

CHECKLIST



Checklist Post-Migración Kotlin 2.0

Ejecutar estos pasos en orden después de hacer pull de los cambios



Pre-Build (preparación)

- **Cerrar Android Studio** (si está abierto)
- **Hacer pull** de los últimos cambios
- **Navegar a directorio:**

```
cd Zabala\Gaietak/android-app
```



Verificación Automática

- **Ejecutar script de verificación:**
./post-migration-check.sh
 - Si todo pasa: continuar
 - Si falla: revisar output y corregir antes de continuar



Build Inicial

- **Abrir Android Studio**
- **Invalidate Caches:**
 - File → Invalidate Caches / Restart
 - Seleccionar “Invalidate and Restart”
 - Esperar reinicio
- **Gradle Sync:**
 - Esperar sincronización automática
 - O manualmente: File → Sync Project with Gradle Files
 - ⏳ Primera vez tomará más tiempo (descarga AGP 9, Kotlin 2.0.21, etc.)
- **Rebuild Project:**
 - Build → Rebuild Project
 - ⏳ Build inicial será lento (~5-10 min)



Testing

- **Unit Tests:**

```
./gradlew :app:testDebugUnitTest
```

- **Compilar Release:**

```
./gradlew :app:assembleRelease
```

- **Crear emulador API 24** (nueva compatibilidad):

- Tools → Device Manager → Create Device
 - API Level: 24 (Android 7.0)
 - Ejecutar app en emulador
-



Seguridad (CRÍTICO)

- **Buscar uso de security-crypto:**

```
grep -r "EncryptedSharedPreferences\|EncryptedFile\|MasterKey" app/src/
```

- **Si hay resultados:**

- Leer sección “Migración de security-crypto” en [MIGRATION_KOTLIN_2.0.md](#)
- Implementar alternativa con Android Keystore
- Probar migración de datos (si hay datos cifrados existentes)
- Ejecutar tests de seguridad

- **Si NO hay resultados:**

- No requiere acción
-



Validación de Mejoras

- **Verificar build speed:**

```
# Build con profiling  
./gradlew :app:assembleDebug --profile
```

```
# Ver reporte generado  
open build/reports/profile/profile-*.html
```

- **Comparar tiempos:**

- Anotar tiempo de incremental build
 - Hacer cambio pequeño en código
 - Re-compilar y comparar (debería ser 30-40% más rápido)
-



Verificaciones Finales

- **Auto-completion funciona** en Android Studio para:
 - Clases anotadas con `@HiltViewModel`
 - DAOs de Room (generados por KSP)
 - Inject constructors
- **No hay errores de compilación** en:

- Activities/Fragments
 - ViewModels con Hilt
 - Room DAOs/Entities
 - Retrofit interfaces
 - **App funciona correctamente:**
 - Login/autenticación
 - Navegación entre pantallas
 - Persistencia de datos (Room)
 - Llamadas a API (Retrofit)
 - Biometría (si aplica)
-

Reportar Problemas

Si encuentras problemas:

1. Verificar versión de Android Studio:

- Help → About
- Debe ser **Ladybug (2024.2)** o superior

2. Verificar JDK:

- File → Project Structure → SDK Location
- Debe ser **JDK 17**

3. Limpiar y rebuild:

```
./gradlew clean  
./gradlew :app:assembleDebug --info
```

4. Revisar logs detallados:

- Build Output en Android Studio
- O terminal: ./gradlew :app:assembleDebug --stacktrace

5. Consultar documentación:

- [MIGRATION_KOTLIN_2.0.md](#) - Guía completa
 - [UPGRADE_SUMMARY.md](#) - Resumen de cambios
-

¡Completado!

Si todos los checks están marcados, la migración fue exitosa.

Próximos pasos: - Continuar desarrollo normal - Disfrutar builds más rápidos con KSP - Monitorear mejoras en velocidad con --profile

Última actualización: 2026-01-23