

Zabala Gailetak HR - Android App

Aplicación móvil Android para el sistema de gestión de recursos humanos de Zabala Gailetak.

Stack Actualizado (2026-01-23)

- **Gradle:** 8.10.2
- **Android Gradle Plugin:** 8.7.3
- **Kotlin:** 2.0.21 (con Compose plugin oficial)
- **KSP:** 2.0.21-1.0.28 (migrado desde KAPT)
- **Min SDK:** 24 (Android 7.0) - ampliada compatibilidad
- **Target SDK:** 35 (Android 15)
- **JDK:** 17

Tecnologías

- **Lenguaje:** Kotlin 2.0.21
- **UI:** Jetpack Compose (BOM 2024.12.01) + Material 3
- **Arquitectura:** Clean Architecture + MVI
- **DI:** Hilt 2.54 (con KSP)
- **Networking:** Retrofit 2.11.0 + OkHttp 4.12.0
- **Local DB:** Room 2.6.1 (con KSP)
- **Async:** Coroutines 1.9.0 + Flow
- **Seguridad:** Credential Manager API 1.5.0, Biometric 1.2.0, Android Keystore

 **Nota:** androidx.security:security-crypto fue eliminado (deprecado). Ver [MIGRATION_KOTLIN_2.0.md](#) para migración.

Requisitos

- **Android Studio:** Koala (2024.1) o superior
- **JDK:** 17
- **Android SDK:** 35
- **Gradle:** 8.10.2 (incluido en wrapper)

Setup del Proyecto

Primera vez (o después de actualización)

1. Clonar el repositorio
2. Abrir el proyecto en Android Studio
3. **Importante:** Si vienes de versión anterior, ejecutar script de migración:

```
cd android-app  
./post-migration-check.sh
```

4. En Android Studio: **File → Invalidate Caches / Restart**
5. Sync Gradle files (puede tardar en primera sincronización con AGP 9)
6. Configurar emulador o dispositivo físico
7. Run app

Después de actualización Kotlin 2.0 / AGP 9

Si experimentas problemas después del pull:

- Ejecutar `./post-migration-check.sh` para verificar configuración
- Limpiar caches: `./gradlew clean`
- Iniciar caches de Android Studio
- Consultar [MIGRATION_KOTLIN_2.0.md](#) para detalles completos

Estructura del Proyecto

```
app/  
  -- src/main/  
    -- java/com/zabalagaileta/hrapp/  
      -- HrApplication.kt          # Application class  
      -- data/  
        -- local/                 # Data layer  
        -- remote/                # Local data (Room)  
        -- repository/            # Remote data (Retrofit)  
        -- repository/             # Repository implementations  
      -- di/                      # Dependency injection  
        -- AppModule.kt  
        -- NetworkModule.kt  
        -- DatabaseModule.kt  
        -- RepositoryModule.kt  
      -- domain/  
        -- model/                 # Domain layer  
        -- model/                  # Domain models
```

```

    └── repository/           # Repository interfaces
        └── usecase/          # Use cases

    └── presentation/         # Presentation layer
        ├── MainActivity.kt
        └── navigation/
            └── ui/
                └── viewmodel/ # ViewModels

    └── security/            # Security utilities
    └── util/                # Utilities

    └── res/                 # Resources
        ├── values/
        ├── drawable/
        └── xml/

└── AndroidManifest.xml

build.gradle.kts

```

Configuración de API

El endpoint de la API se configura en build.gradle.kts:

- **Debug:** `http://10.0.2.2:8080/api/` (localhost desde emulador)
- **Release:** `http://zabalagaitak.rf.gd/api/` (InfinityFree)

Para cambiar la URL en desarrollo, edita la variable `API_BASE_URL` en el `buildType debug`.

Testing

```

# Unit tests
./gradlew test

# Instrumented tests
./gradlew connectedAndroidTest

# Lint check
./gradlew lint

```

Seguridad

-  Network Security Config (solo HTTPS en producción)
-  Certificate pinning
-  EncryptedSharedPreferences para datos sensibles

- Credential Manager API para passkeys
- Biometric authentication
- ProGuard/R8 ofuscación en release
- No cleartext traffic

Features Implementadas

- [] Autenticación (Login/Logout)
- [] MFA (TOTP)
- [] Passkey authentication
- [] Gestión de empleados
- [] Solicitud de vacaciones
- [] Consulta de nóminas
- [] Gestión de documentos
- [] Chat interno
- [] Sistema de quejas
- [] Notificaciones push

Build & Deploy

Debug Build

```
./gradlew assembleDebug
```

Release Build

```
./gradlew assembleRelease
```

El APK se generará en: app/build/outputs/apk/release/app-release.apk

Signing

Para firmar la app en release:

1. Crear keystore
2. Configurar signing en build.gradle.kts
3. Build release

Previews en Android Studio

Esta app incluye **Compose Previews** para todas las pantallas principales, permitiéndote visualizar la UI sin ejecutar la app.

Acceso rápido a Previews

- **LoginScreen:** 3 variantes (normal, loading, error)
- **DashboardScreen:** Vista principal
- **DocumentsScreen:** Lista de documentos
- **PayslipsScreen:** Nóminas (2 variantes)
- **ProfileScreen:** Perfil de usuario
- **VacationDashboardScreen:** Dashboard de vacaciones
- **NewVacationRequestScreen:** Solicitud nueva (2 variantes)

Documentación completa: Ver [PREVIEWS_GUIDE.md](#)

Verificar Previews

```
./verify-previews.sh
```

Este script verifica que todos los archivos tengan previews configurados correctamente.



Documentación

-
- [API Documentation](#)
 - [Security Guidelines](#)
 - [Architecture](#)
 - [Previews Guide - Cómo usar Compose Previews](#)



Contribuir

Ver [CONTRIBUTING.md](#)



Licencia

Propietario - Zabala Galetak

Versión: 1.0.0

Última actualización: Febrero 2026