

Сучасні технології мобільного програмування

Управління станом у Compose

Слайди до лекцій

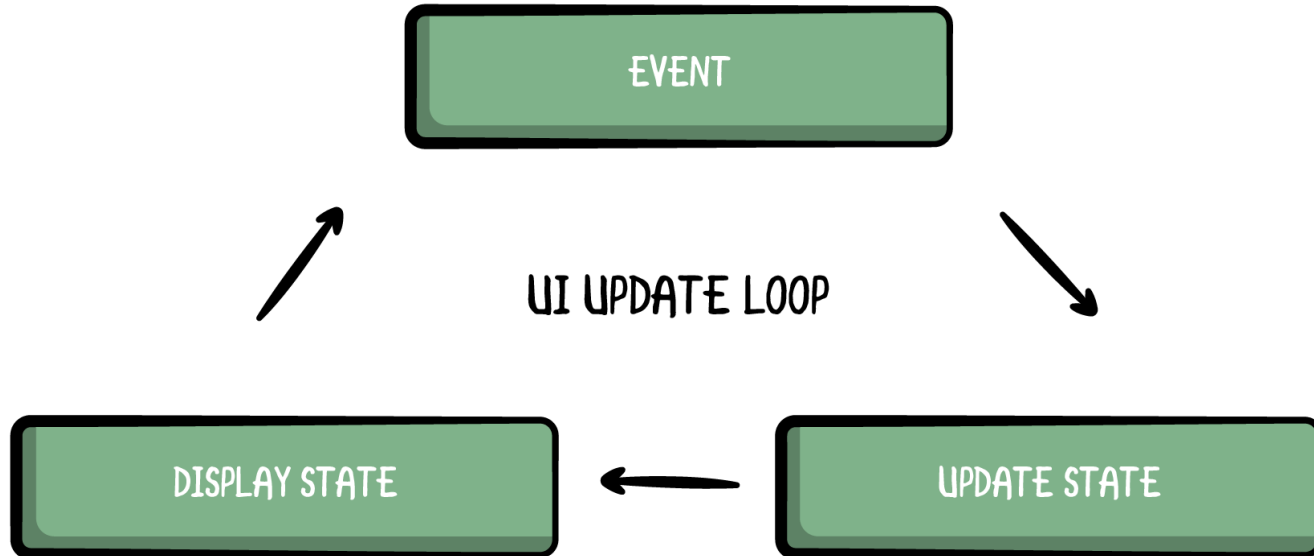
Поняття стану

- ▶ За своєю суттю кожна програма працює з певними властивостями об'єктів, які (властивості) можуть змінювати своє значення. Наприклад, можуть додаватися об'єкти, з якими працює застосунок, заповнюючи дані об'єктів, можуть читатися та змінюватися дані таких об'єктів, видалятися певні дані об'єктів або самі об'єкти тощо.
- ▶ **Стан** – будь-яка властивість об'єкту, з яким працює застосунок, що може змінюватися з часом. Ці значення можуть включати будь-що - від запису у базі даних до властивості класу.
- ▶ При зміні значення хоча б однієї з властивостей об'єкту говорять про **зміну стану**.
- ▶ При зміні стану необхідно оновлювати інтерфейс користувача, щоб відобразити ці зміни.



Взаємодія користувача з UI

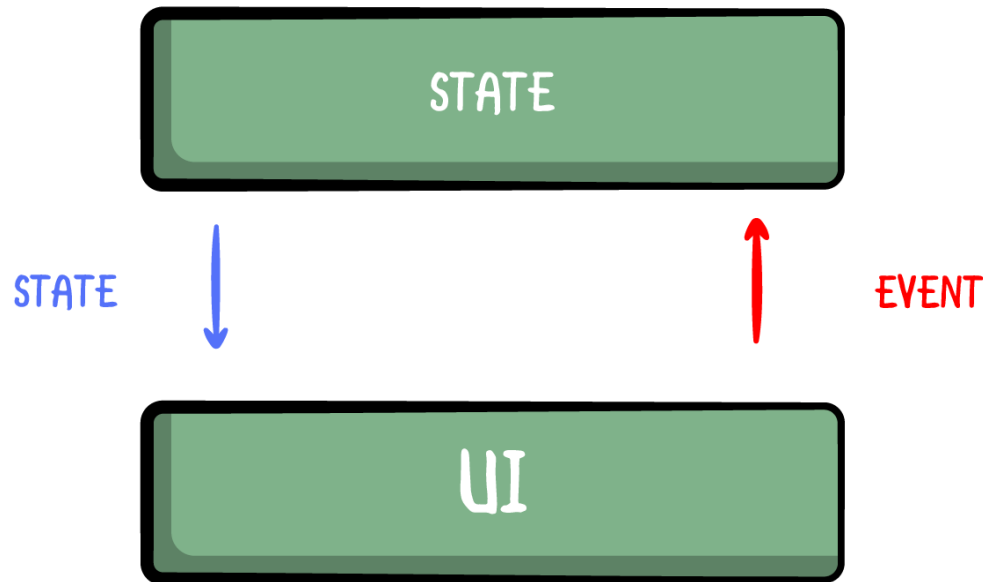
- ▶ Користувачі спілкуються із Android застосунками за допомогою **подій** (клацання, перетягування тощо), а програма відповідає, відображаючи поточний **стан**, який змінюється як реакція на події відповідно до **логіки програми** (реалізованої у обробнику події).



Обробка стану з односпрямованим потоком даних

- ▶ **Односпрямований потік даних** — це концепція, де як зміни стану, так і оновлення інтерфейсу користувача мають лише один напрямок: що події зміни стану можуть надходити лише з одного джерела, як правило, від взаємодії користувача, а оновлення інтерфейсу користувача можуть надходити лише від менеджера стану, обробника подій або моделі, незалежно від того, як ви хочете посилатися на це.
 - ▶ Односпрямований потік даних — не нова концепція в програмуванні. Загальновідомо, що доцільно відокремити компоненти, які відображають стан в інтерфейсі користувача, від частин програми, які зберігають і змінюють стан.
 - ▶ Compose створено з урахуванням односпрямованого потоку даних.
-

Обробка стану з односпрямованим потоком даних

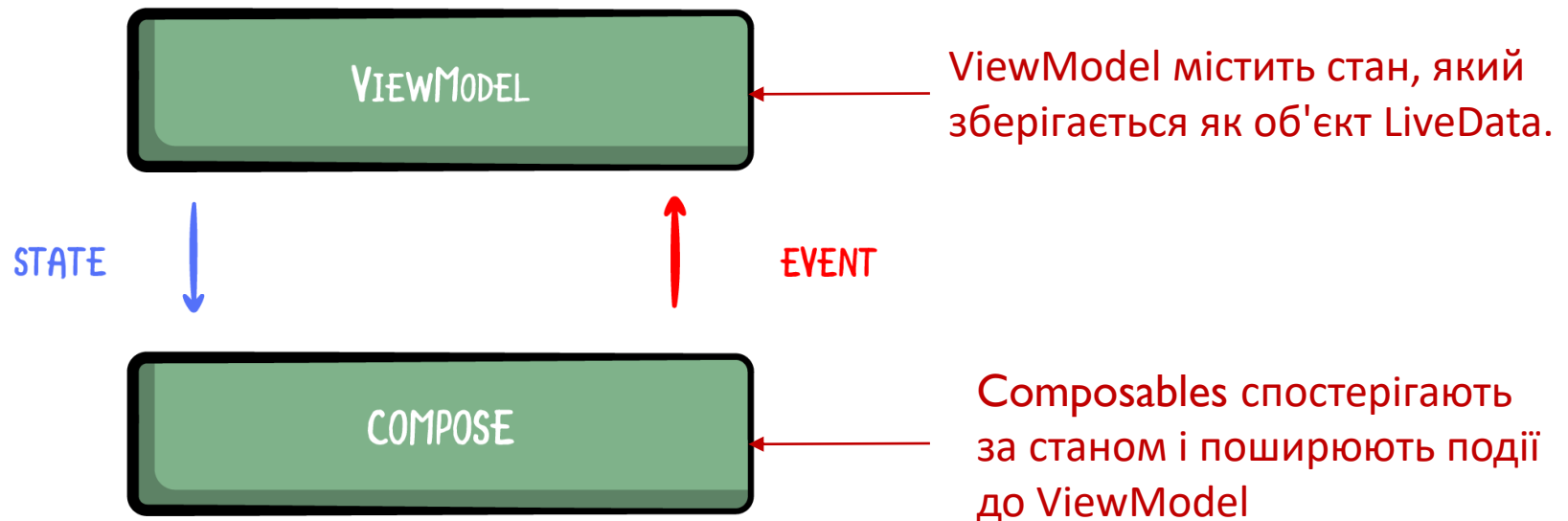


- ▶ Ключова концепція тут полягає в тому, що **стан тече вниз**, а **події течуть вгору**, як показано на зображенні вище.
- ▶ Інша ключова концепція полягає в тому, що **UI спостерігає за станом**. Щоразу, коли з'являється новий стан, інтерфейс користувача відображає його (у Compose - автоматично).

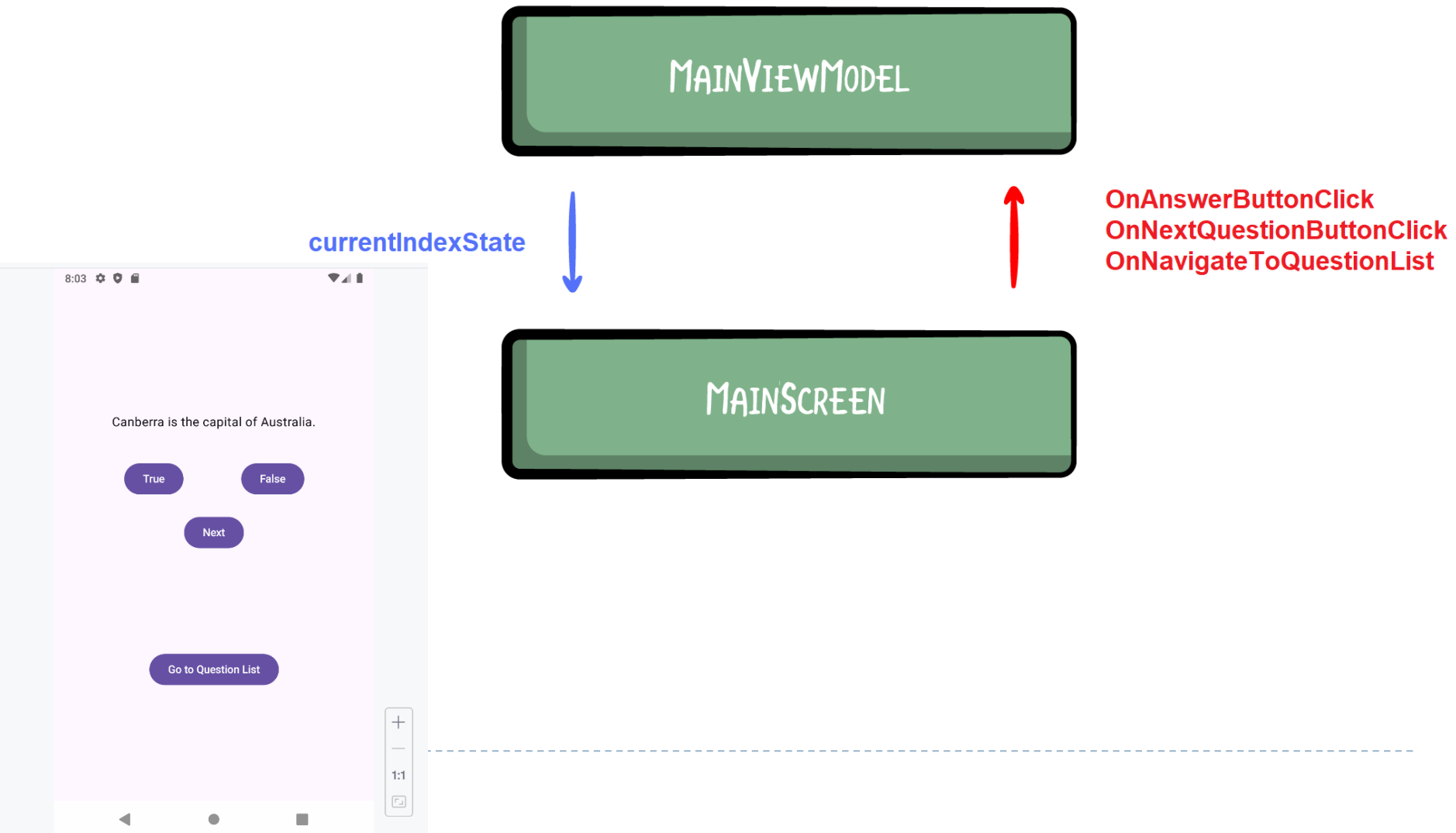


Compose & ViewModel & LiveData

- ▶ Платформа Android пропонує компоненти архітектури Android, які дозволяють організувати односпрямований потік даних та спостереження UI за станом - **ViewModel** і **LiveData**.
- ▶ **ViewModel** дозволяє "витягувати" стан з інтерфейсу користувача та визначати події, які інтерфейс користувача може викликати для оновлення цього стану.
- ▶ **LiveData** дозволяє створювати утримувачі (холдери) стану, які надають можливість будь-кому спостерігати за його змінами.



Обработка стану з односпрямованим потоком даних - MainScreen & MainViewModel



Обработка стану з односпрямованим потоком даних

```
class QuestionListActivity : ComponentActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        ...  
        val questionRepository = QuestionRepository  
        val viewModel = MainViewModel(questionRepository)  
        setContent {  
            ...  
            val questionList by viewModel.questionBank.observeAsState(emptyList())  
            QuestionList(  
                questionList = questionList,  
                onClick = viewModel::onClick,  
                // viewModel = MainViewModel(questionRepository),  
                modifier = Modifier.padding(innerPadding)  
            )  
        }  
    }  
}
```

Стан передається вниз

Подія передається вгору

