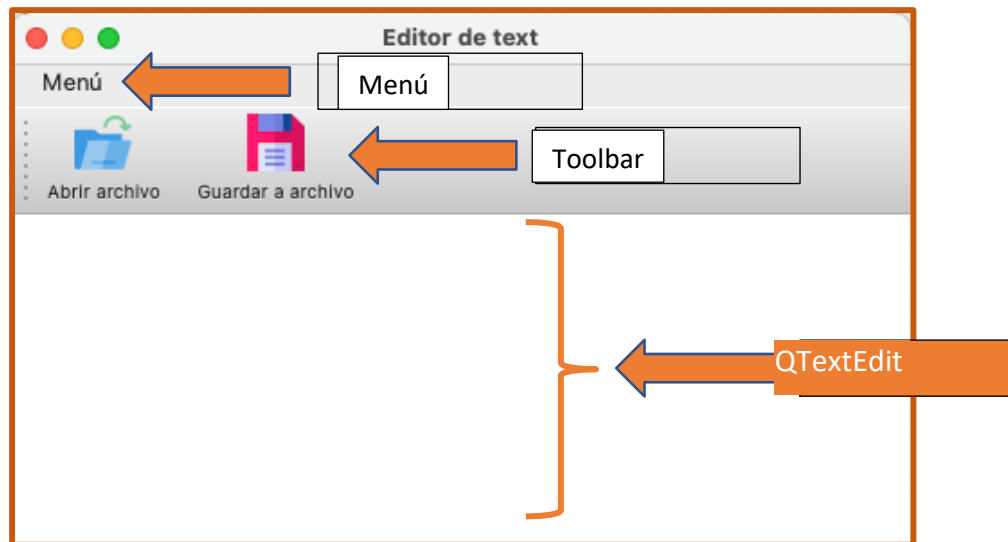


Aquest repte consisteix en crear un editor de text senzill que ens permeti obrir fitxers de text (.txt), una vegada oberts els podem editar i guardar amb els canvis realitzats.

En la següent imatge veiem l'aspecte que ha de tindre l'editor de text, hem de procurar que siga el més semblant possible a la imatge.



Com podem comprovar disposem d'un **menú** i d'una **toolbar**. També disposem d'un **QTextEdit** en la part central amb una grandària de 500x500.



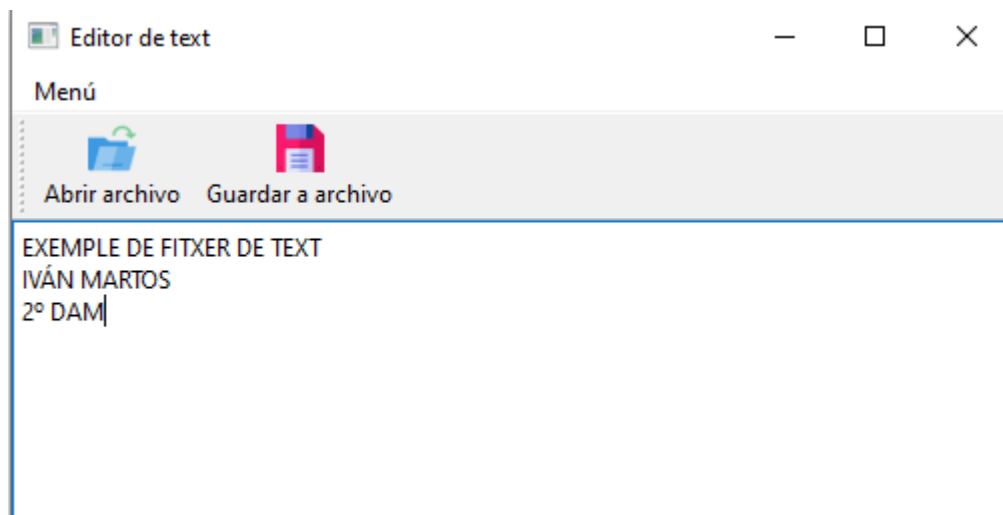
El menú tindrà les següents opcions com podreu comprovar a la imatge següent:

- Obrir arxiu
- Guardar a arxiu
- Tancar (tancar arxiu)
- Eixir

Podreu observar que tres de les quatre opcions tenen **tecles ràpides d'abreviatura (shortcut)** (Control + O, Control + S y Control + Q). En el menú principal NO s'han de mostrar icones.



Si cliquem a obrir arxiu, ens permetrà obrir un arxiu. Prèviament podeu crear un fitxer que s'anomeni fitxer.txt (al mateix directori del projecte) per exemple, i l'obrim per visualitzar el contingut.



Per a obrir el fitxer farem ús del dialogs de PySide, en aquestcas són: **QFileDialog**. Per això disposem de mètodes com **getOpenFileName**. El còdi per obrir fitxers podria ser el següent:

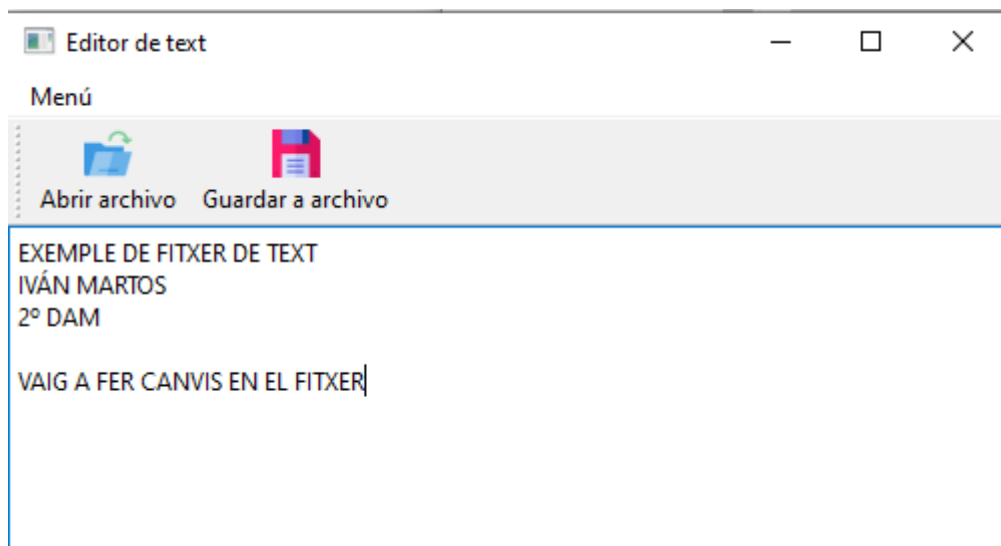
```
def abrir_archivo(self):  
    ventana_dialogo = QFileDialog.getOpenFileName(  
self, caption="Abrir archivo ...", dir=".",  
filter="Documentos de texto (*.txt)",  
selectedFilter="Documentos de texto (*.txt)")  
    self.archivo = ventana_dialogo[0]  
    with open(self.archivo, "r+") as archivo:  
self.centralWidget().setText(archivo.read())
```

Si escrivim sobre el fitxer, després podrem guardar els canvis realitzats sobre el fitxer en qüestió.

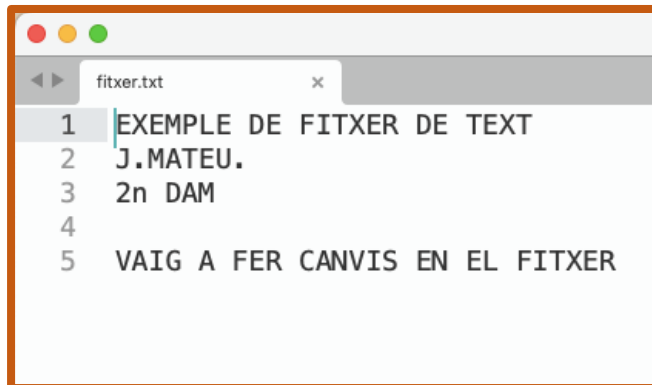
Per a emmagatzemar el fitxer disposem de mètodes com **getSaveFileName**.

El codi es molt paregut al de obrir fitxer, hem de tindre en compte que soles permeti emmagatzemar fitxers .txt, per tant, heu de crear un filtre.

A més, si pressionem sobre el botó de Emmagatzemar i no tenim prèviament ningun arxiu obert, automàticament deuria obrir-se la finestra de diàleg per a deixar-me obrir un arxiu.



Si obrim el fitxer en qualsevol altre editor després de desar, podrem veure que apareixen els canvis realitzats.



Si donem a tancar, es tancarà el fitxer actual i es quedarà buit el QTextEdit. Per últim, si donem a eixir, evidentment es tancarà l'aplicació.

