



MOBDEV
WE CHANGE THE GAME





PARTE 2 - MIGRANDO A KOTLIN MULTIPLATFORM

POR CARLOS HINOSTROZA
PRINCIPAL SOFTWARE ENGINEER EN MOBDEV



kotlin

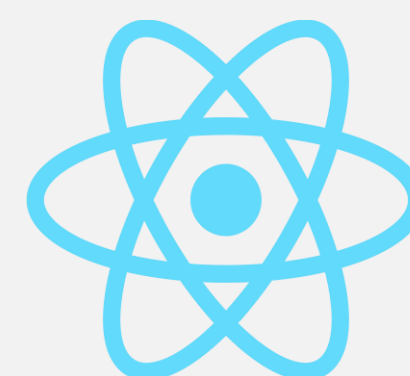
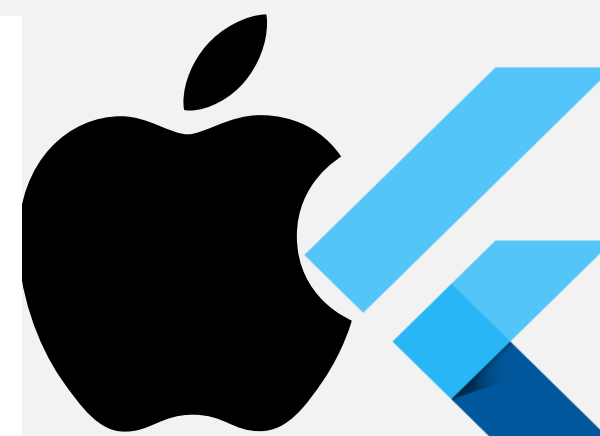
¿ INTRODUCCIÓN ?



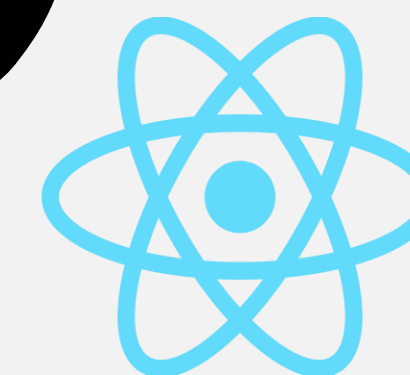
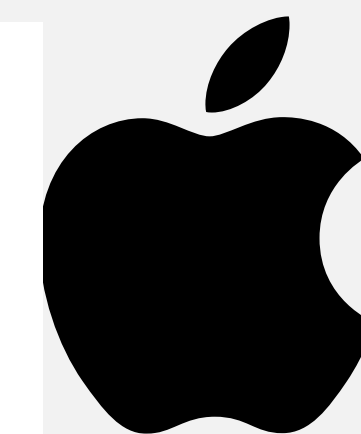


¿ MODELO DE NEGOCIOS ?

TODO DEPENDE



Deseabilidad
¿Qué necesitan las personas?



La idea no se sustenta
a nivel financiero

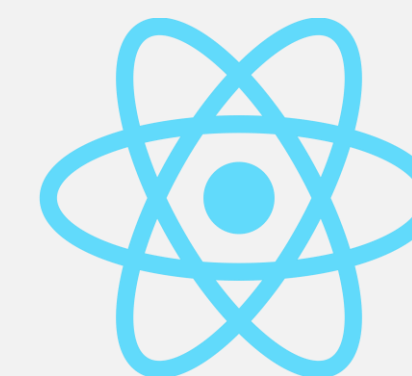
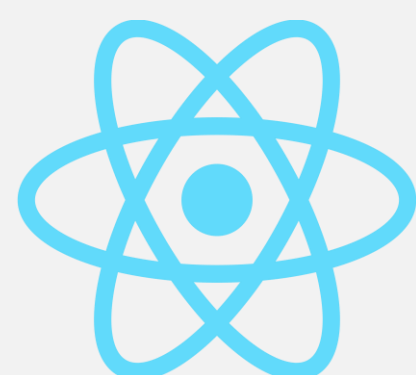
La idea es imposible o
muy difícil de ejecutar

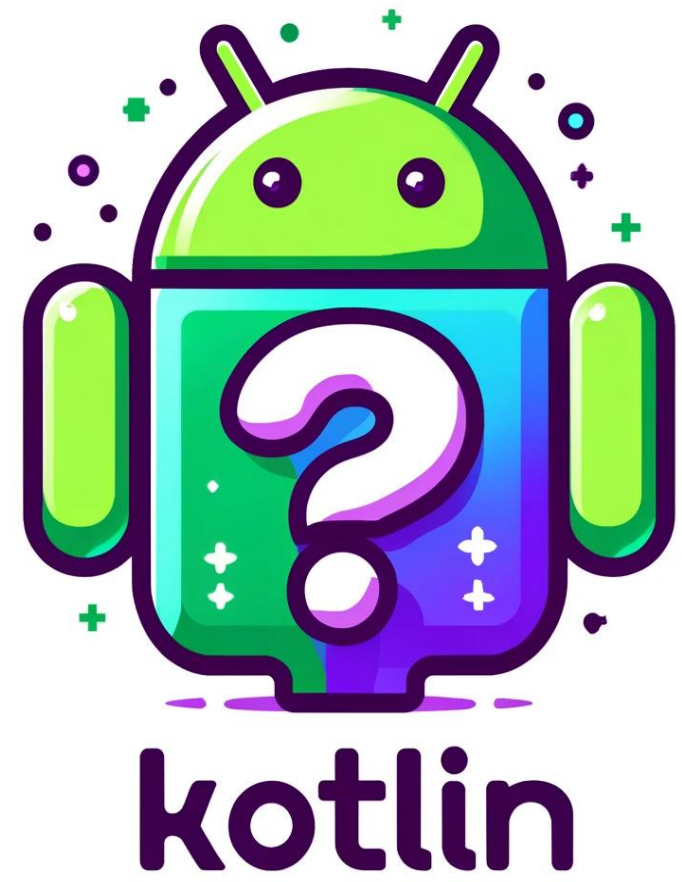
**Idea
Exitosa**

Factibilidad
¿Que es factible
tecnológicamente?

Viabilidad
¿Qué es financieramente
viable?

**Es algo que nadie
quiere**





¿ QUE ES KOTLIN MULTIPLATFORM ?

COMPARTIR CÓDIGO

MANTENIENDO LA FLEXIBILIDAD Y
BENEFICIOS DE LA
PROGRAMACIÓN NATIVA

¿POR QUE ESTO ES IMPORTANTE?



¿ QUE MÁS ES KOTLIN MULTIPLATFORM ?

LEGACY - ROADMAP BROWNFIELD

INTEROPERABILIDAD

RENDIMIENTO

LIBRARY VS FRAMEWORK

MÓDULOS EXISTENTES

COMUNIDAD

KOTLIN

DEVX





ESTADO INICIAL ANTES DE LA MIGRACIÓN

analytics
authorization
biometrics

catalog
crashlytics
network
...

80 .aar

base

Feature 1
Feature 2
Feature 3
...
Feature 20



ESTADO INICIAL

TECNOLOGIAS UTILIZADAS

RETROFIT
DAGGER
FLOW
NAVIGATION
MOCKK
COMPOSE

KOTLIN
CLEAN ARCHITECTURE
MVI - MVVM
ETC
ANDROID LIBRARY
ETC...



PLANIFICACIÓN DE LA MIGRACIÓN

ESTADO COMENZANDO

POC – APP ACTUAL MONTAR KOTLIN MULTIPLATFORM

POC – APP ACTUAL COMPARTIR CAPA DE DATA

POC – APP ACTUAL COMPARTIR CAPA DE DOMINIO

UI NATIVA – SWIFT UI Y JETPACK COMPOSE

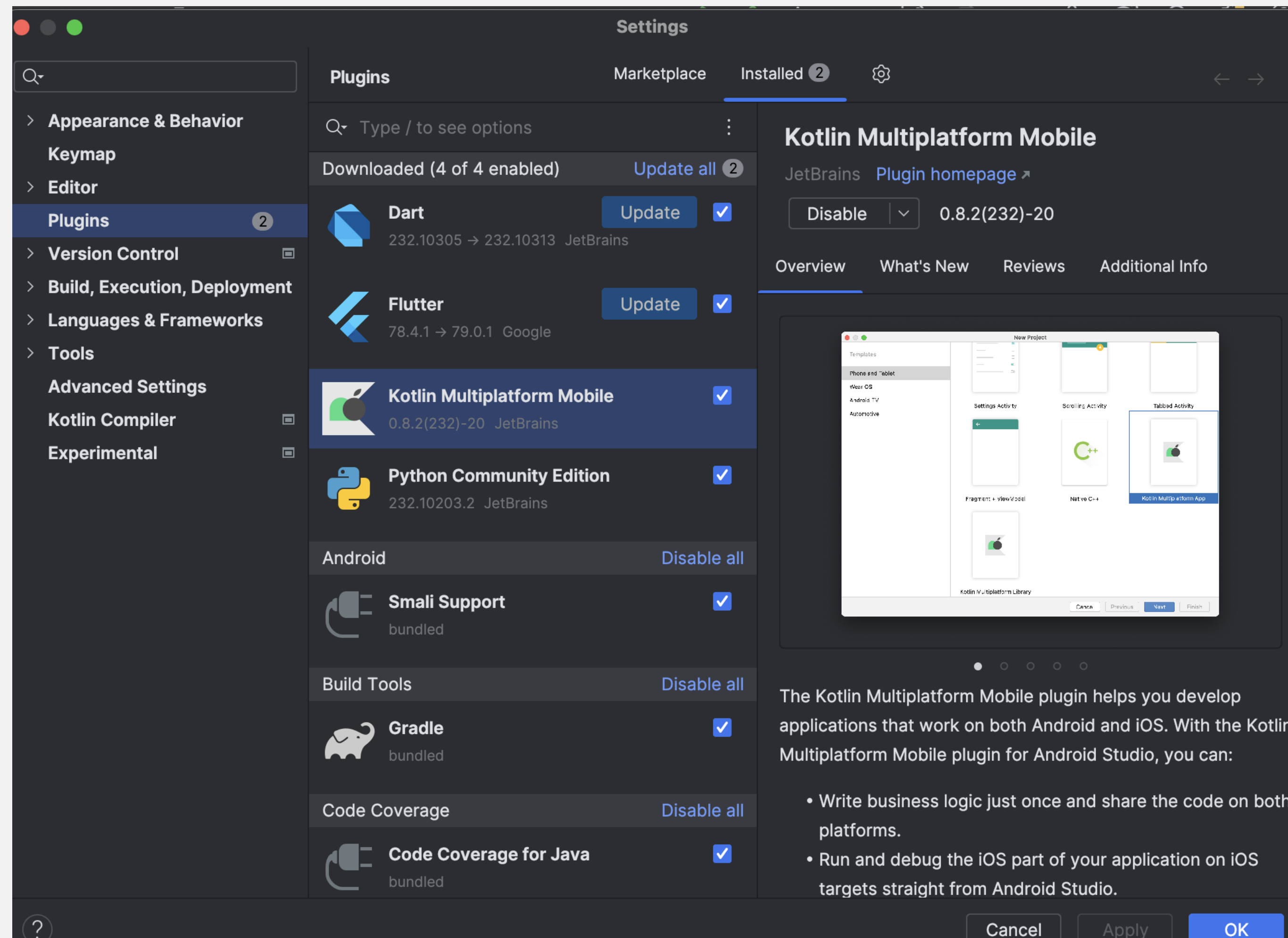
POC – LIBRERÍA DE COMPONENTE

COMPOSE MULTIPLATFORM -PARA APP PEQUEÑAS Y MVP – DEPENDE EL CASO



PRIMEROS PASOS

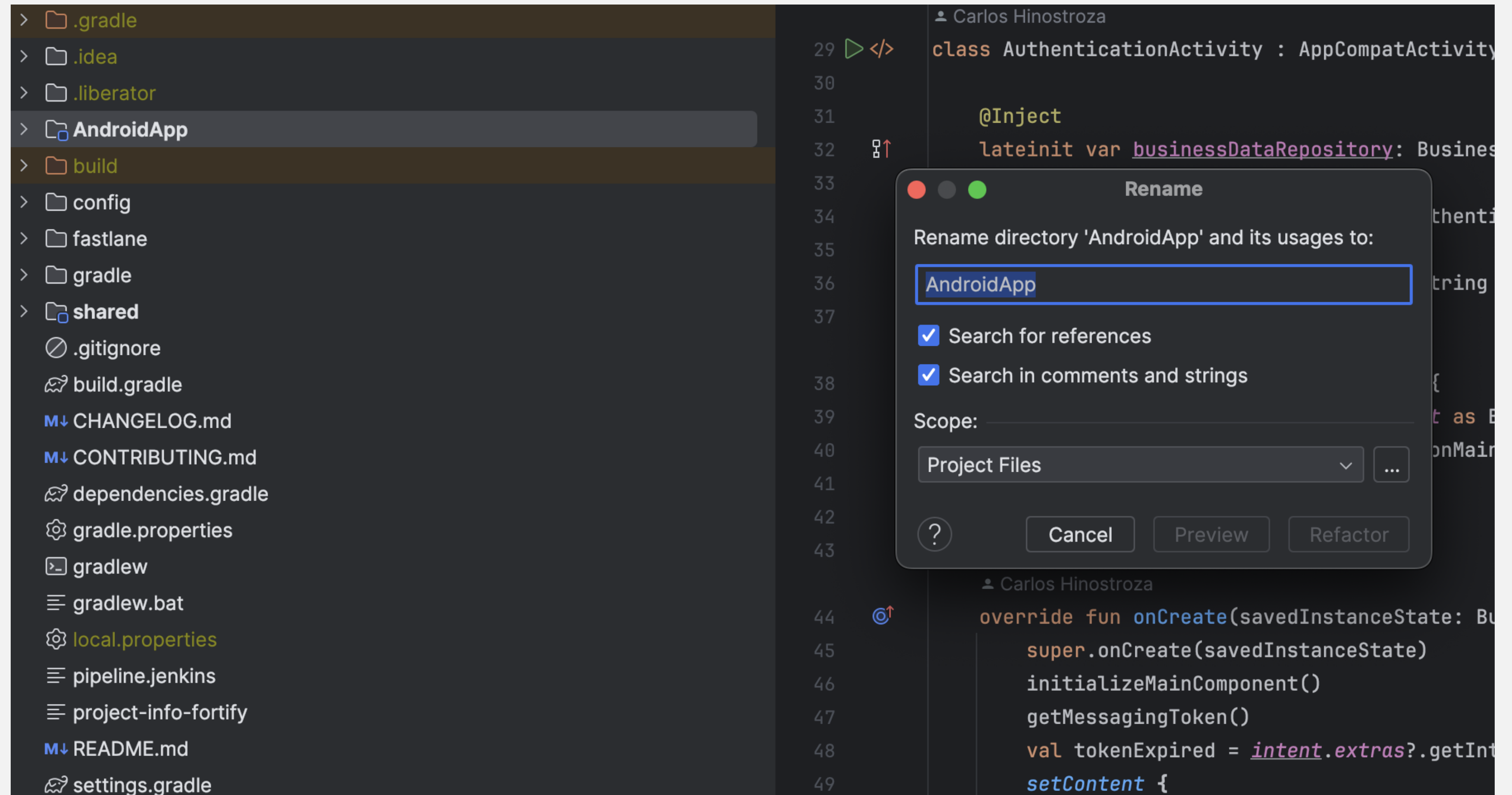
A KOTLIN MULTIPLATFORM - PLUGIN





PRIMEROS PASOS

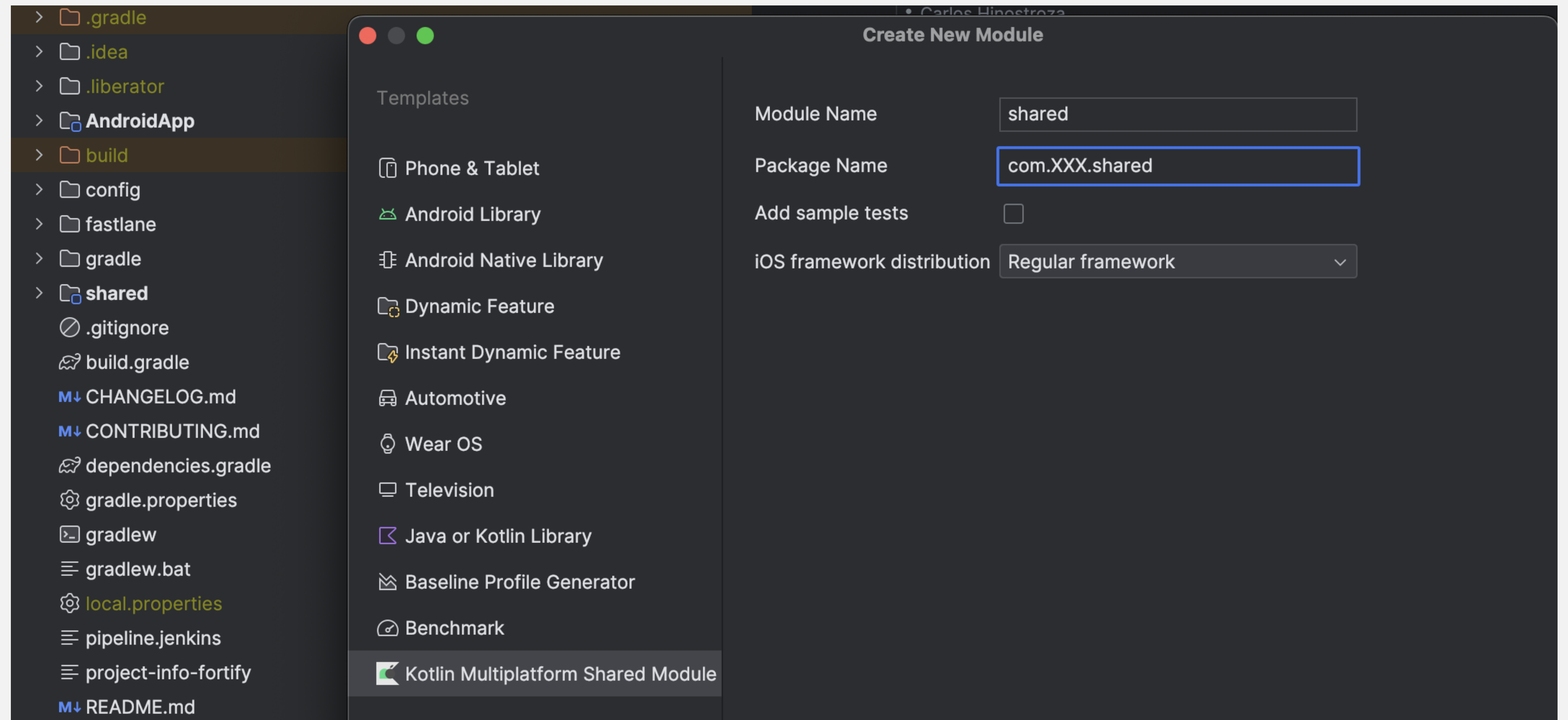
A KOTLIN MP - RENAME BASE MODULE





PRIMEROS PASOS

A KOTLIN MP - CREATE SHARED MODULE





PRIMEROS PASOS A KOTLIN MP - VERSIÓN CATALOG

Sharing dependency versions between projects

Central declaration of dependencies

Using a version catalog

A *version catalog* is a list of dependencies, represented as dependency coordinates, that a user can pick from when declaring dependencies in a build script.

For example, instead of declaring a dependency using a string notation, the dependency coordinates can be picked from a *version catalog*:

Example 1. Using a library declared in a version catalog

Kotlin

Groovy

build.gradle.kts

```
dependencies {  
    implementation(libs.groovy.core)  
}
```



PRIMEROS PASOS

A KOTLIN MP - PROYECTO -> BUILD.GRADLE

The screenshot shows an IDE interface. On the left, a file explorer displays a project structure with folders like .gradle, .idea, .liberator, AndroidApp, build, config, fastlane, gradle, shared, .gitignore, build.gradle, CHANGELOG.md, and CONTRIBUTING.md. The 'build' folder is selected. The main editor area shows the content of 'build.gradle' with the following Kotlin code:

```
7 plugins {
8     alias libs.plugins.application apply false
9     alias libs.plugins.library apply false
10    alias libs.plugins.kotlin apply false
11    alias libs.plugins.ksp apply false
12    alias libs.plugins.parcelize apply false
13    alias libs.plugins.navigationSafeArgs apply false
14    alias libs.plugins.artifactory
15    alias libs.plugins.sonarqube
16    alias libs.plugins.detekt
17    alias libs.plugins.kotlinMultiplatform apply false
18 }
```

Below the editor, the Git history for 'build.gradle' is shown. It lists three commits by Carlos Hinostrroza:

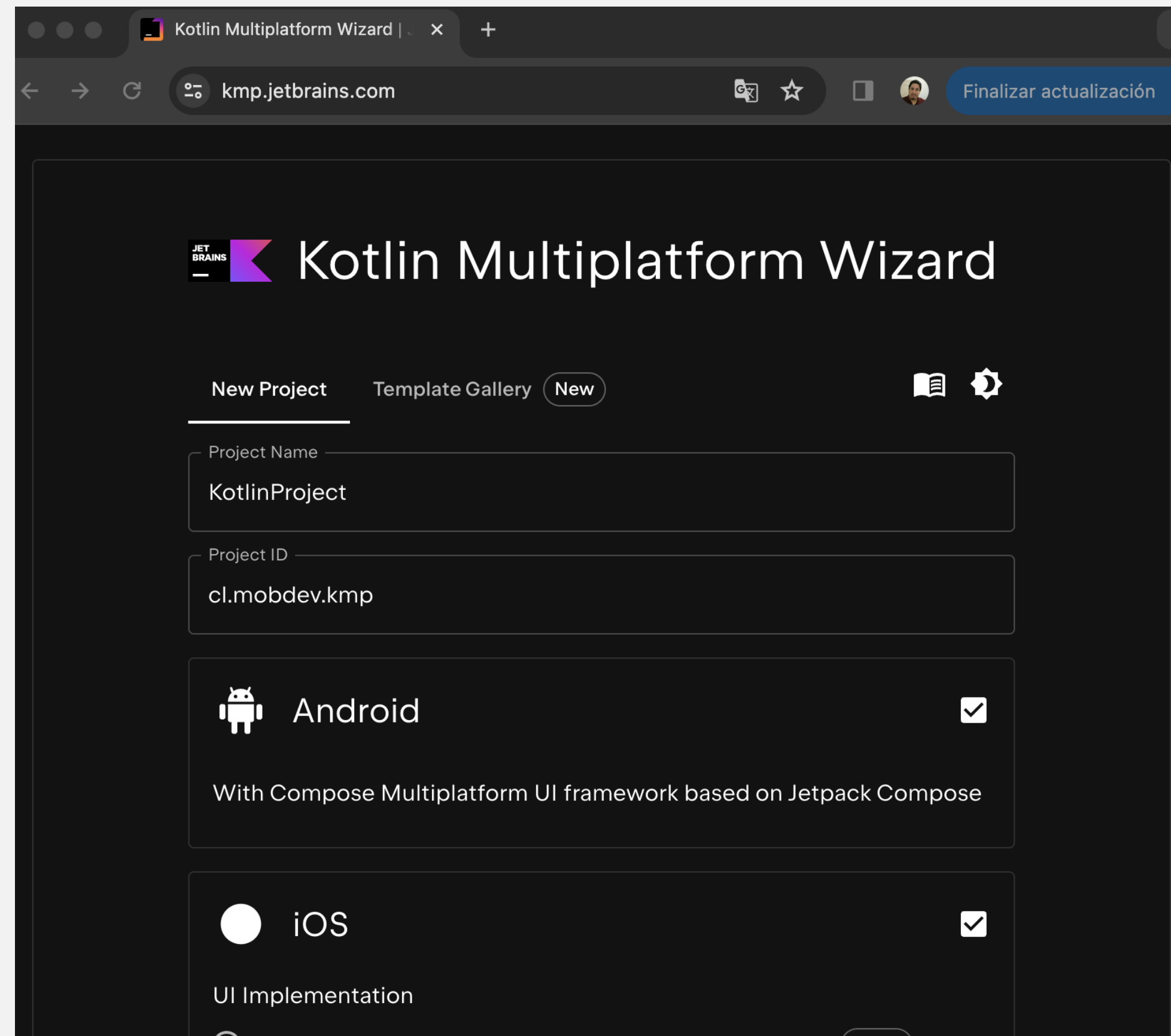
- 03-04-24, 6:34 PM: react native ok (flutter)
- 01-04-24, 3:21 PM: kotlin multiplatform run...
- 27-03-24, 10:04 AM: init repo

The right side of the Git panel shows a diff view for commit 50238b6e (build.gradle), highlighting the change to delete rootProject.buildDir.



PRIMEROS PASOS

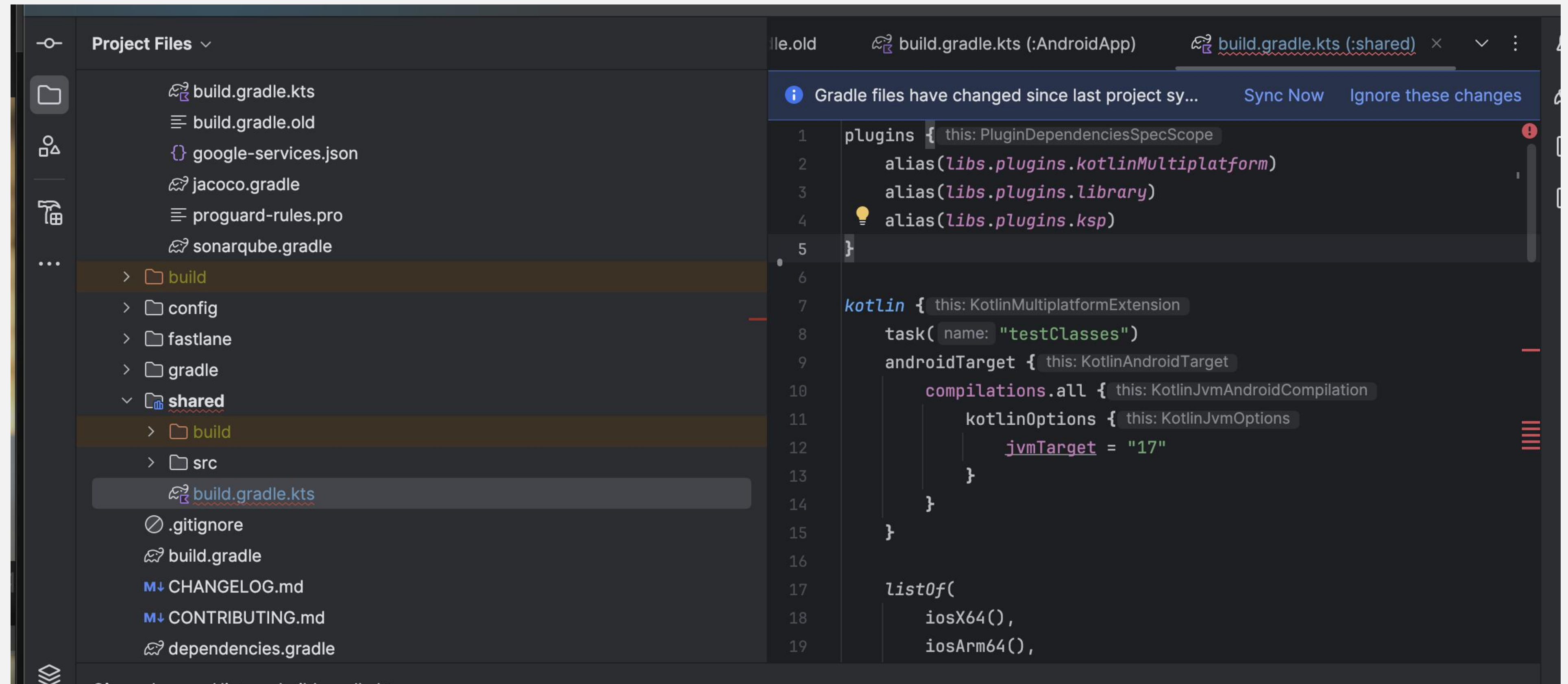
A KOTLIN MP - ANDROID-APP-> BUILD.GRADLE.KTS
Build.gradle(respalda) migrar a Build.gradle.kts





PRIMEROS PASOS

A KOTLIN MP - SHARED MODULE-> BUILD.GRADLE.KTS



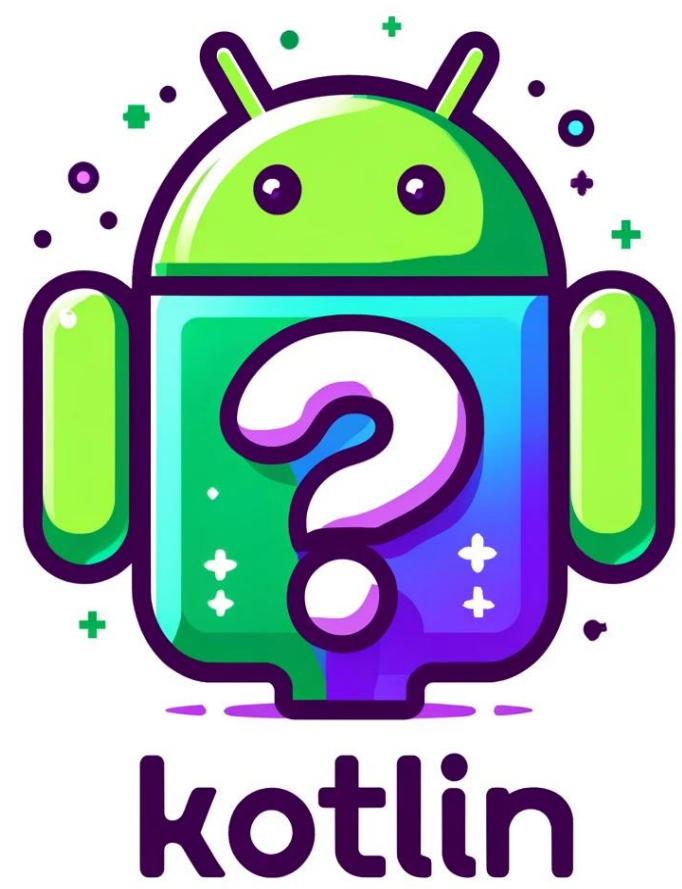


kotlin

DESAFIOS Y SOLUCIONES

PRINCIPALES OBTACULOS





PRÓXIMOS PASOS

...

POC – APP ACTUAL MONTAR KOTLIN MULTIPLATFORM

POC – APP ACTUAL COMPARTIR CAPA DE DATA

POC – APP ACTUAL COMPARTIR CAPA DE DOMINIO

UI NATIVA – SWIFT UI Y JETPACK COMPOSE

NAVIGATION, DI, ETC...

POC – LIBRERÍA DE COMPONENTE
COMPOSE MULTIPLATFORM – PARA APP PEQUEÑAS Y MVP – DEPENDE
EL CASO

PREGUNTAS Y RESPUESTAS

