

ЛАБОРАТОРНА РОБОТА № 2

ДОСЛІДЖЕННЯ РОБОТИ З КОМПОНЕНТОМ FRAGMENT

Мета роботи: дослідити створення та взаємодію з компонентом Фрагмент (Fragment) компоненту Діяльність та набуті практичні навички з використання фрагментів для інтерфейсу користувача.

ЗАВДАННЯ

Написати програму під платформу Андроїд, яка має інтерфейс, побудований з декількох фрагментів згідно варіанту. Перший фрагмент представляє з себе форму для введення даних та кнопку підтвердження («ОК»), а інший фрагмент відображає результат взаємодії. Тобто другий фрагмент містить тестове поле з результатом та кнопкою «Cancel» (якщо згідно варіанту така існує, якщо ж за варіантом її немає – можете додати за власним бажанням), яка очищає або приховує (або видаляє) другий фрагмент та очищає форму введення з першого фрагменту. Зверніть увагу, що робота з фрагментами відбувається в рамках однієї Діяльності.

Примітка: завдання відповідає варіанту лабораторної роботи №1.

ХІД ВИКОНАННЯ

Під час виконання роботи спочатку був продубльований графічний інтерфейс додатку з першої лабораторної(рисунок 1.1).

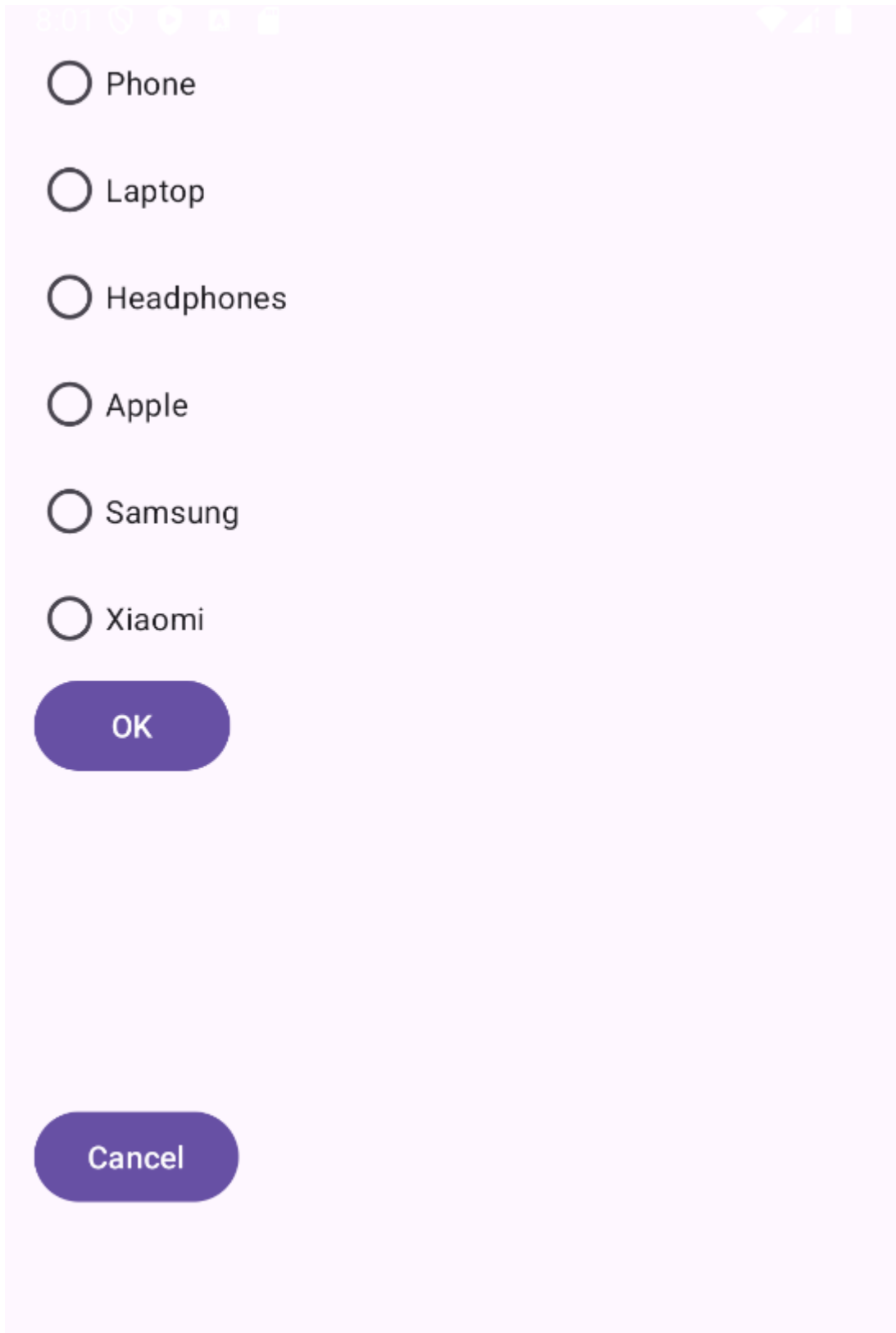


Рисунок 1.1 – Графічний інтерфейс

Після чого було реалізовано 2 фрагменти: `Fragment_input` - цей фрагмент відповідає за введення даних від користувача. Він містить дві групи радіокнопок (`RadioGroup`), де користувач може вибрати тип продукту та компанію. Також є кнопка "OK", яку користувач натискає для підтвердження вибору. `Fragment_result` - Цей фрагмент відповідає за відображення результатів вибору користувача. Він містить `TextView` для показу результату (який продукт та компанію обрав користувач) та кнопку "Cancel",

яка, використовується для скасування вибору (рисунок 1.2). Також було оброблено ситуацію, коли не було обрано відповідні опції та користувач натиснув кнопку “OK”, в такому випадку з'явиться відповідне повідомлення, яке інформує користувача, що він повинен обрати 2 опції(рисунок 1.3)

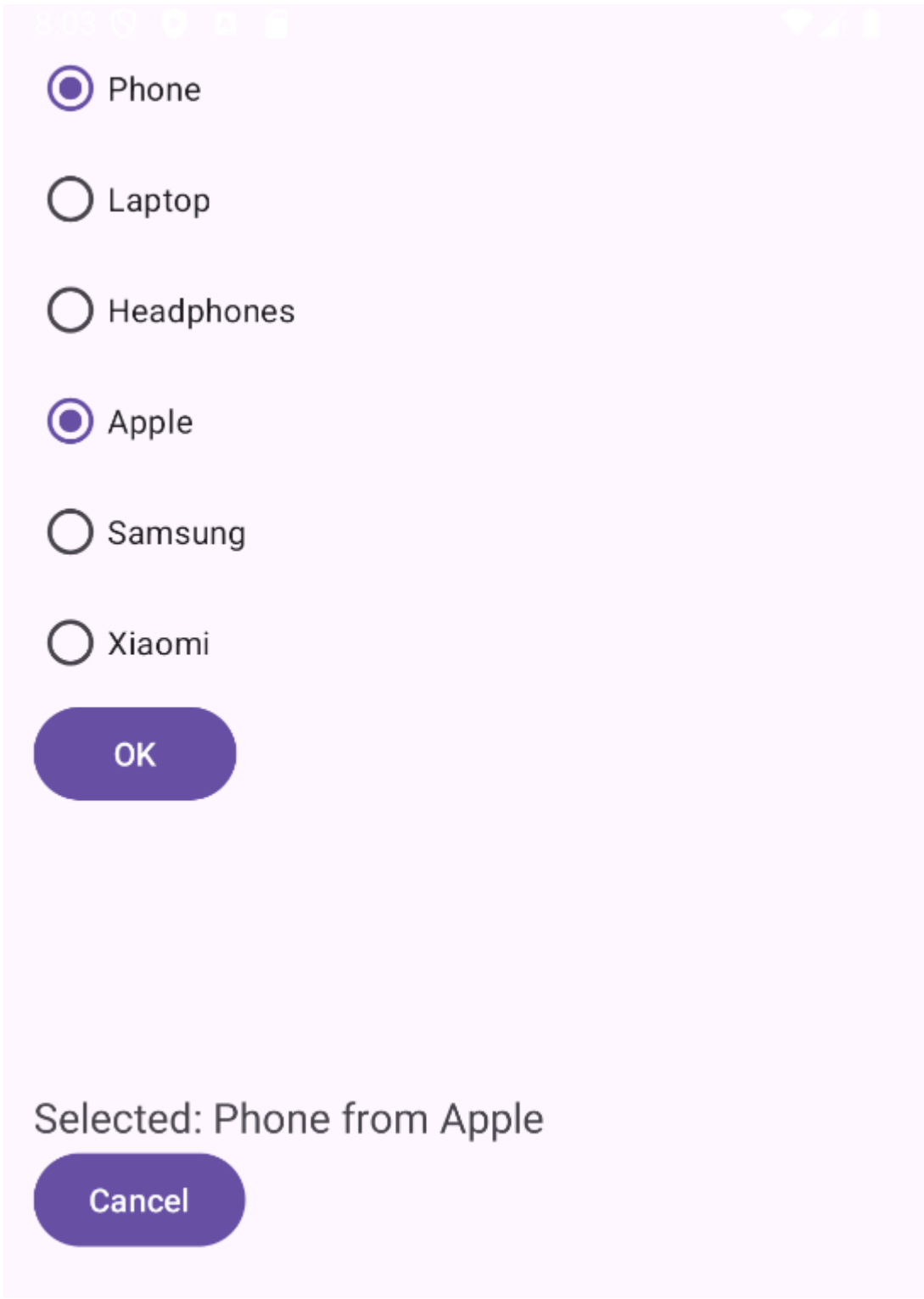


Рисунок 1.2 – Відображення обраних варіантів після натискання на кнопку “OK”

☐ Phone

☐ Laptop

☐ Headphones

☐ Apple

☒ Samsung

☐ Xiaomi

OK

Cancel


 Please select both a product type and a company.

Рисунок 1.3 – Відображення повідомлення про необхідність обрання двох опцій

КОНТРОЛЬНІ ПИТАННЯ

1. Призначення та можливості компоненту Фрагмент.

Фрагмент(клас `Fragment`) представляє поведінку або частину користувацького інтерфейсу в Діяльності (клас `Activity`): можна об'єднувати декілька фрагментів в одній Діяльності для побудови багато-панельного інтерфейсу та повторного використання фрагмента в декількох Діяльностях.

2. Життєвий цикл компонента Фрагмент.

Управління життєвим циклом фрагмента багато в чому аналогічно управлінню життєвим циклом Діяльності (фрагмент може існувати в одному з трьох станів): `resumed`, `paused`, `stopped`.

3. Способи створення компонента Фрагмент.

Для створення Фрагмента необхідно описати підклас класу `Fragment` (або його існуючого підкласу). Фрагмент може також бути створеним через розширення підкласів класу `Fragment`.

4. Опишіть способи управління компонентом Фрагмент.

Для управління фрагментами в Діяльності потрібен клас `FragmentManager`, який дозволяє: отримувати фрагменти, наявні в Діяльності, за допомогою методу `findFragmentById()` (для фрагментів, що надають користувацький інтерфейс в макетіДіяльності) або `findFragmentByTag()` (як для фрагментів, що мають користувацький інтерфейс, так і для фрагментів без нього); додавати транзакцію в стек переходів назад методом `addToBackStack()`; знімати фрагменти зі стеку переходів назад методом `popBackStack()` (імітуючи натискання кнопки Назад користувачем); реєструвати процес-слухач змін в стеці переходів назад за допомогою методу `addOnBackStackChangeListener()`.

5. Опишіть способи взаємодії між Фрагментами.

Один із найбільш класичних способів — використовувати інтерфейси для взаємодії між фрагментами через активність. Активність виступає як посередник між фрагментами. Для взаємодії між фрагментами можна використовувати спільний `ViewModel`, який допомагає зберігати та обмінюватися даними між фрагментами безпосередньо через активність. Це один із найпоширеніших підходів, особливо в сучасних програмах. Третій підхід для передачі даних між фрагментами через `Bundle`. Під час створення нового фрагмента можна передати дані через аргументи, використовуючи метод `setArguments()`. Також можна використовувати механізм `FragmentManager.ResultListener` для передачі результатів між фрагментами. Цей підхід не потребує активності як посередника, і фрагменти можуть обмінюватися результатами напряму.

6. Наведіть поняття системи, малої системи та мобільної платформи.

Система — це сукупність взаємопов'язаних елементів, які функціонують разом для досягнення певної мети. Ключовими характеристиками будь-якої системи є: компоненти, зв'язки та функціональність. Мала система — це система, що має обмежені ресурси, можливості або розміри, проте здатна виконувати конкретні завдання. Вона зазвичай виконує вузько спеціалізовані функції та використовує мінімум ресурсів. Мобільна платформа — це операційна система та програмне середовище, призначені для роботи на мобільних пристроях, таких як смартфони, планшети або смарт-годинники. Мобільні платформи забезпечують користувачів інтерфейсом для взаємодії з пристроєм і доступом до додатків, а розробникам надають середовище для створення цих додатків.

7. Опишіть типи мобільних застосунків.

Нативні застосунки створюються спеціально для певної операційної системи. Вони розробляються з використанням мов програмування, що підтримуються конкретною платформою. Гібридні застосунки є комбінацією нативних і веб-застосунків. Вони розробляються за допомогою веб-технологій, але запускаються у спеціальному контейнері, що дозволяє їм працювати як нативні застосунки на різних платформах. Кросплатформені застосунки пишуться один раз і працюють на різних платформах без значних змін.

8. Наведіть класифікацію та загальну характеристику середовищ розробки мобільних застосунків.

Нативні - Ці середовища використовуються для розробки мобільних застосунків під конкретну платформу, забезпечуючи повний доступ до функцій і API операційної системи. Кросплатформенні середовища дозволяють створювати застосунки одночасно для кількох платформ, використовуючи один кодовий базис.

9. Наведіть класифікацію та загальну характеристику мобільних платформ.

За операційною системою, за типом пристроїв, за архітектурою процесора

ВИСНОВКИ

У ході виконання лабораторної роботи було розроблено програму, яка складається з двох фрагментів, що взаємодіють між собою в рамках однієї діяльності (Activity). Перший фрагмент слугує для введення даних користувачем, зокрема вибору параметрів через радіокнопки та кнопки підтвердження «ОК». Другий фрагмент відображає результат вибору у вигляді текстового поля та надає можливість користувачу скасувати або змінити свій вибір за допомогою кнопки «Cancel». Програма продемонструвала успішну роботу з фрагментами в межах однієї Activity, надаючи зручний інтерфейс для користувача з можливістю динамічно змінювати вміст залежно від взаємодії. Використання фрагментів дозволило забезпечити гнучкість і масштабованість інтерфейсу.