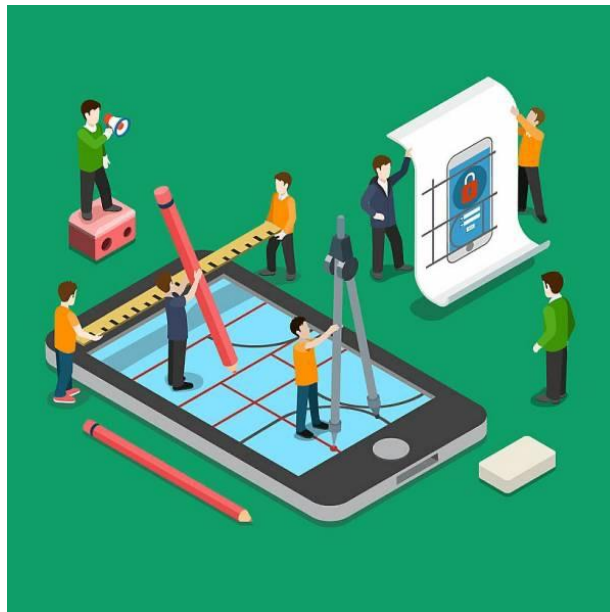




DESENROTLLAMENT D'INTERFÍCIES 2N DAM

SA2: CREACIÓ I US DE COMPONENTS.



Tasca A24: Repte final

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.

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

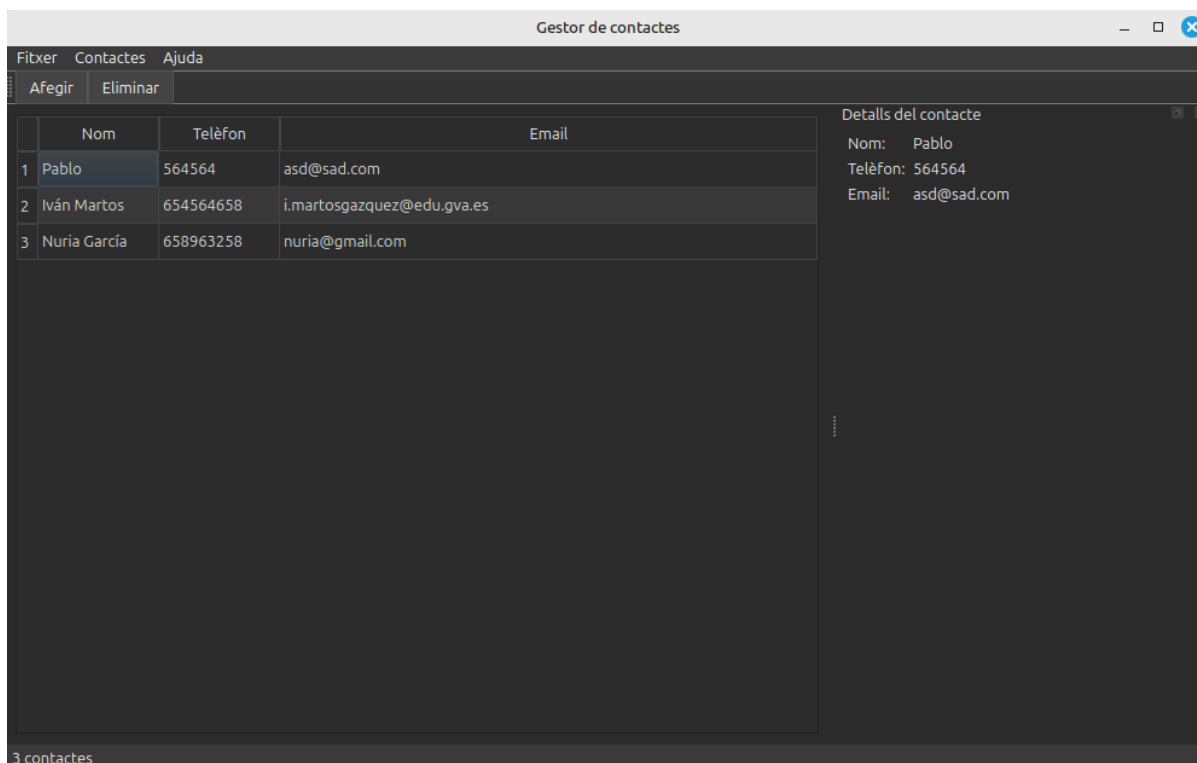
L'activitat està prevista en una estimació de tres sessions lectives de 55 minuts cada sessió.

REPTE 1: EDITOR DE TEXT AMB PYSIDE6

(5 punts)



Aquest repte consisteix en crear un gestor de contactes que permeti guardar i eliminar contactes en una base de dades SQLite. Per això, disposarem d'una taula de contactes on emmagatzemarem l'ID, el nom, el telèfon i el correu electrònic del contacte en qüestió. El aspecte de la interfície ha de ser semblant al següent:



Com podem comprovar disposem d'un **menú** i d'una **toolbar**. També disposem d'un **QTableView** per a mostrar els contactes que tenim a la base de dades SQLite. A la dreta tenim un **QDockWidget** per a mostrar les dades del contacte seleccionat al QTableView.

També disposa d'un **QStatusBar** per a mostrar la quantitat de contactes que hi ha a la base de dades.

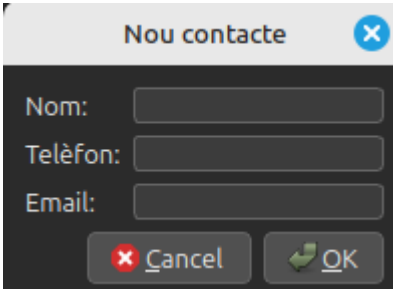
El menú tindrà les següents opcions:

- Fitxer
 - Eixir (Tanca la aplicació)
- Contactes
 - Afegir (Per a afegir un nou contacte). Tindrà un shortcut (Control + A)
 - Eliminar (Per a eliminar el contacte que seleccionem a la taula). Tindrà un shortcut (Control + E)
- Ajuda
 - Informació (Mostra informació de la aplicació a un **QMessageBox**)

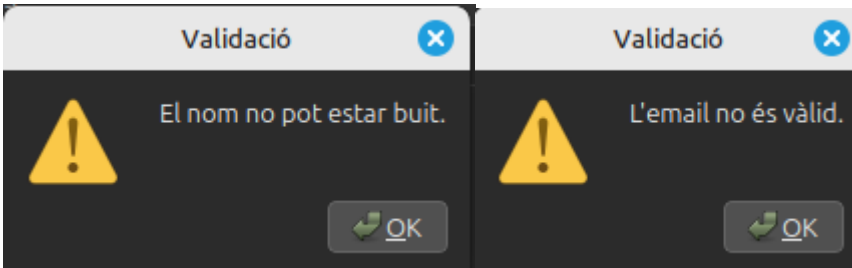
El **ToolBar** tindrà les següents opcions:

- Afegir (Per a afegir un nou contacte)
- Eliminar (Per a eliminar el contacte que seleccionem a la taula).

Si cliqueu sobre Afegir eixirà un **QDialog** com aquest:



Després de validar que el nom no estiga buit i que el correu tinga una sintaxi adequada a la de un correu, el contacte se afegirà a la base de dades i es mostrarà a la interfície automàticament.

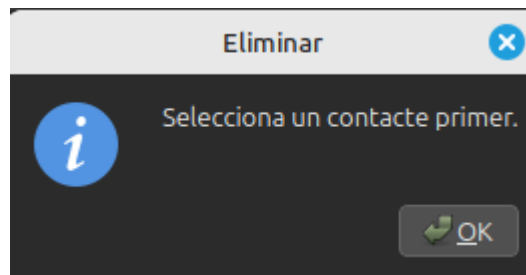


El còdec per a afegir un registre a la base de dades podria ser paregut al següent:

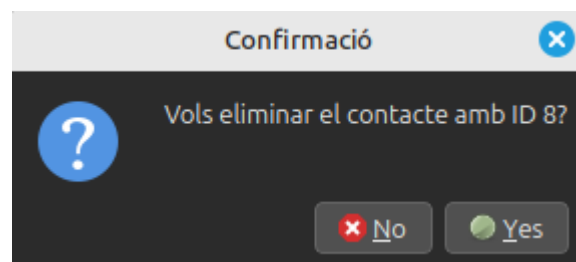


```
def _afegir_contacte(self):
    dlg = ContacteDialog(self)
    if dlg.exec():
        nom, tel, email = dlg.get_dades()
        query = QSqlQuery()
        query.prepare("INSERT INTO contactes (nom, telefon, email) VALUES (?, ?, ?)")
        query.addBindValue(nom)
        query.addBindValue(tel)
        query.addBindValue(email)
        if not query.exec():
            QMessageBox.warning(self, "Error BD", "No s'ha pogut inserir el contacte.")
        self.model.select()
        self._actualitza_estat()
```

Per a eliminar un contacte, en primer lloc hem de seleccionar-lo al QtableView i després pressionar el botó d'eliminar. Si no tenim ningú contacte seleccionat ha d'eixir un **QMessageBox** avisant a l'usuari que ha de seleccionar primer el contacte:



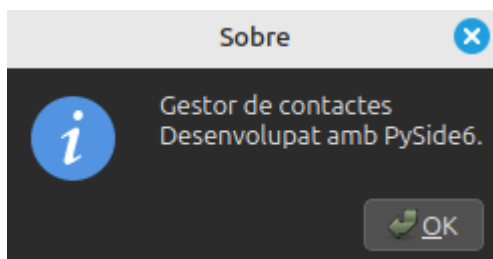
Una vegada seleccionat, abans de eliminar demanarà confirmació:



El còdec per a eliminar un registre de la base de dades podria ser paregut al següent:

```
if confirm == QMessageBox.Yes:
    query = QSqlQuery()
    query.prepare("DELETE FROM contactes WHERE id=?")
    query.addBindValue(contact_id)
    if not query.exec():
        QMessageBox.warning(self, "Error BD", "No s'ha pogut eliminar el contacte.")
    self.model.select()
    self._actualitza_estat()
```

Si cliquem sobre la opció del menú Ajuda/Informació ha de aparèixer un **QMessageBox** com aquest:



Al QDockWidget lateral ha d'aparèixer la informació del contacte seleccionat.

L'aplicació ha de tindre persistència de la finestra i del estat del dock, si canviem de tamany el dock, al reobrir permanixerà igual. (**geometry i windowState**).

Per a els estils, heu d'utilitzar un fitxer .qss personalitzat, podeu donar els estils que vullgau, però al menys heu de personalitzar, el QMainWindow, botons, dock, barra d'estat, cuadros de diàleg, QMessageBox.

Puntuacions del exercici:

Bloc	Criteri	Punts
BD/Model	QSqlTableModel + QTableView operatius, refresc després d'altres/baixes	2
UI bàsica	Menú, Toolbar, StatusBar, QTableWiew i dock correctes i coherents	2
Dock	Mostra correctament detalls del contacte seleccionat	1
StatusBar	Mostra el nombre de contactes i s'actualizta al afegir o eliminar.	1
Diàleg	Validacions: nom i email obligatoris	1
QSS	Personalització d'estils a la aplicació. QMainWindow, botons, dock, barra d'estat, cuadros de diàleg, QMessageBox	1,5
QSettings	Guarda/restaura geometry + windowState sense acció de l'usuari	0,5
Qualitat	Codi net/organitzat, noms clars, comentaris essencials	1