

# Compose Multiplatform iOS and Android

# План доклада

- 1 Какие проблемы решает Compose Multiplatform
- 2 Настройка окружения для работы с Compose Multiplatform
- 3 Поверстать реальные примеры и на iOS и на Android
- 4 Что сейчас можно писать, а что еще пока недоступно



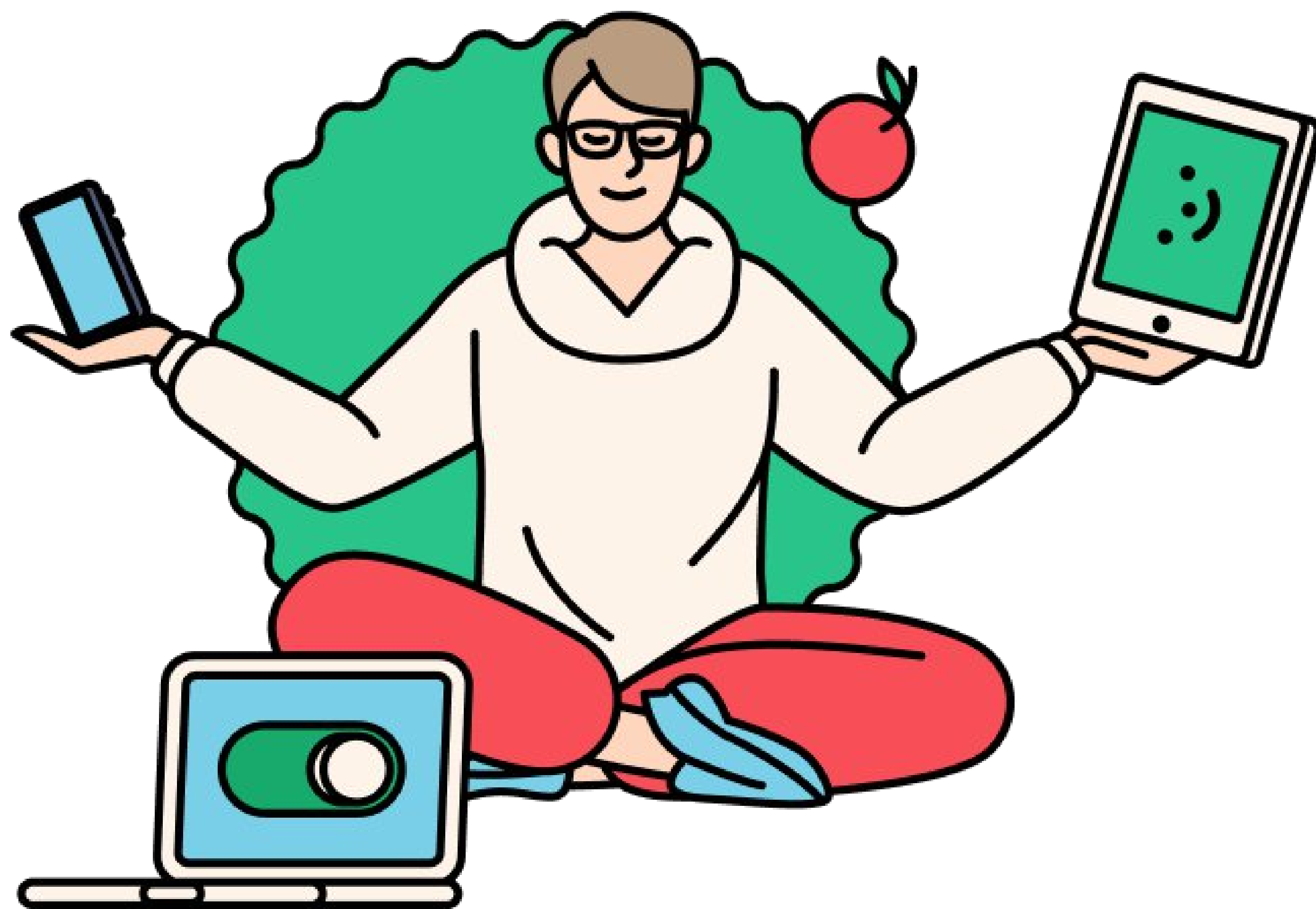
# Какие проблемы решает Compose Multiplatform





Бизнес хочет **быстро**  
запускаться, **быстро**  
проводить  
эксперименты





# Все мы пишем код с багами

(что уж скрывать)





# Правка багов на одной стороне

так как верстка  
сразу на iOS & Android



# В итоге

сокращается time-to-market  
Бизнес доволен, все счастливы

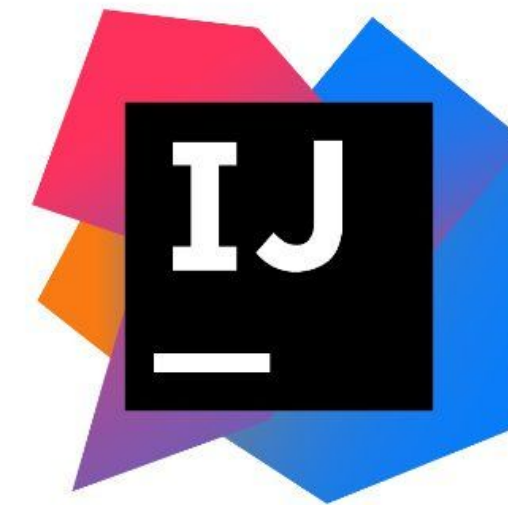




# Настройка окружения

для работы с Compose Multiplatform

# Варианты настройки



# Плюсы и Минусы



## Плюсы

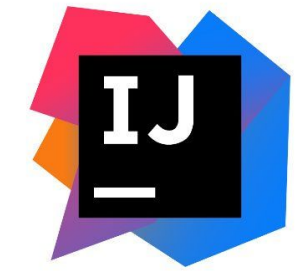
- Сразу создается iOS и Android.

## Минусы

- Нет setup'а для Compose.



# Плюсы и Минусы



## Плюсы

- Есть сразу setup'а для Compose.

## Минусы

- Создается Android + Desktop. iOS таргет придется создавать самому.
- Нет поддержки CocoaPods.
- Гораздо больше настроек, чтобы запуститься, чем в 1-ом способе.

# Как Compose рисует вьюшки и на Android и на iOS?



# Как Compose рисует вьюшки и на Android и на iOS?

**Skia**



Stackoverflow



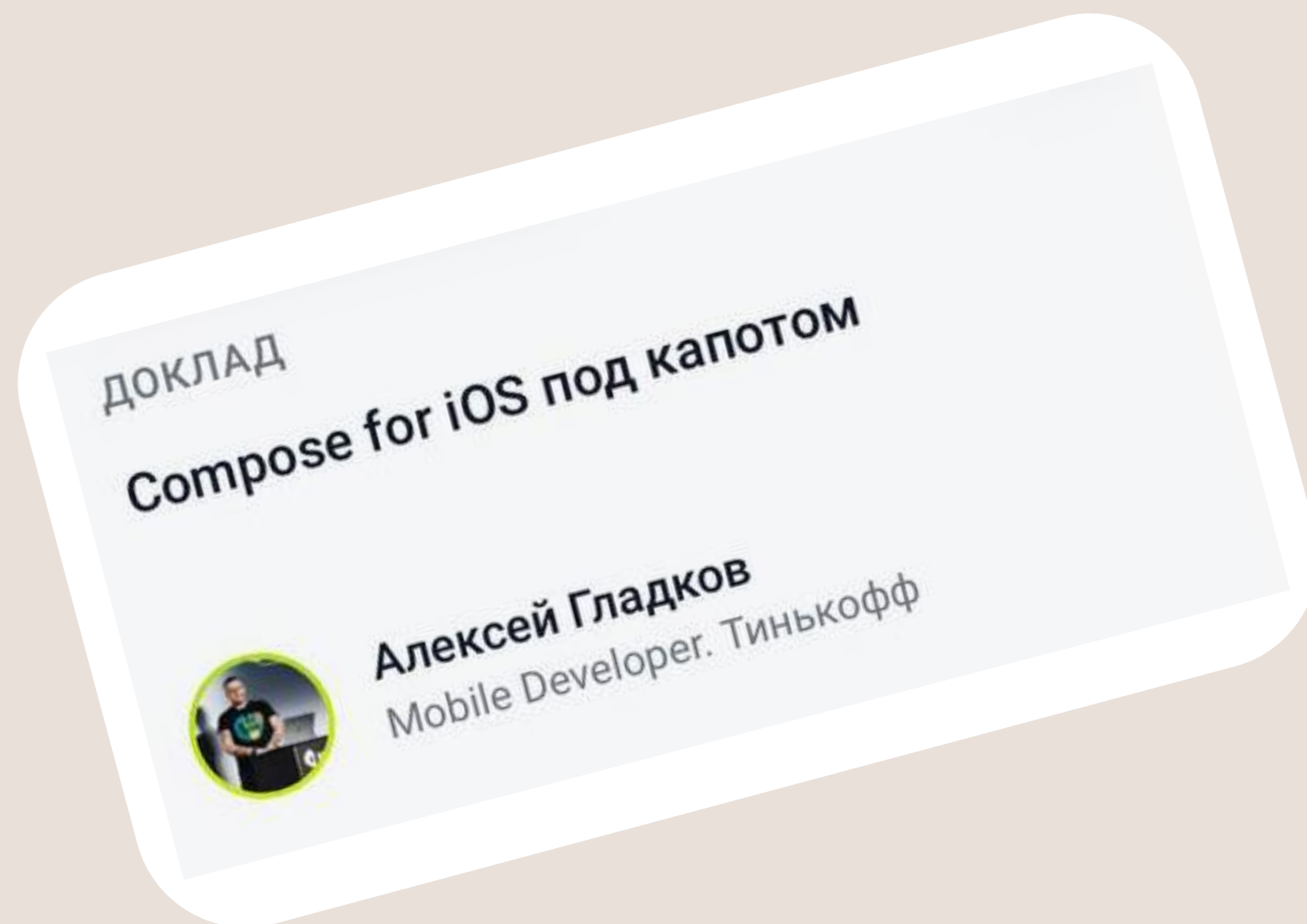
# В итоге

все сводиться к такой функции

```
@Composable inline fun Layout(  
    content: @Composable () -> Unit,  
    modifier: Modifier = Modifier,  
    measure Policy: MeasurePolicy  
) {  
    val density = LocalDensity.current  
    val layoutDirection = LocaleLayoutDirection.current  
    val view Configuration = LocalViewConfiguration.current  
    Reusable ComposeNode<ComposeUiNode, Applied<Any>>(  
        factory = Compose UiNode.Constructor,  
        update = { this: Update<ComposeUiNode>  
            set(measurePolicy, ComposeUiNode. SetMeasurePolicy)  
            set(density, ComposeUiNode.SetDensity)  
            set(layoutDirection, ComposeUiNode. SetLayoutDirection)  
            set(viewConfiguration, ComposeUiNode. SetViewConfiguration)  
        }  
        skippableUpdate = materializerOf(modifier),  
        content = content  
    )  
}
```

# В итоге

все сводиться к такой функции



```
@Composable inline fun Layout(  
    content: @Composable () -> Unit,  
    modifier: Modifier = Modifier,  
    measure Policy: MeasurePolicy  
) {  
    val density = LocalDensity.current  
    val layoutDirection = LocaleLayoutDirection.current  
    val view Configuration = LocalViewConfiguration.current  
    Reusable ComposeNode<ComposerUiNode, Applied<Any>>(  
        factory = Compose UiNode.Constructor,  
        update = { this: Update<ComposeUiNode>  
            set(measurePolicy, ComposeUiNode. SetMeasurePolicy)  
            set(density, ComposeUiNode.SetDensity)  
            set(layoutDirection, ComposeUiNode. SetLayoutDirection)  
            set(viewConfiguration, ComposeUiNode. SetViewConfiguration)  
        }  
        skippableUpdate = materializerOf(modifier),  
        content = content  
    )  
}
```





# Реальные примеры

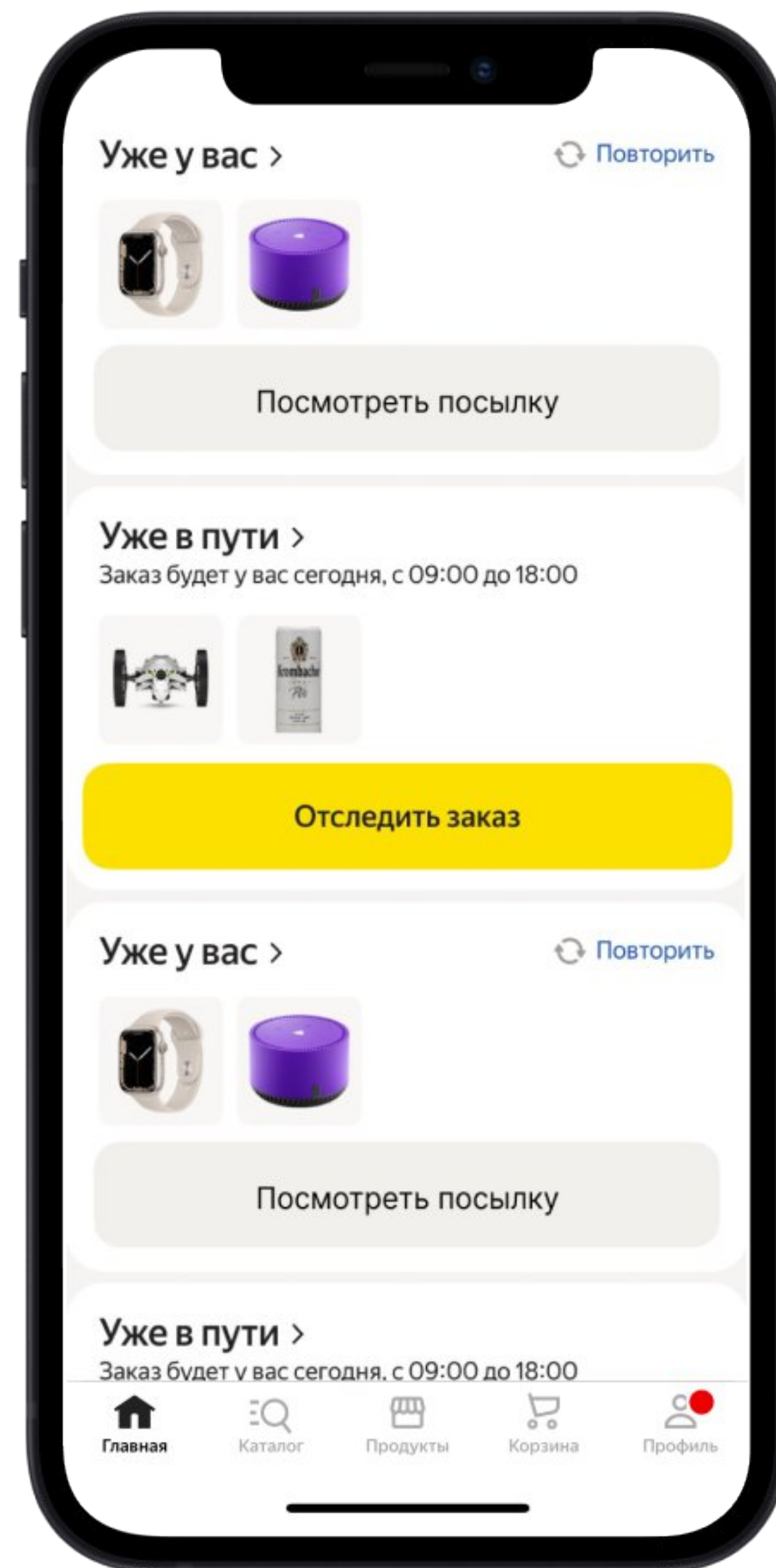


Настройка  
окружения —  
Android Studio

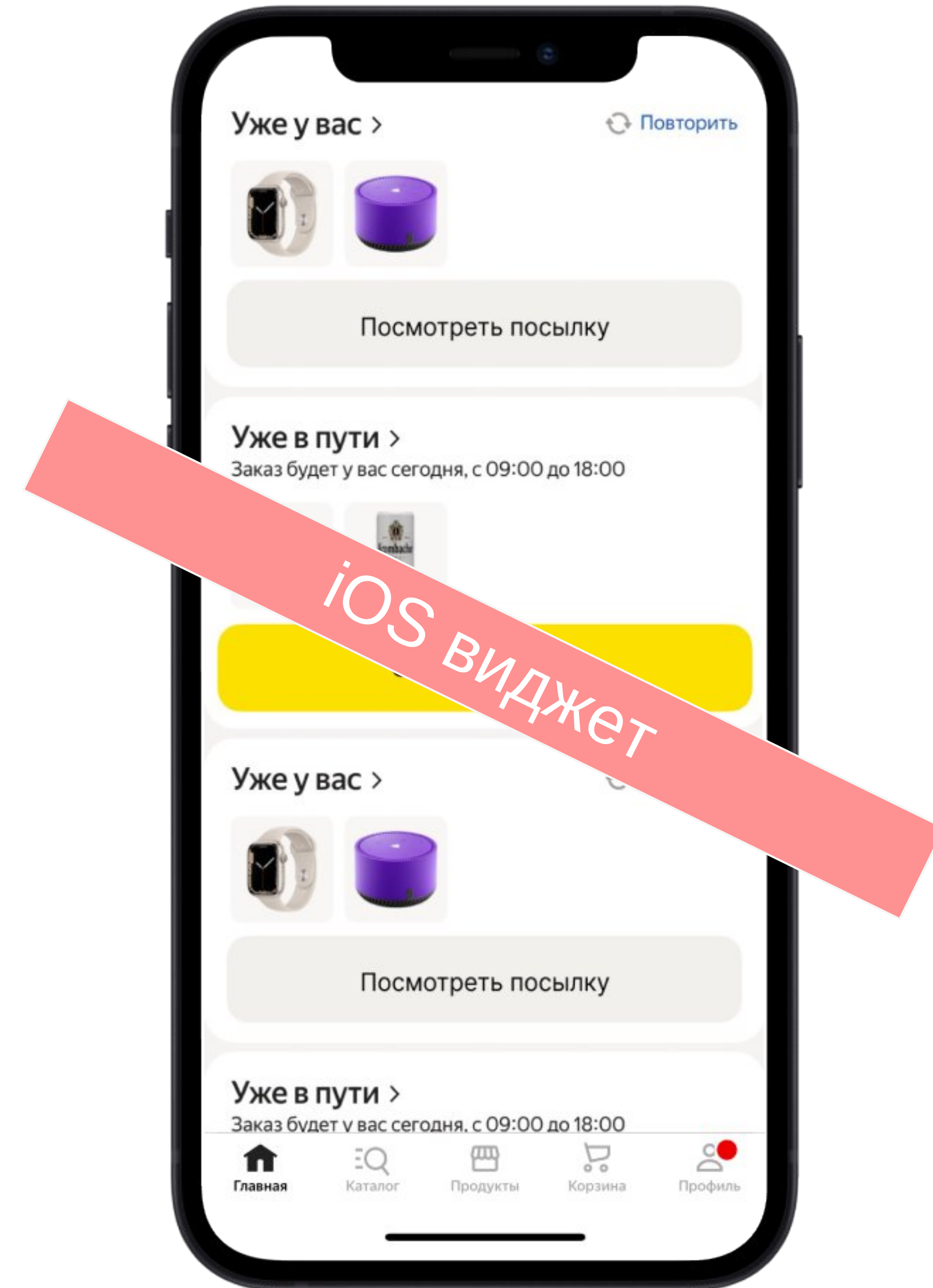


Настройка  
окружения —  
IntelliJ Idea

# Реальные примеры



# UIKitView

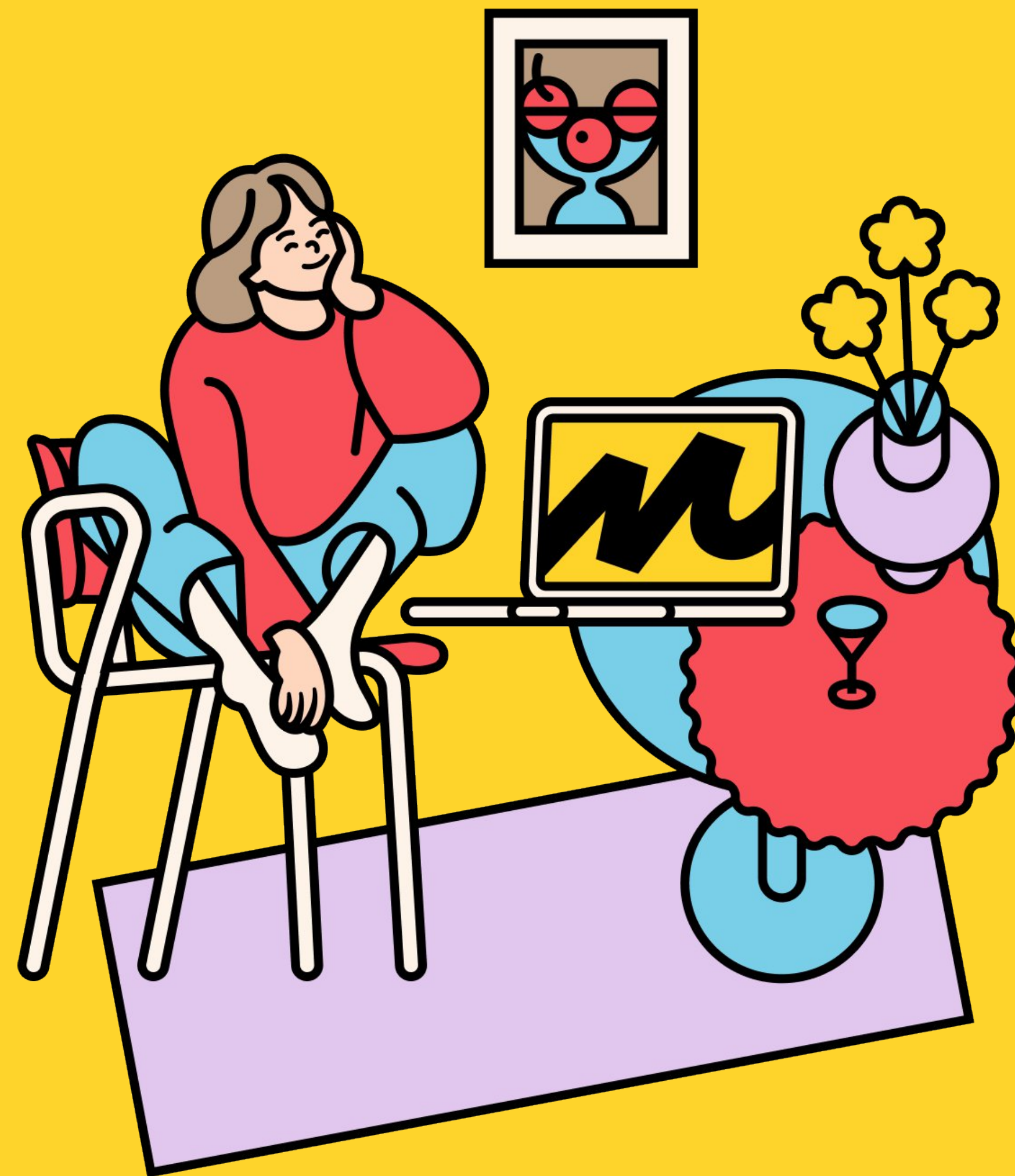




# Пример приложения на iOS и Android



Примеры  
с презентации



# Что можно? А что пока недоступно?

## Можно

- Все что связано с Android и есть уже сейчас.
- Базовый UI писать и для iOS.
- Нативные iOS компоненты можно встроить в Compose.

## Недоступно

- Тестирование ui.
- Preview есть только для Android and Desktop.
- Нельзя вызывать обращение к системе (камера).
- Выпадающие меню, диалоги.

# Выводы

1

Какие проблемы решает  
Compose Multiplatform

2

Настройка окружения  
для работы с Compose  
Multiplatform

3

Поверстать  
реальные примеры  
и на iOS и на Android

4

Что сейчас можно  
писать, а что еще  
пока недоступно

# Куда дальше?



KotlinKonf 23



Примеры от JB



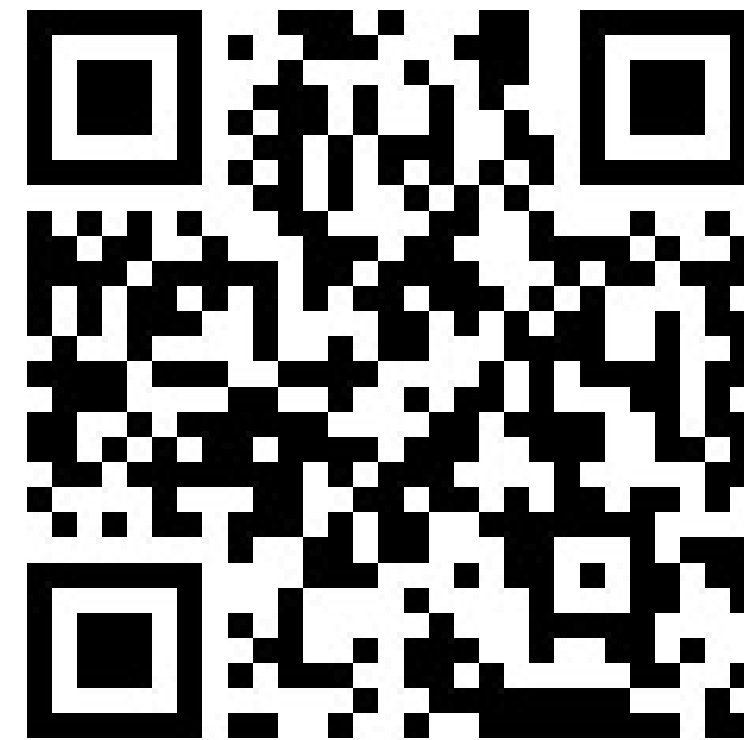
Compose сайт

# Q&A

**Аникин Кирилл**

Андроид Разработчик, Яндекс Маркет

 @ANIKINKIRILL



Мой канал



Примеры  
с презентации