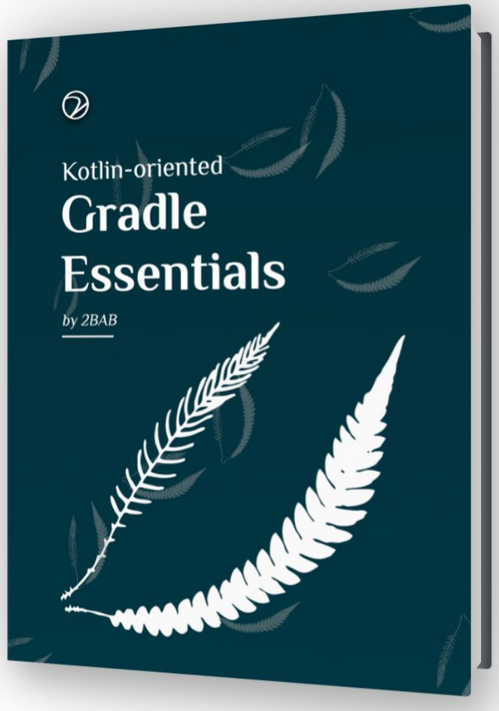
Google I/O Extended Guangzhou

On-device Model 在 KMP 的集成与用例

El Zhang @2BAB / Android GDE / Singapore

Google I/O Extended

《Android 构建与架构实战》



《面向 Kotlin 的 Gradle 基础指南》

2bab.me/zh/book/

Summary

Table of Content

Google I/O Extended

Edge Computing

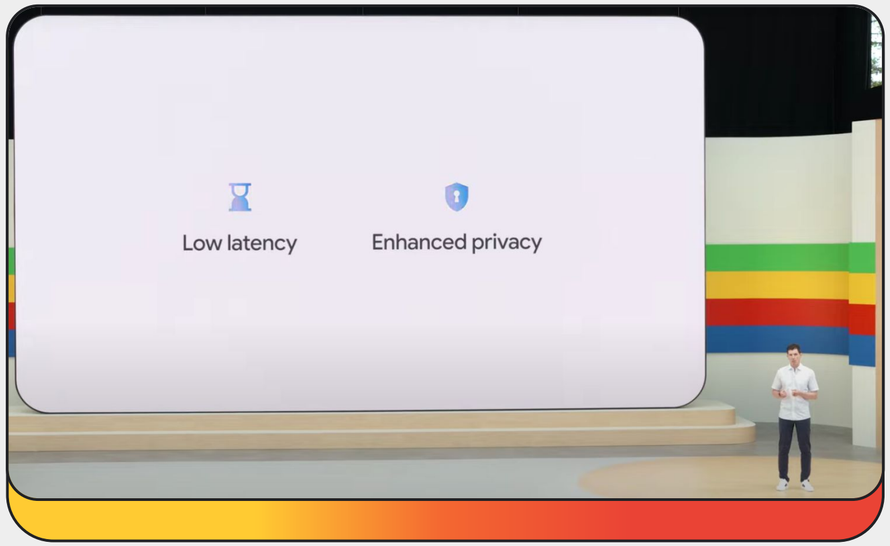
On-device Model

Low latency responses and **enhanced data privacy**.

Edge Computing

Gemini Nano

Google's **most efficient model** for on-device tasks. It **runs directly on mobile**.

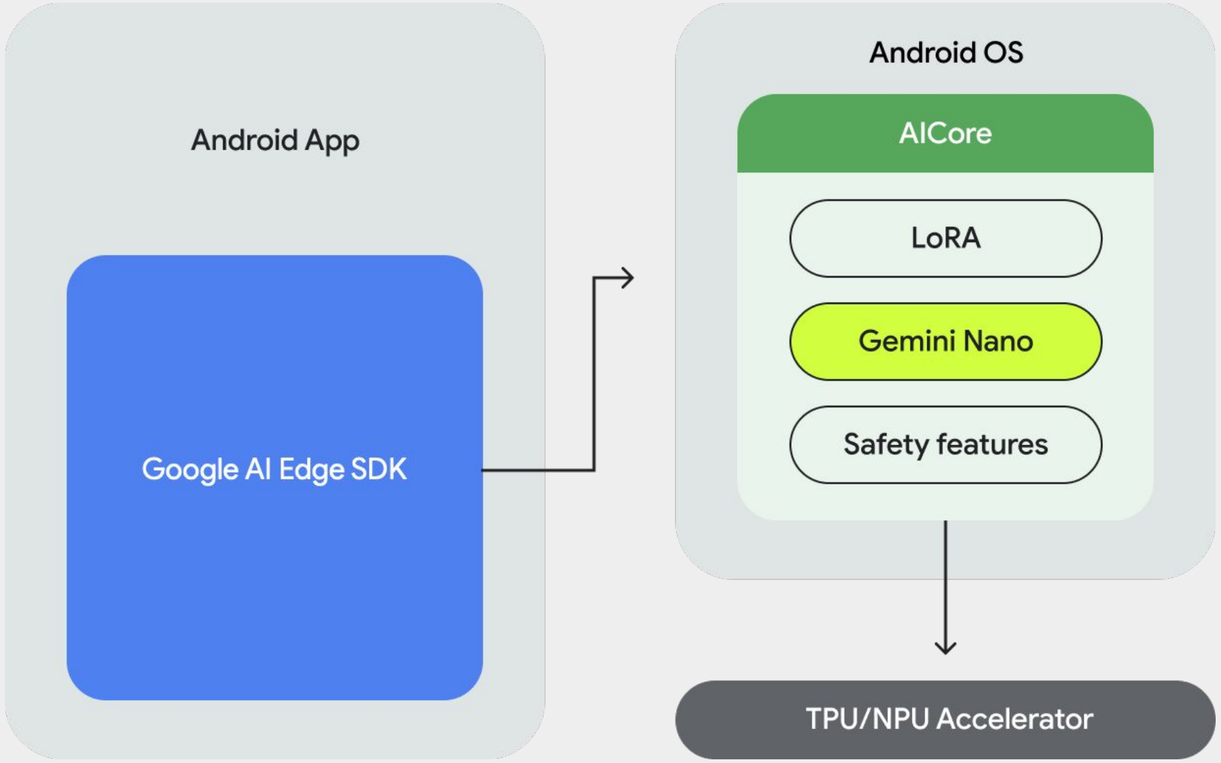


Edge Computing

Gemini Nano

You can **fine-tune** Gemini Nano with **LoRA(Low-Rank Adaptation)** to perform specialized tasks as well as its larger counterparts. Gemini Nano runs in **Android's AICore system service**.

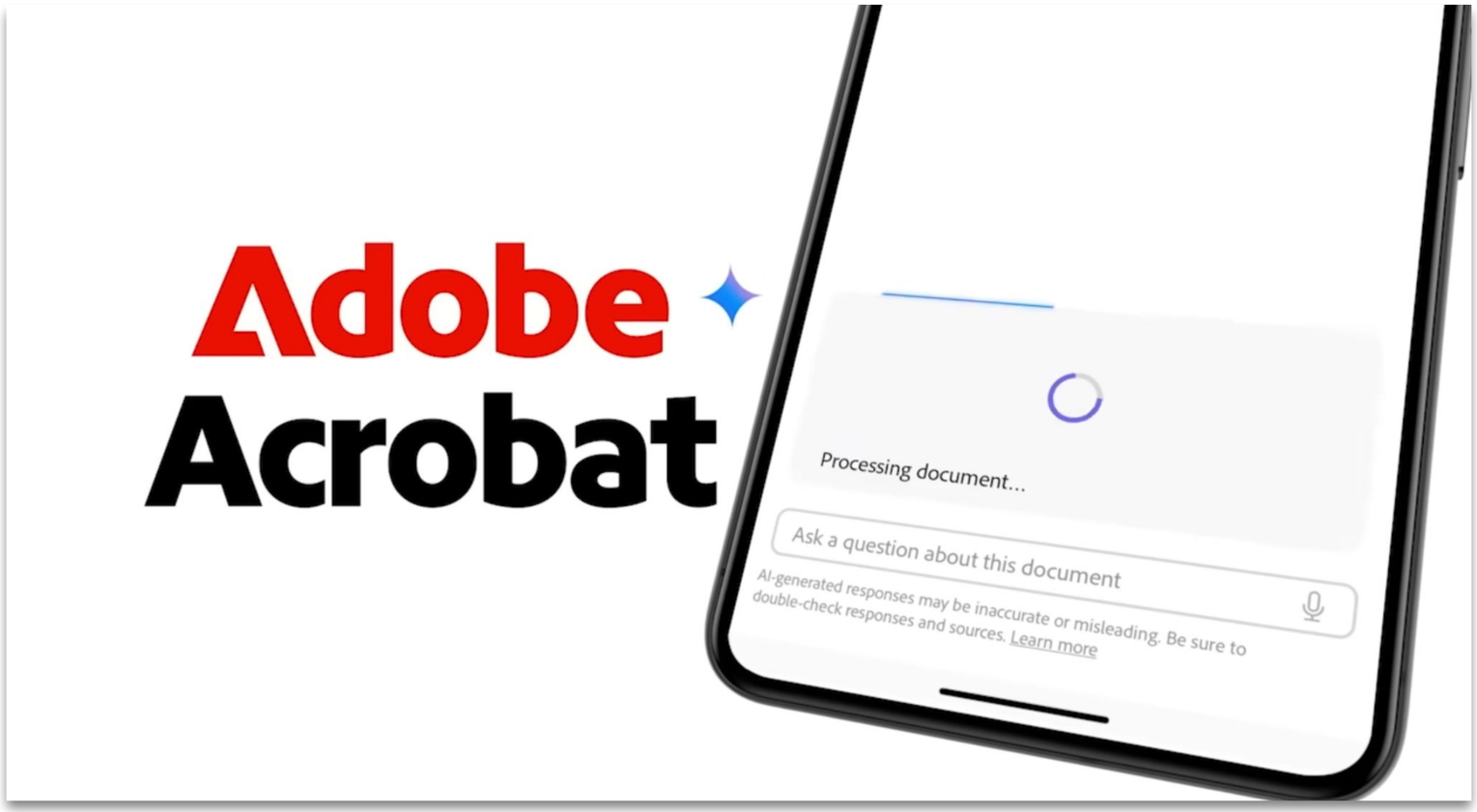
Gemini Nano is currently **only available** on Google Pixel 8 Pro, Pixel 8, Pixel 8a, and Samsung S24 Series devices, with support for more devices coming soon.



Edge Computing

Gemini Nano

Only few partners that were cooperated with Google in this early stage.



Edge Computing

MediaPipe

MediaPipe Solutions provides a suite of libraries and tools for you to **quickly** apply **artificial intelligence (AI)** and **machine learning (ML)** techniques in your applications.

https://mediapipe-studio.webapps.google.com/home

Solution	Android	Web	Python	iOS	Customize model
LLM Inference API	●	●		●	●
Object detection	●	●	●	●	●
Image classification	●	●	●	●	●
Image segmentation	●	●	●		
Interactive segmentation	●	●	●		
Hand landmark detection	●	●	●	●	
Gesture recognition	●	●	●	●	●
Image embedding	●	●	●		
Face detection	●	●	●	●	
Face landmark detection	●	●	●		
Face stylization	●	●	●		●
Pose landmark detection	●	●	●		
Image generation	●				●
Text classification	●	●	●	●	●
Text embedding	●	●	●		
Language detector	●	●	●		
Audio classification	●	●	●		

Edge Computing

MediaPipe with Gemma

The LLM Inference API lets you run large language models (LLMs) completely on-device.

1. It supports **Gemma 2B** and 7B.
2. **Gemma** is a **lightweight**, **open** models built from the **same research** and **technology** used to create the **Gemini** models.
3. It also supports the following external models: Phi-2, Falcon-RW-1B and StableLM-3B.

Edge Computing

Gemma (1.1) 2B vs Gemini Nano (1.8B, 3.5B)

1. Gemma 2B: 开源，通用性强，并为“使用 CPU 执行”而优化，目前尚未支持在终端设备上直接微调，通常是优化和微调模型后，下载一个完整的新模型到终端/App。
2. Gemini Nano：非开源，需要特定芯片加速（例如 Pixel 8Pro 的 Tensor G3），有更强大的表现，目前已有终端设备上直接微调的 LoRA 支持。
3. 都是 On-device Model，二者技术同源。

Google I/O Extended

Mobile

Kotlin Multiplatform (KMP)

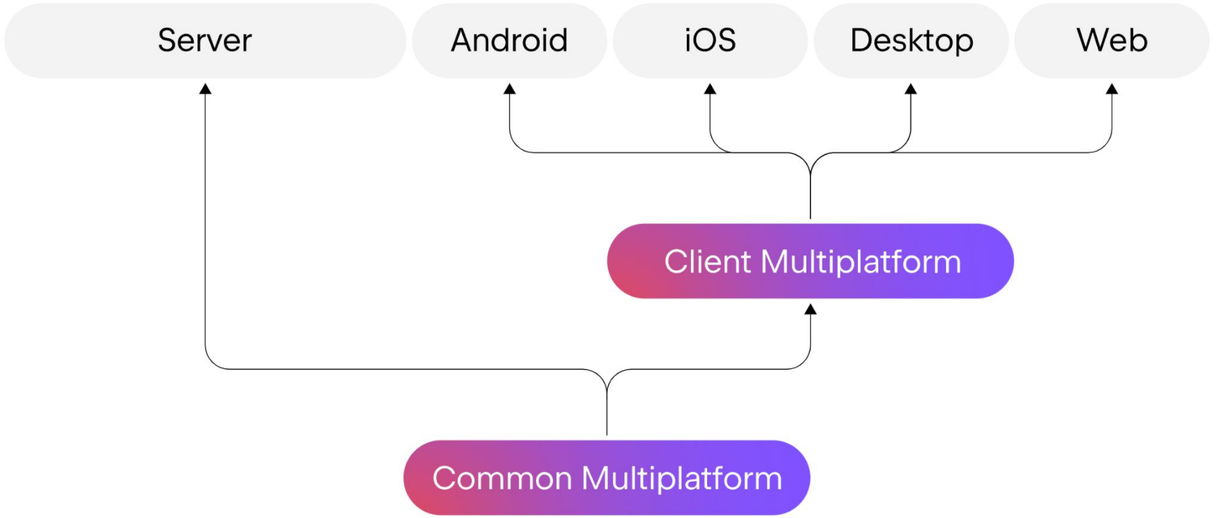
Sharing your code up to 99%.

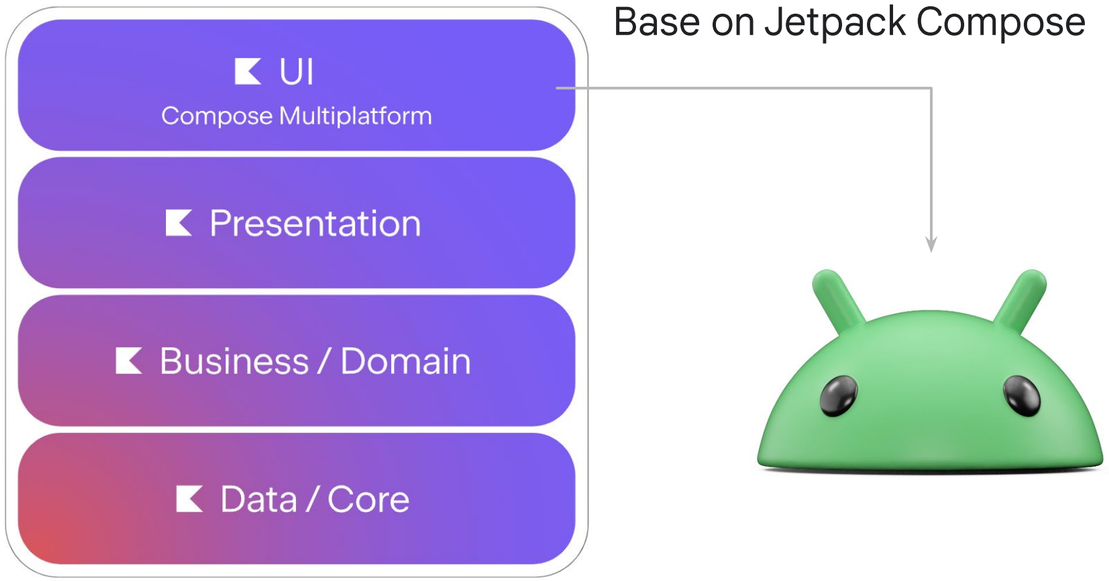
Mobile

Kotlin Multiplatform

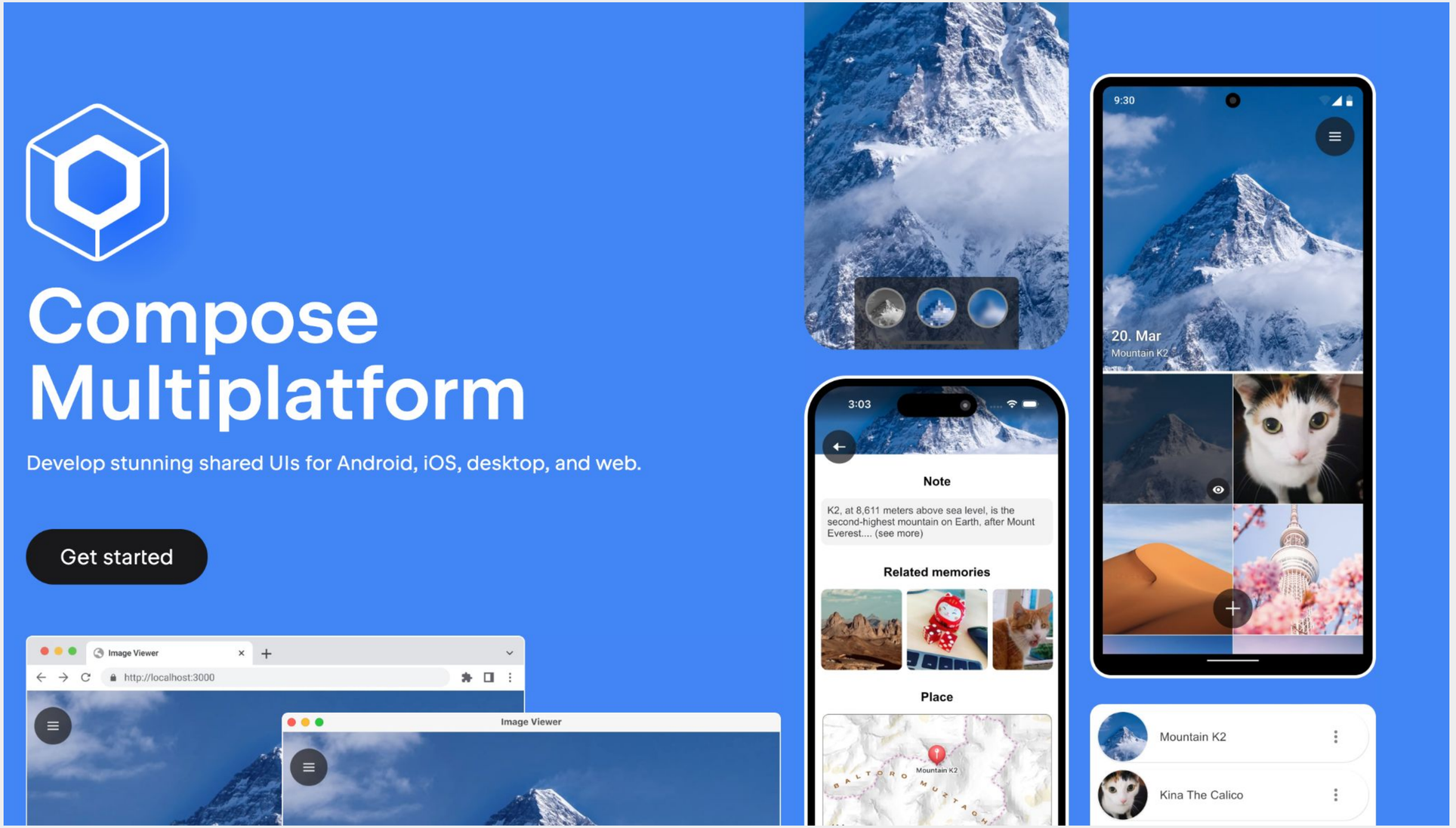
It was introduced in **2017**, enables developers to write **cross-platform** code using Kotlin, sharing **business logic** and **UI**(optional) across

- Android (with Kotlin to JVM)
- **iOS** (with Kotlin to Native)
- **Web** (with Kotlin to JS and WASM),
- **Desktop** including Windows/Linux/macOS (with Kotlin to Native and JVM)





Mobile


Compose Multiplatform
Develop stunning shared UIs for Android, iOS, desktop, and web.
Get started
Image Viewer
http://localhost:3000
Image Viewer
3:03
Note
K2, at 8,611 meters above sea level, is the second-highest mountain on Earth, after Mount Everest.... (see more)
Related memories
Place
Mountain K2
9:30
20. Mar
Mountain K2
Mountain K2
Kina The Calico

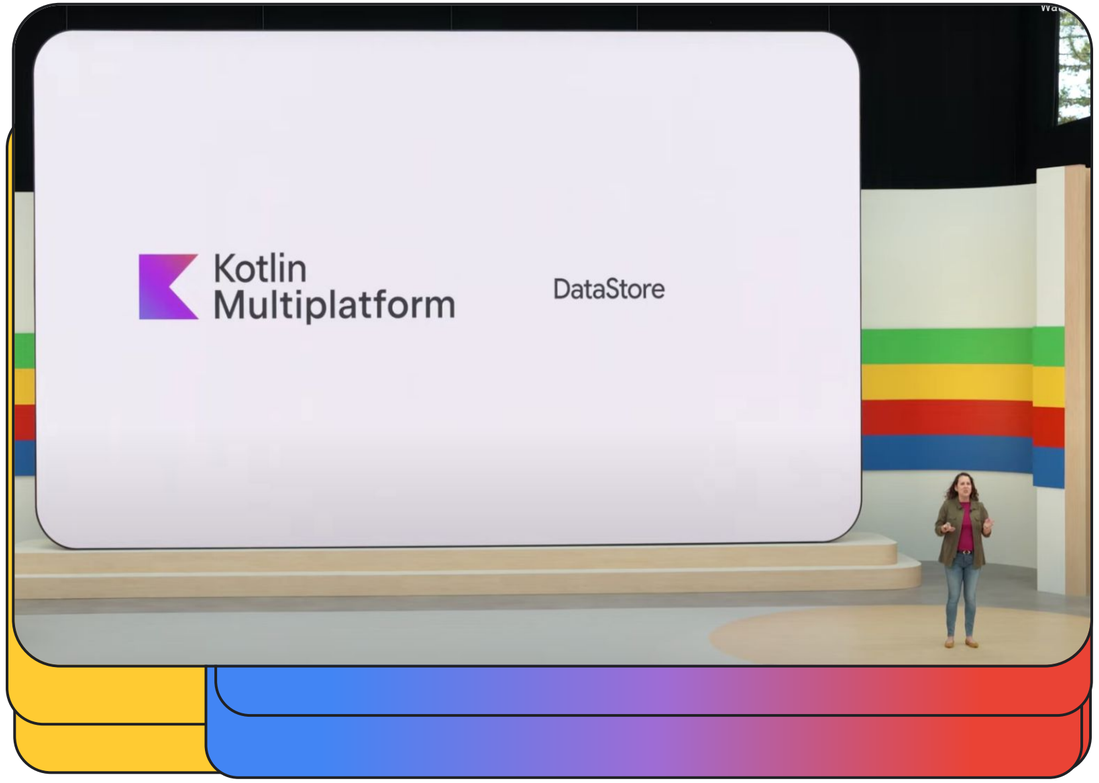
Mobile

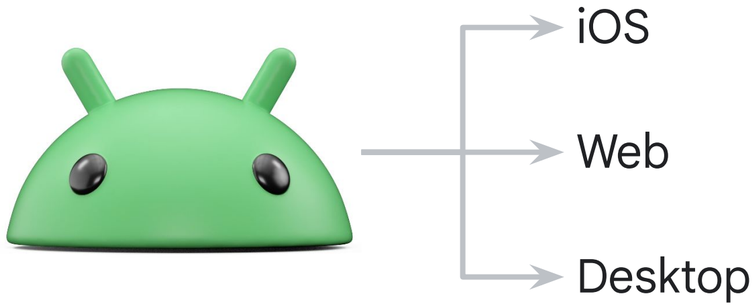
Kotlin Multiplatform on Android

First-class tooling and **library support** for Kotlin Multiplatform on Android.

Now expanded to your favourite libraries like Room / DataStore.

Leveraging KMP for business logic across Android, iOS and web. Strongly **enhancing** the **ecosystem** of KMP.

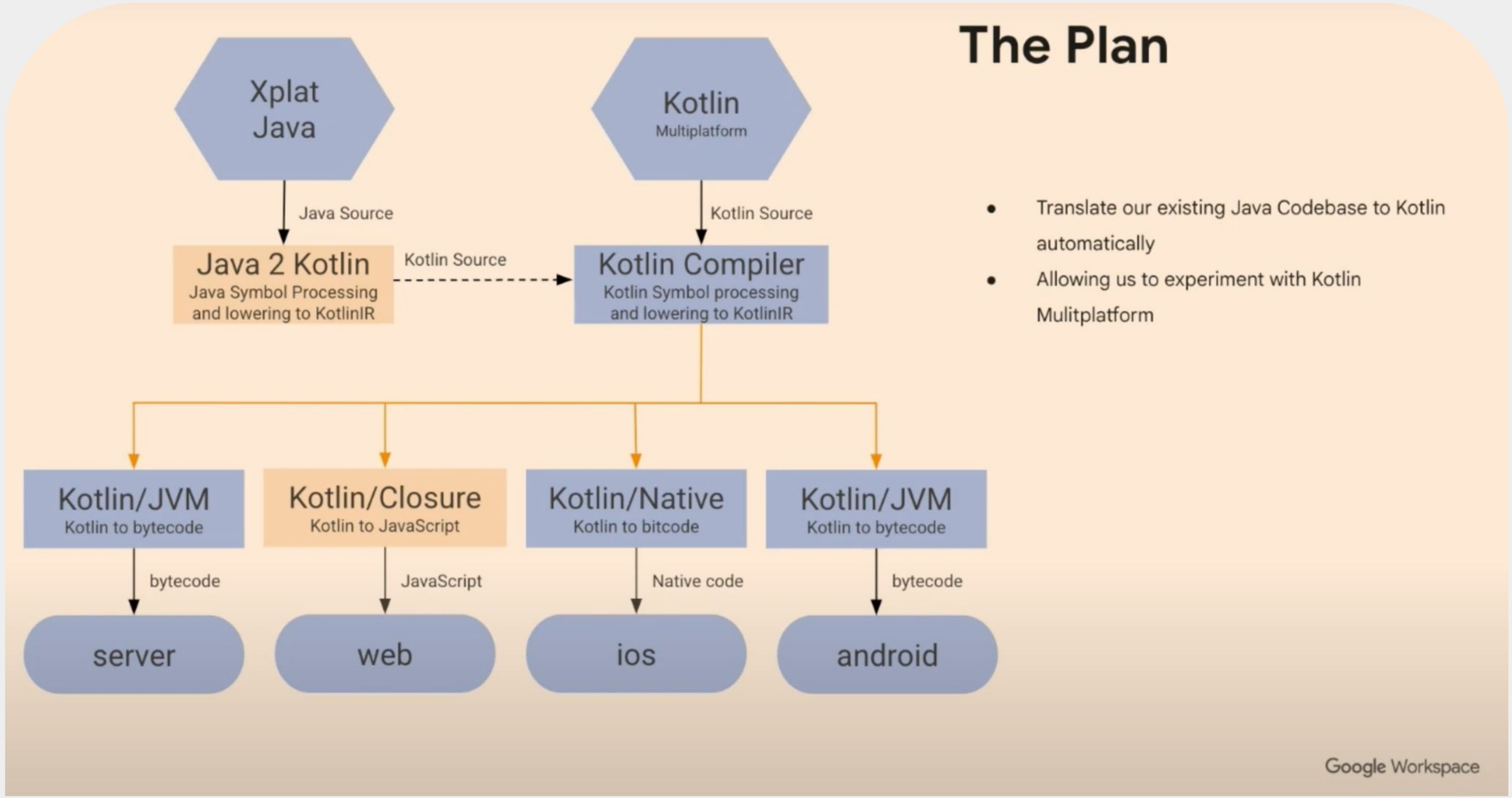




Mobile

Case Studies

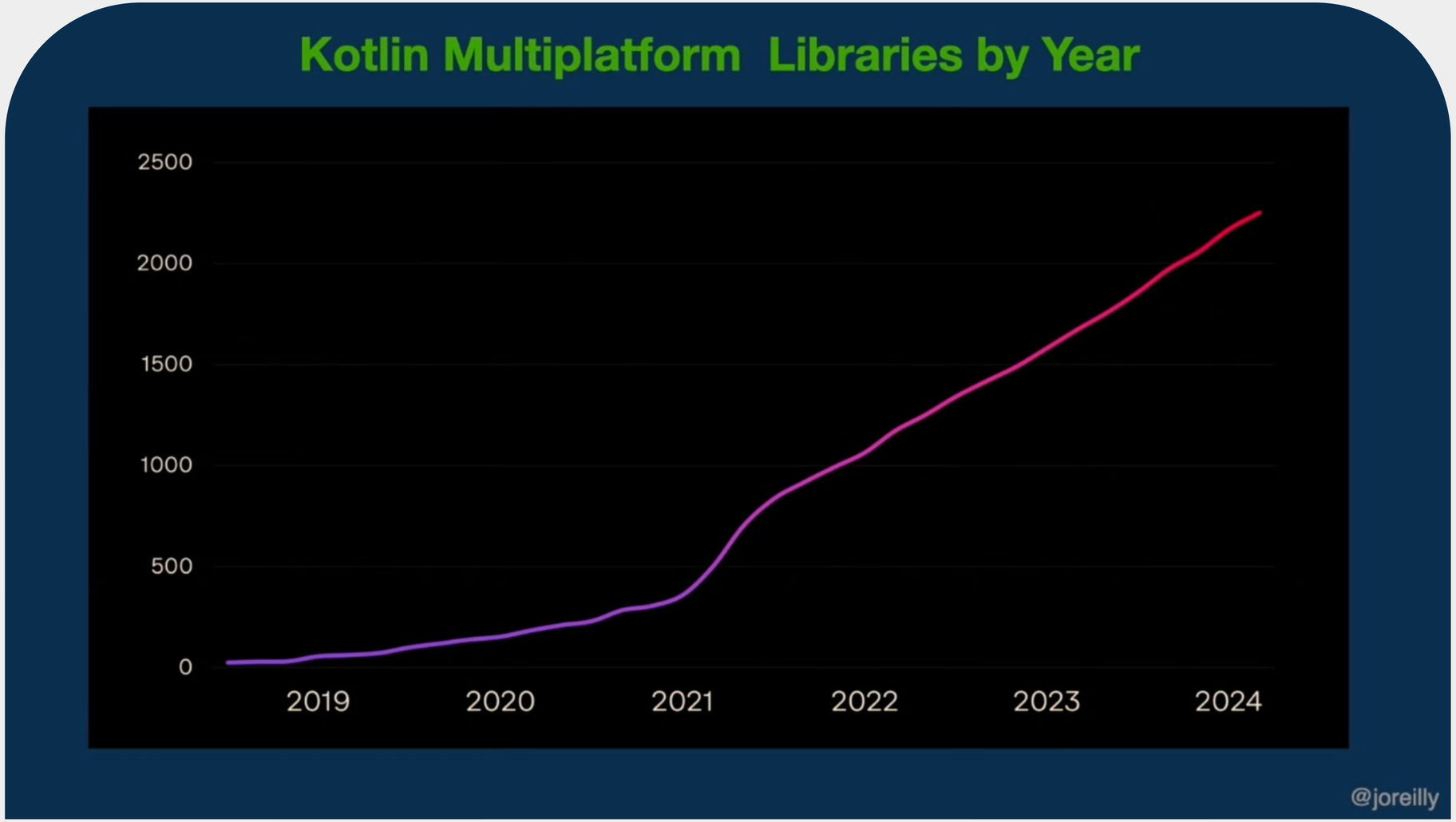
Mobile



Kotlin Multiplatform in Google Workspace

- @Jason Parachoniak
KotlinConf 2024

Mobile



Hitchhiker’s Guide to Kotlin Multiplatform Libraries

- @John O’Reilly
KotlinConf 2024

Google I/O Extended

Hands-on

Demonstration

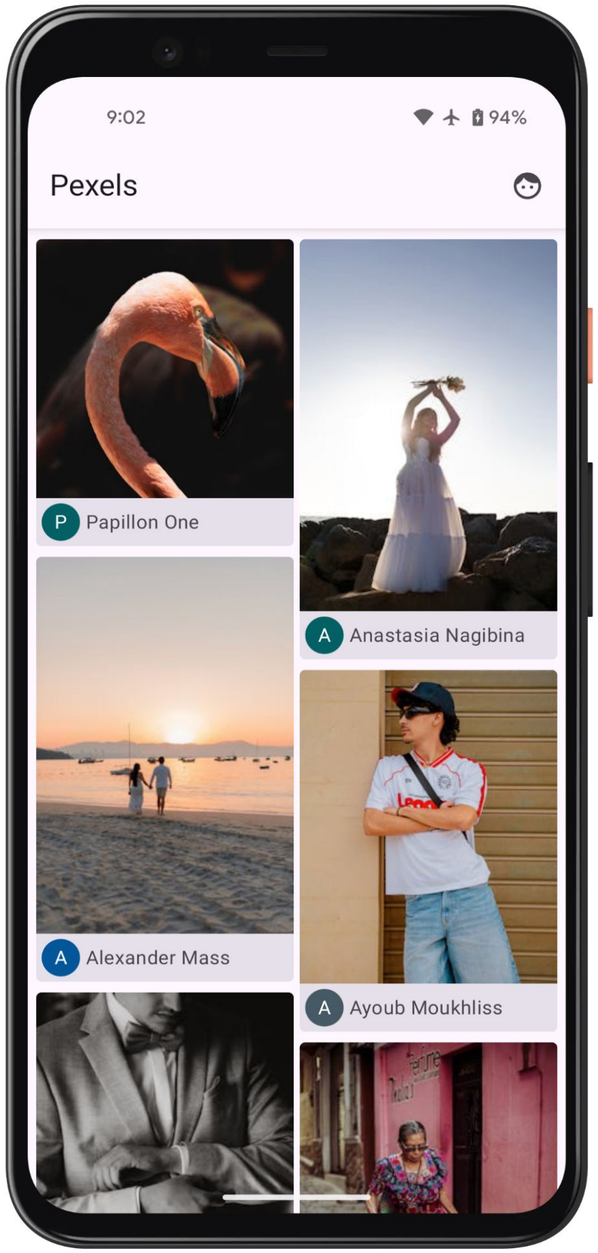
Hands-on

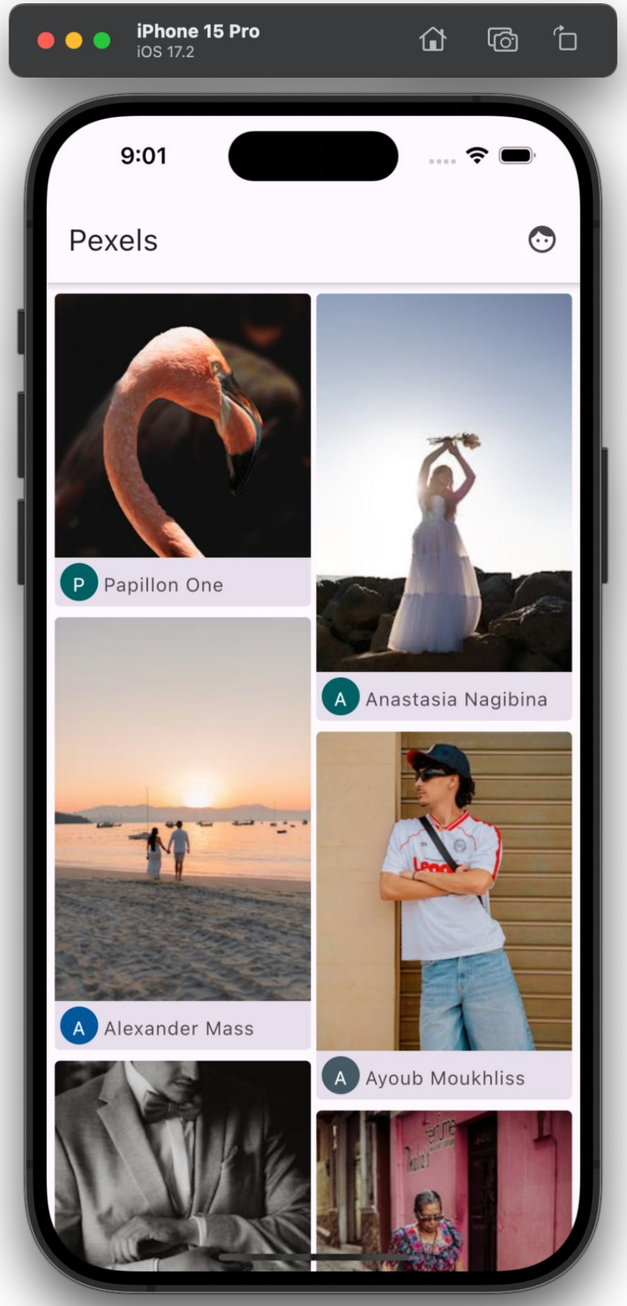
Pexels

Showing up a common 2 columns list with image and text items. The codebase forked from below:

https://github.com/linroid/Pexels
@Lindroid

(It serves as a template for the next demo)



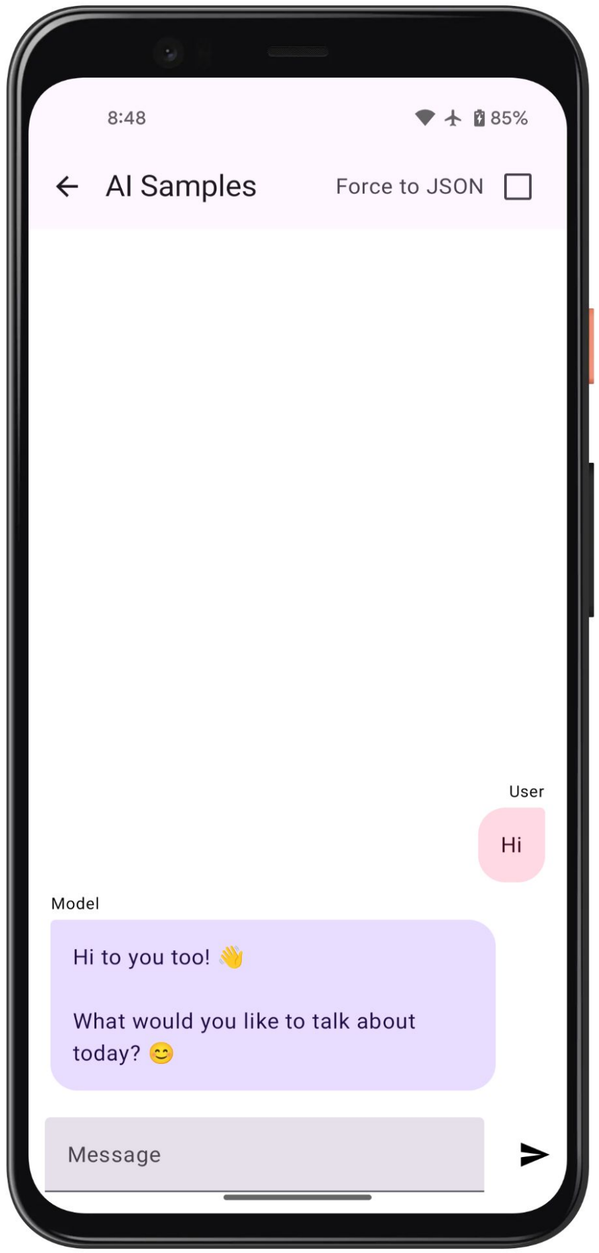


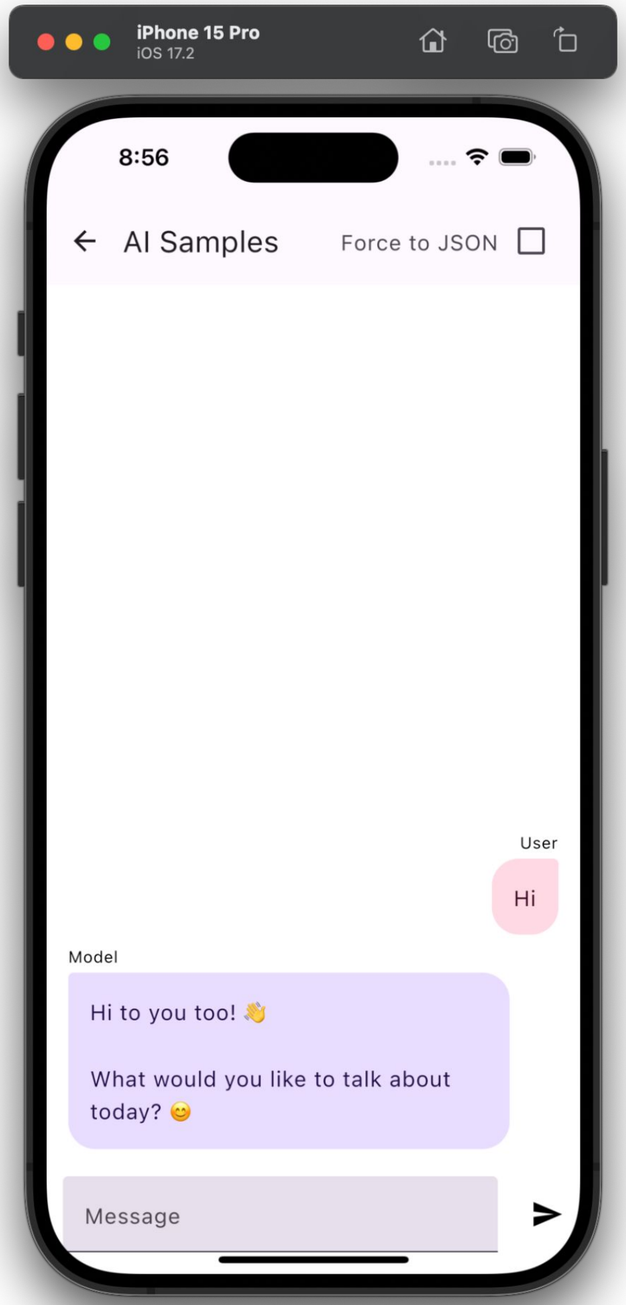
Hands-on

Gemma 2B 1.1

Showing up a conversational chat interface with gemma-2b-1.1 models. The UI design and original codebase come from Google.

https://github.com/google-ai-edge/mediapipe-samples/tree/main/examples/llm_inference/android





Google I/O Extended

Let’s build sth fun!

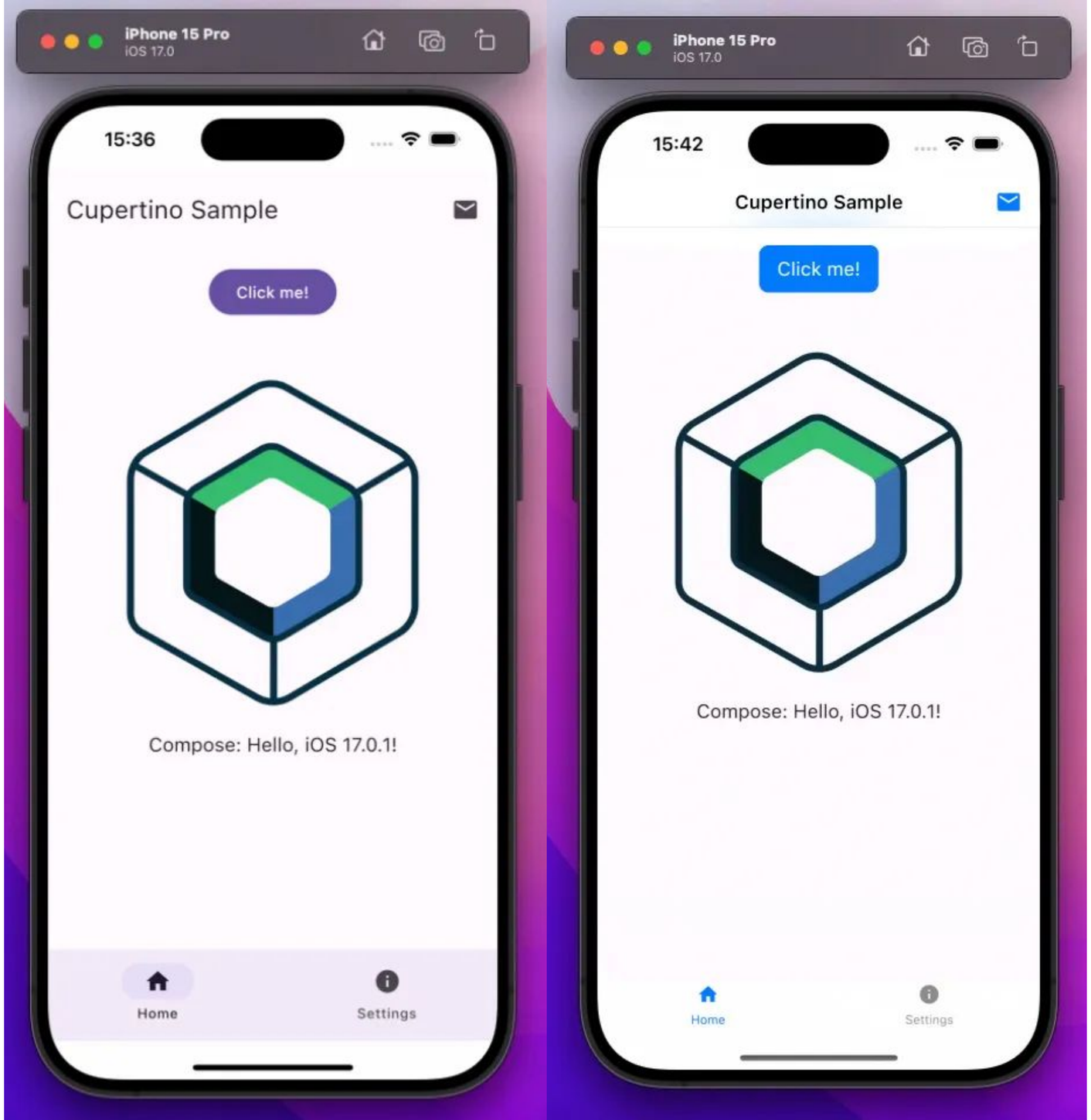
https://github.com/2BAB/Pexels/

Hands-on

Cupertino

Choose the **Material** design for a native Android experience and the **Cupertino** design for a native iOS experience.

https://github.com/alexzhirkevich/compose-cupertino/



Hands-on

Summary of Use cases

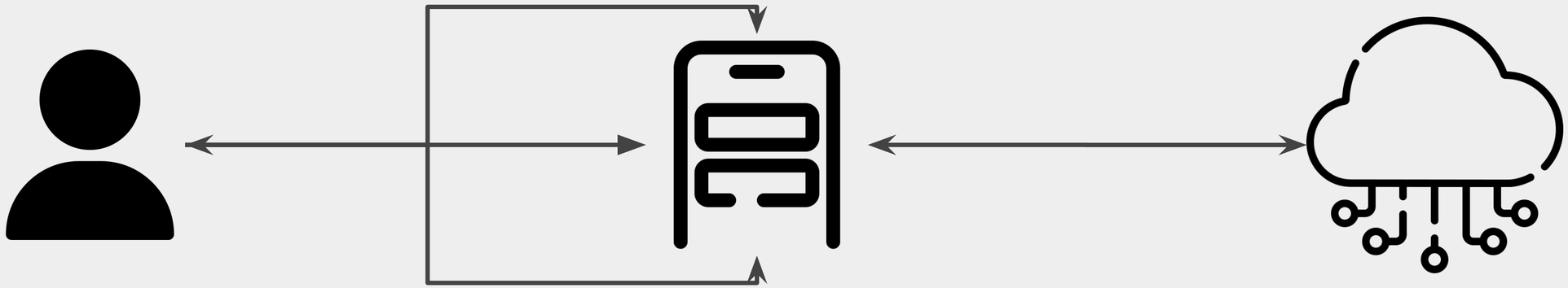
Focus on **Write**

1. Generate emails / stories / poems / jokes / lyrics / etc.
2. Paraphrasing / Grammar Recifying
3. Smart templates in various data reports.
4. ...

Focus on **Read / Analysis / Inference**

1. General Q&A with knowledge base
2. Classification
3. Summary
4. Search with RAG mechanism
5. Work with Multimodal to express multi-media response
6. Work with Function Calling to integrate more system features
7. Work with Response Formatter to generate structural response
8. Analyze sensitive data

Hands-on



未来，用户的多数简单需求可以在本地被满足

Google I/O Extended

Summary

Strengths of On-device Model and KMP

Let’s talk about WHY?

Summary

Why on-device model?

1. Low Cost
2. Low Latency (Yet seen in today’s demo)
3. Privacy

Summary

Why KMP?

Progressive Adoption 渐进式集成

你始终可以选择仅用它来实现部分功能，并充分利用现有的 Android 代码库。以 iOS 为例：一个基于 Kotlin 的 Android 应用可以通过简单的修改轻松移植到 iOS，无需重写整个应用。

Native-level Perf 原生级性能

以 iOS 为例：多数常见场景都能达到 Swift 级别的性能。

Fluent Interop 流畅的互操作

以 iOS 为例：C-Interop 让 Kotlin 获得直接调用 Swift/OC/iOS 依赖的能力；反之 Kotlin 基于 LLVM 编译而成的 .framework 也可以让 Swift 无缝调用 Kotlin 写的功能。

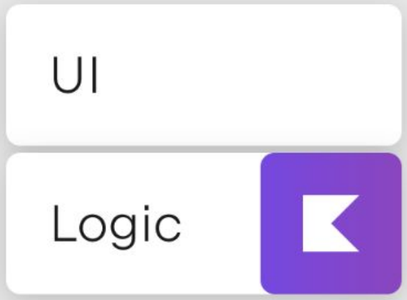
Summary

Why KMP?

Suitable for all kinds of projects

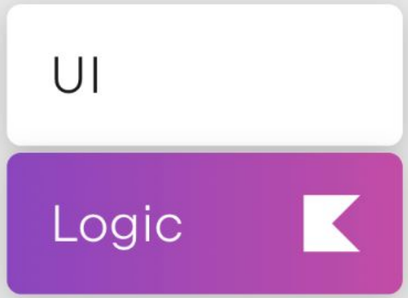
Share a piece of logic

Improve your app's stability by sharing an isolated and critical part of the app. Reuse the Kotlin code you already have to keep the applications in sync.



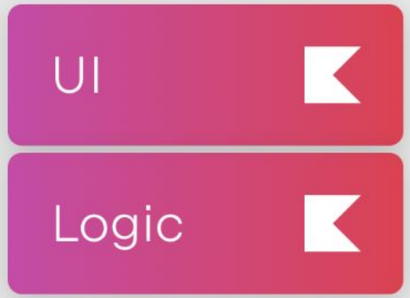
Share logic and keep the UI native

Use Kotlin Multiplatform when you start a new project, and implement data handling and business logic just once. Keep the UI native to meet the most stringent requirements.



Share up to 100% of the code

Elevate development efficiency and share up to 100% of your code with Compose Multiplatform – a modern declarative framework by JetBrains for sharing UI across multiple platforms.



Google I/O Extended

(个人微信)

2bab.me/zh/book/

Thank you.

xx2bab@gmail.com

github.com/2bab

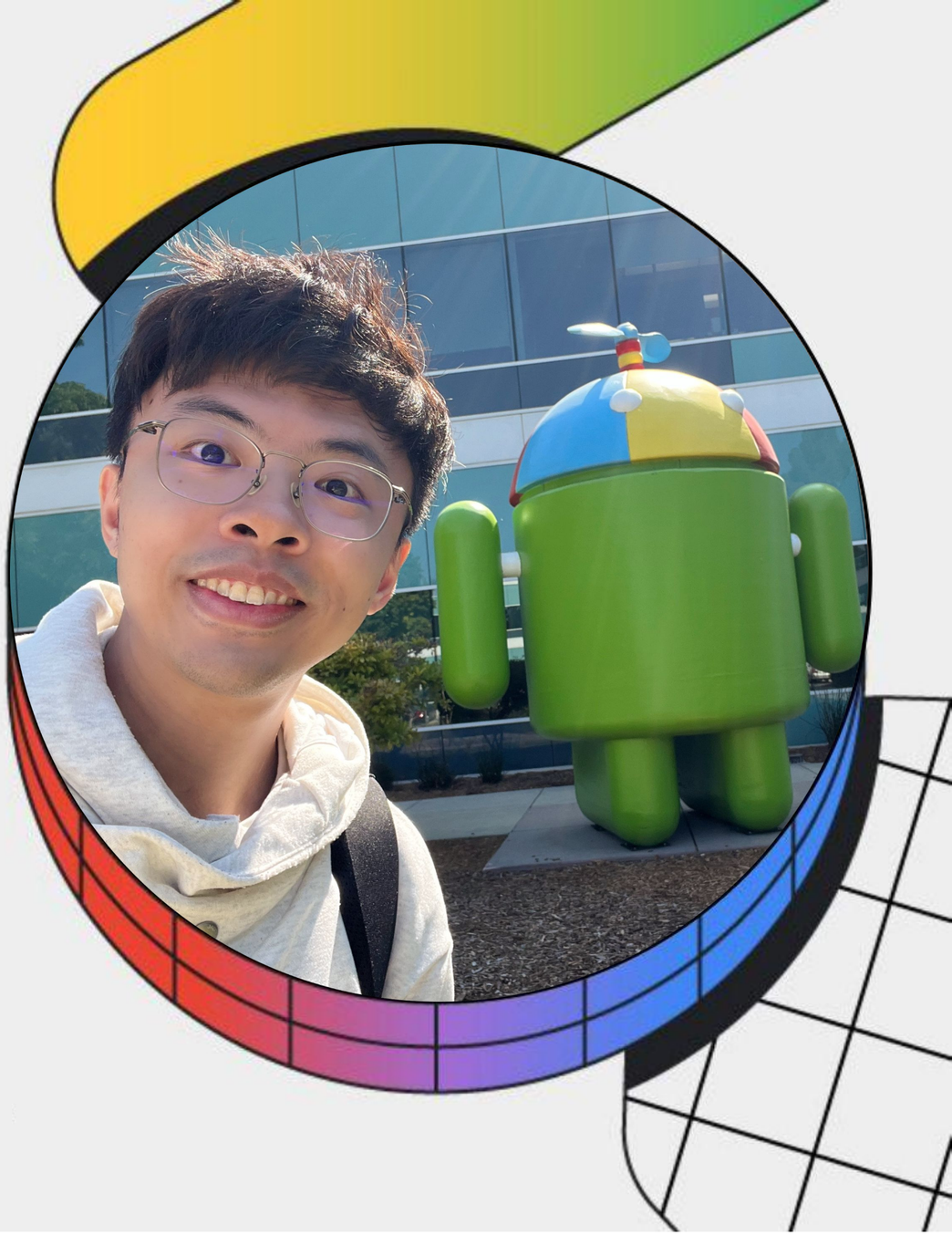
2bab.com

Google I/O Extended Xiamen

June 30, 2024
1:30-5:30 PM

El Zhang @2BAB

On-device Model
Integration and Use Cases (Colab)



Google I/O Extended GuangZhou

July 20, 2024
1:00-6:00 PM

El Zhang @2BAB

On-device Model Integration with Kotlin Multiplatform