

MANUAL TECNICO - PROYECTO COMPI1 ANDROID

DESCRIPCION GENERAL

- Tipo de proyecto: Aplicacion Android (Kotlin + Gradle)
- Paquete principal: com.cunoc.compi1
- Objetivo: Analizar pseudocodigo, generar estructura de nodos y mostrar un diagrama de flujo en pantalla.
- Componentes clave:
 - UI Android: MainActivity + DiagramaFlujoView
 - Analisis: Lexer (JFlex) + parser (CUP)
 - Construcccion de modelo: GraphBuilder y ConfigBuilder

VERSIONES Y REQUISITOS TECNICOS

2.1 Herramientas base

- Gradle Wrapper: 9.2.1
 - Archivo: gradle/wrapper/gradle-wrapper.properties
 - distributionUrl: <https://services.gradle.org/distributions/gradle-9.2.1-bin.zip>
- Android Gradle Plugin (AGP): 9.0.1
 - Archivo: gradle/libs.versions.toml
- Java para compilacion del modulo app: Java 21
 - Archivo: app/build.gradle.kts
 - sourceCompatibility = JavaVersion.VERSION_21
 - targetCompatibility = JavaVersion.VERSION_21

2.2 Android SDK

- compileSdk: 36 (minorApiLevel 1)
- targetSdk: 36
- minSdk: 24
- versionCode: 1
- versionName: 1.0

2.3 Librerías principales (según catálogo de versiones)

- androidx.core:core-ktx:1.17.0
- androidx.appcompat:appcompat:1.6.1
- com.google.android.material:material:1.10.0
- androidx.activity:activity:1.12.4
- androidx.constraintlayout:constraintlayout:2.1.4
- junit:junit:4.13.2
- androidx.test.ext:junit:1.3.0
- androidx.test.espresso:espresso-core:3.7.0

2.4 Dependencia parser runtime

- app/libs/java-cup-11b-runtime.jar
- Esta librería es necesaria para ejecutar parser.java y sym.java generados por CUP.

ESTRUCTURA TÉCNICA RELEVANTE

- app/src/main/java/com/cunoc/compi1/MainActivity.kt
 - Entrada de texto, disparo de análisis y render del diagrama.
- app/src/main/java/com/cunoc/compi1/compiler/DiagramAnalyzer.kt
 - Orquestación de lexer + parser + construcción de nodos/configuración.
- app/src/main/java/com/cunoc/compi1/compiler/ParseErrorHelper.kt
 - Formateo de errores léxicos/sintácticos/excepciones.
- app/src/main/java/com/cunoc/compi1/nodos/GraphBuilder.kt
 - Convierte tokens en nodos de dominio.
- app/src/main/java/com/cunoc/compi1/nodos/ConfigBuilder.kt
 - Lee configuraciones por clave e índice.
- app/src/main/java/com/cunoc/compi1/nodos/DiagramaFlujoView.kt
 - Render de nodos y estilos.
- app/src/main/java/compi1/practica_1_compi/lexer/
 - jflex.jflex (fuente lexer)

- Lexer.java (generado)
- app/src/main/java/com/pi/practica_1_compi/parser/
 - parser.cup (fuente parser)
 - parser.java, sym.java (generados)

INSTALACION (MODO PRODUCCION CON APK)

4.1 Ubicacion del instalador

- El APK de distribucion se encuentra en la carpeta /documentacion/.

4.2 Requisitos del dispositivo Android

- Android 7.0 o superior (minSdk 24).
- Espacio disponible para instalacion.

4.3 Instalacion manual desde el telefono

1. Copiar el APK al dispositivo.
2. Abrir el APK desde el explorador de archivos.
3. Si Android lo solicita, habilitar "Instalar aplicaciones desconocidas" para la app desde la que se abre el APK.
4. Confirmar instalacion.

4.4 Instalacion por ADB (opcional tecnico)

- Con el dispositivo conectado y depuracion USB activa: adb install -r "ruta\al\archivo.apk"

4.5 Verificacion post-instalacion

- Abrir la app instalada.
- Ingresar un pseudocodigo simple.
- Ejecutar analisis y confirmar que el diagrama se dibuja en pantalla.

INSTALACION DESDE CODIGO FUENTE (SOLO DESARROLLO)

5.1 Requisitos previos en maquina (Windows)

- JDK 21 instalado y configurado en PATH o JAVA_HOME.
- Android Studio reciente (recomendado) con SDK 36 instalado.
- Gradle no necesita instalacion global, porque se usa Gradle Wrapper.

5.2 Abrir proyecto

- Abrir la carpeta raíz del proyecto: `c:\Users\antho\Documents\Kotlin_projects`

5.3 Configurar local.properties

- Debe existir `local.properties` con `sdk.dir` apuntando al Android SDK local.
- Ejemplo (ajustar ruta real):
`sdk.dir=C:\Users\TU_USUARIO\AppData\Local\Android\Sdk`

5.4 Sincronizar dependencias

- Android Studio: usar "Sync Project with Gradle Files".
- CLI (PowerShell en raíz del proyecto): `.\gradlew.bat --version` `.\gradlew.bat tasks`

5.5 Compilar e instalar debug

- Compilar: `.\gradlew.bat assembleDebug`
- Instalar: `.\gradlew.bat installDebug`

5.6 Pruebas (opcional)

- Unitarias: `.\gradlew.bat test`
- Instrumentadas: `.\gradlew.bat connectedAndroidTest`

FLUJO FUNCIONAL DEL PROGRAMA

- El usuario escribe pseudocódigo en la UI.
- `MainActivity` invoca `DiagramAnalyzer.analizar(codigo)`.
- `Lexer` y `parser` analizan entrada.
- Si hay errores, se construye reporte detallado para UI.
- Si no hay errores:
 - `GraphBuilder.construir(tokens)` genera nodos.
 - `ConfigBuilder.construir(tokens)` genera configuración por clave/índice.
- `DiagramaFlujoView` dibuja nodos aplicando estilos configurados.
- REGENERACION DE LEXER/PARSER (CUANDO SE MODIFICA `jflex.jflex` o `parser.cup`)

IMPORTANTE:

- En el repositorio ya existen archivos generados (`Lexer.java`, `parser.java`, `sym.java`).
- Si se modifica la gramática o reglas léxicas, se deben regenerar y volver a compilar.

7.1 Regenerar lexer con JFlex (ejemplo)

- `jflex app/src/main/java/com/pi/practica_1_compi/lexer/jflex.jflex`

7.2 Regenerar parser con CUP (ejemplo)

- `java -jar java-cup-11b.jar`
`-parser parser`
`-symbols sym`
`-destdir app/src/main/java/com/pi/practica_1_compi/parser`
`app/src/main/java/com/pi/practica_1_compi/parser/parser.cup`

Notas:

- El runtime de CUP ya esta en app/libs (java-cup-11b-runtime.jar).
- El jar generador (java-cup-11b.jar) y JFlex pueden requerir instalacion externa si no estan en la maquina.

8. CONFIGURACION DSL (RESUMEN)

- Bloque algoritmo: INICIO ... FIN
- Bloque configuracion separado por: %%%%
- Configuracion por indice con formato: %CLAVE = valor | indice

Ejemplos de claves:

- %COLOR_BLOQUE
- %COLOR_TEXTO_BLOQUE
- %FIGURA_BLOQUE
- %LETRA_BLOQUE
- %LETRA_SIZE_BLOQUE
- %COLOR_SI, %FIGURA_SI, etc.

Formato de color soportado en ConfigBuilder:

- Hex con prefijo H: H1E88E5
- RGB con comas: 255,255,255

PROBLEMAS COMUNES Y SOLUCIONES

9.1 Error "Unresolved reference: android"

- Causa: proyecto no sincronizado como Android o SDK faltante.
- Solucion:

- Abrir en Android Studio.
- Verificar local.properties (sdk.dir).
- Sincronizar Gradle.

9.2 Error "Unresolved reference: java_cup"

- Causa: classpath sin runtime CUP.
- Solucion:
 - Verificar que exista app/libs/java-cup-11b-runtime.jar.
 - Verificar dependencia en app/build.gradle.kts:
implementation(files("libs/java-cup-11b-runtime.jar"))

9.3 Error de version Kotlin metadata (2.3.0 vs 2.1.0)

- Causa: desalineacion entre plugin Kotlin del IDE y stdlib/caché.
- Solucion recomendada:
 - Limpiar caches de Gradle/IDE.
 - Reimportar proyecto.
 - Alinear version de plugin Kotlin de IDE con toolchain del proyecto.

9.4 Parser/lexer no reflejan cambios de .cup/.jflex

- Causa: no se regeneraron los archivos Java generados.
- Solucion:
 - Regenerar Lexer.java, parser.java y sym.java.
 - Reconstruir proyecto.

COMANDOS RAPIDOS (WINDOWS - DESARROLLO)

- Ver informacion Gradle: .\gradlew.bat --version
- Limpiar build: .\gradlew.bat clean
- Compilar debug: .\gradlew.bat assembleDebug
- Ejecutar tests: .\gradlew.bat test

RECOMENDACIONES DE MANTENIMIENTO

- Mantener sincronizados:
 - parser.cup <-> parser.java/sym.java

- jflex.jflex <=> Lexer.java
- Documentar cualquier cambio de gramatica en este manual.
- Evitar actualizar AGP/Gradle/JDK sin validar compatibilidad.