

# App Control (Flutter)

Todos los repositorios se guardan en:

C:\\overskull\\

## ? 1. Versiones necesarias para este proyecto

Herramienta	Versión
Flutter	1.17.5
Dart	Incluido en Flutter 1.17.5
JDK	Amazon Corretto 8
Gradle	5.6.2 (wrapper del proyecto)

⚠ Este proyecto es LEGACY. No actualizar versiones.

## ? 2. Descargar Flutter (versión obligatoria del proyecto)

Descargar Flutter **1.17.5** desde el archivo histórico:

<https://docs.flutter.dev/release/archive>

Guardar Flutter en:

C:\\overskull\\SDKFlutter\\

Ejemplo:

C:\\overskull\\SDKFlutter\\sdk1.17.5\\flutter

No usar Flutter global del sistema.

## ? 3. Instalar JDK 8 (Amazon Corretto)

Descargar desde:

<https://docs.aws.amazon.com/corretto/latest/corretto-8-ug/downloads-list.html>

☐ Guardar Java en:

```
C:\\overskull\\java\\
```

Ejemplo:

```
C:\\overskull\\java\\amazon-corretto-8
```

## ? 4. Clonar el proyecto

```
git clone git@storegit.overskull.com:shalom-over-app/shalom_control.git
```

Entrar al proyecto:

```
cd shalom_control
```

## ? 5. build.gradle (revisión inicial necesaria para arrancar el proyecto)

Ruta:

```
C:\\overskull\\shalom_control\\android\\build.gradle
```

Ejemplo de referencia:

```
buildscript {
    repositories {
        google()
        jcenter()
    }

    dependencies {
        classpath 'com.android.tools.build:gradle:3.5.0'
        classpath 'com.google.gms:google-services:4.2.0'
    }

    subprojects {
        project.configurations.all {
            resolutionStrategy.eachDependency { details ->
                if (details.requested.group == 'com.android.support'
```

```

        && !details.requested.name.contains('multidex') ) {
            details.useVersion "27.1.1"
        }
        /*if (details.requested.group == 'androidx.core'
            && !details.requested.name.contains('androidx') ) {
                details.useVersion "1.0.1"
            }
        if (details.requested.group == 'androidx.lifecycle' &&
            !details.requested.name.contains('androidx')) {
                details.useVersion "2.0.0"
            }*/

        if (details.requested.group == 'androidx.core') {
            details.useVersion "1.0.1"
        }
        if (details.requested.group == 'androidx.lifecycle') {
            details.useVersion "2.0.0"
        }
        if (details.requested.group == 'androidx.versionedparcelable') {
            details.useVersion "1.0.0"
        }
        if (details.requested.group == 'androidx.fragment') {
            details.useVersion "1.0.0"
        }
        if (details.requested.group == 'androidx.appcompat') {
            details.useVersion "1.0.1"
        }
    }
}

allprojects {
    repositories {
        google()
        mavenCentral()
        maven { url '<https://jitpack.io>' }
    }
    configurations.all {
        resolutionStrategy {

```

```

        force 'pub.devrel:easypermissions:3.0.0'
    }
}

rootProject.buildDir = '../build'
subprojects {
    project.buildDir = "${rootProject.buildDir}/${project.name}"
}
subprojects {
    project.evaluationDependsOn(':app')
}

task clean(type: Delete) {
    delete rootProject.buildDir
}

```

Ruta:

```
C:\overskull\shalom_control\android\app\build.gradle
```

Ejemplo de referencia:

```

def localProperties = new Properties()
def localPropertiesFile = rootProject.file('local.properties')
if (localPropertiesFile.exists()) {
    localPropertiesFile.withReader('UTF-8') { reader ->
        localProperties.load(reader)
    }
}

def flutterRoot = localProperties.getProperty('flutter.sdk')
if (flutterRoot == null) {
    throw new GradleException("Flutter SDK not found. Define location with flutter.sdk in the
local.properties file.")
}

def flutterVersionCode = localProperties.getProperty('flutter.versionCode')
if (flutterVersionCode == null) {
    flutterVersionCode = '1'
}

```

```
}

def flutterVersionName = localProperties.getProperty('flutter.versionName')
if (flutterVersionName == null) {
    flutterVersionName = '1.0'
}

apply plugin: 'com.android.application'
apply from: "$flutterRoot/packages/flutter_tools/gradle/flutter.gradle"

def keystoreProperties = new Properties()
def keystorePropertiesFile = rootProject.file('key.properties')
if (keystorePropertiesFile.exists()) {
    keystoreProperties.load(new FileInputStream(keystorePropertiesFile))
}

android {
    compileSdkVersion 30

    lintOptions {
        disable 'InvalidPackage'
    }

    defaultConfig {
        // TODO: Specify your own unique Application ID
        (<https://developer.android.com/studio/build/application-id.html>).
        applicationId "com.shalom.overskull"
        minSdkVersion 16
        targetSdkVersion 35
        versionCode 452
        versionName '4.5.2'
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }

    signingConfigs {
        release {
            keyAlias keystoreProperties['keyAlias']
            keyPassword keystoreProperties['keyPassword']
            storeFile keystoreProperties['storeFile'] ? file(keystoreProperties['storeFile'])
        }
    }
}
```

```
        storePassword keystoreProperties['storePassword']
    }
}
buildTypes {
    release {
        signingConfig signingConfigs.release
    }
}
}

flutter {
    source '../..'
}

dependencies {
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
    implementation 'pub.devrel:easypermissions:3.0.0'
}
```

☐ Verificar:

- minSdkVersion y targetSdkVersion originales
- No modificar plugins

⚠ *Cambios en este archivo pueden romper la app.*

## ?? 6. Gradle Wrapper (configuración obligatoria)

Este proyecto **NO usa gradle.properties** para Java.

Editar:

```
C:\\overskull\\shalom_control\\android\\gradle\\wrapper\\gradle-wrapper.properties
```

Contenido de referencia:

```
#Thu May 08 17:02:41 PET 2025
distributionBase=GRADLE_USER_HOME
distributionPath=wrapper/dists
distributionUrl=https\://services.gradle.org/distributions/gradle-5.6.2-all.zip
zipStoreBase=GRADLE_USER_HOME
zipStorePath=wrapper/dists
```

**No cambiar la versión de Gradle.**

---

## ? 7. Configurar la versión de Flutter usada por el proyecto

Editar:

```
C:\\overskull\\shalom_control\\.vscode\\settings.json
```

Contenido:

```
{
  "dart.flutterSdkPath": "C:\\\\SDKFlutter\\\\sdk1.17.5\\\\flutter"
}
```

## ? 8. Limpiar el proyecto

```
cd android
gradlew clean
```

## ? 9. Descargar dependencias

```
flutter pub get
```

## ? 10. Ejecutar el proyecto

```
flutter run
```

## ? Notas importantes

- Este proyecto usa **Flutter 1.17.5 + Java 8**.
- No actualizar Flutter, Gradle ni Java.
- No migrar a AndroidX ni cambiar plugins.

- Recomendado usar este proyecto **solo para mantenimiento.**
- Si falla la compilación:

---

Revisión #4

Creado 2025-12-13 12:47:50 -05 por Daisy

Actualizado 2025-12-13 13:10:59 -05 por Daisy