

Mobil programozás haladó

Féléves beadandó



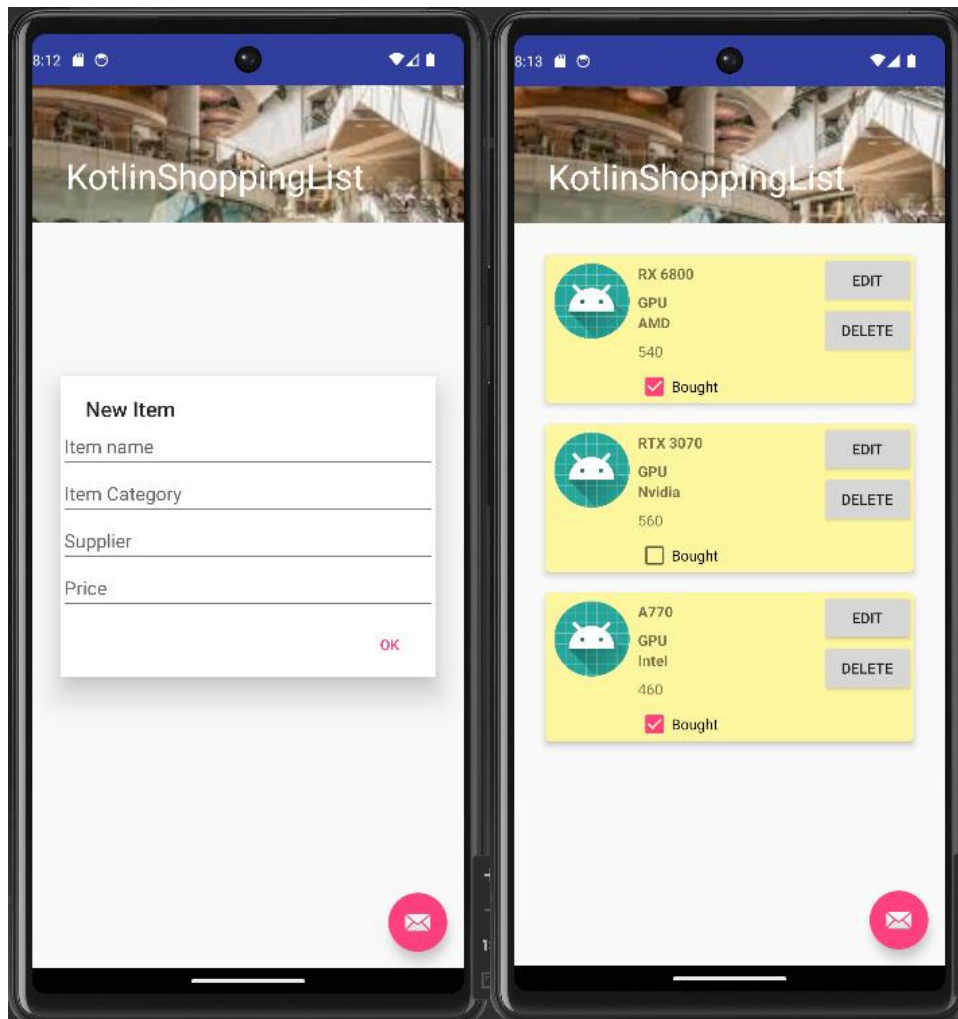
Miskolci Egyetem

Gépészmérnöki és Informatikai Kar

Feladat: Egy órán vett alkalmazás új funkciókkal való kiegészítése

Ahogy órán Tanárnő javasolta feladatomban az órán vett Shopping list applikációt módosítottam.

Módosítás előtt egy termék létrehozásánál meg lehetett adni a termék nevét és árát, módosítás után a termék további tulajdonságait, a termék kategóriáját és beszállítóját lehet megadni.



1. és 2. ábra: Módosítás után megadhatjuk a termék kategóriáját és beszállítóját is

1. lépésben a termék hozzáadásához tartozó XML-hez, a dialog_create_item.xml-hez adtam hozzá a két új kategóriát:

```
<EditText
    android:id="@+id/etCategory"
    android:hint="Item Category"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<EditText
    android:id="@+id/etSupplier"
    android:hint="Supplier"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />
```

3. ábra: Két új kategória a dialog_create_item.xml-ben

Ezután a standard megjelenítés XML-jéhez, a row_item.xml-hez is hozzáadtam a két új elemet:

```
<TextView
    android:id="@+id/tvCategory"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="8dp"
    android:textStyle="bold"
    app:layout_constraintLeft_toRightOf="@+id/ivItemLogo"
    app:layout_constraintTop_toBottomOf="@+id/tvName"
    android:text="@{item.category}" />

<TextView
    android:id="@+id/tvSupplier"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginTop="0dp"
    android:textStyle="bold"
    app:layout_constraintLeft_toRightOf="@+id/ivItemLogo"
    app:layout_constraintTop_toBottomOf="@+id/tvCategory"
    android:text="@{item.supplier}" />
```

4. ábra: A két új elem a row_item.xml-ben

Miskolci Egyetem

Gépészmérnöki és Informatikai Kar

Az új elemeket úgy kényszerítettem hogy azok egymát kövessék sorban, majd a „tvPrice” TextView-nál beállítottam, hogy a „tvSupplier” TextView után következzen. Zárásul a Delete gombnál is beállítottam, hogy közvetlenül az Edit gomb alatt helyezkedjen el, mivel furán nézett ki, hogy eredetileg a témék ablakjának aljához volt igazítva.

```
<Button
    android:id="@+id/btnDelete"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Delete"
    app:layout_constraintTop_toBottomOf="@+id/btnEdit"
    app:layout_constraintEnd_toEndOf="parent" />
```

5. ábra: A Delete gomb az Edit gomb aljához van igazítva

Ezek után az adatbázishoz is hozzáadtam a két új elemet, a ShoppingItem.kt módosításával, ami az adatbázis építőköve:

```
1 package hu.aut.android.kotlinshoppinglist.data
2
3 import ...
4
5
6
7
8 /*db-hez a tábla és mezői*/
9 @Entity(tableName = "shoppingitem")
10 data class ShoppingItem(@PrimaryKey(autoGenerate = true) var itemId: Long?,
11                          @ColumnInfo(name = "name") var name: String,
12                          @ColumnInfo(name = "category") var category: String,
13                          @ColumnInfo(name = "supplier") var supplier: String,
14                          @ColumnInfo(name = "price") var price: Int,
15                          @ColumnInfo(name = "bought") var bought: Boolean
16 ) : Serializable
```

6. ábra: A ShoppingItem.kt-ba is beszúrtam a két elemet

Új termék felvételélél és meglévő termék szerkesztésénél és törlésénél a ShoppingItemDialog.kt végzi el a műveleteket, így ennek a fájlnak is kibővítettem a függvényeit:

```
private lateinit var etItem: EditText
private lateinit var etCategory: EditText
private lateinit var etSupplier: EditText
private lateinit var etPrice: EditText
```

7. ábra: Az új változók a ShoppingItemDialog-ban

```
private fun initDialogContent(builder: AlertDialog.Builder) {
    /*etItem = EditText(activity)
    builder.setView(etItem)*/

    val rootView = activity!!.layoutInflater.inflate(R.layout.dialog_create_item, root: null)
    etItem = rootView.etName
    etCategory = rootView.etCategory
    etSupplier = rootView.etSupplier
    etPrice = rootView.etPrice
    builder.setView(rootView)

    val arguments = this.arguments
    if (arguments != null &&
        arguments.containsKey(MainActivity.KEY_ITEM_TO_EDIT)) {
        val item = arguments.getSerializable(
            MainActivity.KEY_ITEM_TO_EDIT) as ShoppingItem
        etItem.setText(item.name)
        etCategory.setText(item.category)
        etSupplier.setText(item.supplier)
        etPrice.setText(item.price.toString())

        builder.setTitle("Edit todo")
    }
}
```

8. ábra: „Dialog Box” bővítése

```
private fun handleItemCreate() {
    shoppingItemHandler.shoppingItemCreated(ShoppingItem(
        itemId: null,
        etItem.text.toString(),
        etCategory.text.toString(),
        etSupplier.text.toString(),
        etPrice.text.toString().toInt(),
        bought: false
    ))
}
```

9. ábra: Új elem létrehozásának bővítése

```
private fun handleItemEdit() {
    val itemToEdit = (arguments?.getSerializable(
        MainActivity.KEY_ITEM_TO_EDIT) as ShoppingItem).copy(name = etItem.text.toString(), category = etCategory.text.toString(), supplier = etSupplier.text.toString(), price = etPrice.text.toStri
    shoppingItemHandler.shoppingItemUpdated(itemToEdit)
}
```

10. ábra: Meglévő elem szerkesztési funkciójának bővítése