

Tổng hợp mã nguồn dự án: AnTamViecLam

.gitignore

```
<<<<<<< HEAD
```

```
local.properties
```

```
>>>>>>> origin/main
```

README.md

```
# AnTamViecLam
```

build.gradle.kts

```
// Top-level build file where you can add configuration options common to all sub-projects/modules.
```

```
plugins {  
    alias(libs.plugins.android.application) apply false  
    alias(libs.plugins.kotlin.android) apply false  
    // alias(libs.plugins.kotlin.compose) apply false  
    alias(libs.plugins.android.library) apply false  
    alias(libs.plugins.hilt) apply false  
    alias(libs.plugins.ksp) apply false  
    alias(libs.plugins.google.services) apply false  
    alias(libs.plugins.firebase.crashlytics) apply false  
    alias(libs.plugins.android.secrets.gradle.plugin) apply false  
  
}
```

gradle.properties

```
# Project-wide Gradle settings.  
# IDE (e.g. Android Studio) users:  
# Gradle settings configured through the IDE *will override*  
# any settings specified in this file.  
# For more details on how to configure your build environment visit  
# http://www.gradle.org/docs/current/userguide/build_environment.html  
# Specifies the JVM arguments used for the daemon process.  
# The setting is particularly useful for tweaking memory settings.  
org.gradle.jvmargs=-Xmx2048m -Dfile.encoding=UTF-8  
# When configured, Gradle will run in incubating parallel mode.
```

```
# This option should only be used with decoupled projects. For more details, visit
# https://developer.android.com/r/tools/gradle-multi-project-decoupled-projects
# org.gradle.parallel=true
# AndroidX package structure to make it clearer which packages are bundled with the
# Android operating system, and which are packaged with your app's APK
# https://developer.android.com/topic/libraries/support-library/androidx-rn
android.useAndroidX=true
# Kotlin code style for this project: "official" or "obsolete":
kotlin.code.style=official
# Enables namespacing of each library's R class so that its R class includes only the
# resources declared in the library itself and none from the library's dependencies,
# thereby reducing the size of the R class for that library
android.nonTransitiveRClass=true
```

gradlew

```
#!/bin/sh
```

```
#
# Copyright © 2015 the original authors.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#   https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
# SPDX-License-Identifier: Apache-2.0
#

#####
#####

#
# Gradle start up script for POSIX generated by Gradle.
#
# Important for running:
#
# (1) You need a POSIX-compliant shell to run this script. If your /bin/sh is
```

```

# noncompliant, but you have some other compliant shell such as ksh or
# bash, then to run this script, type that shell name before the whole
# command line, like:
#
# ksh Gradle
#
# Busybox and similar reduced shells will NOT work, because this script
# requires all of these POSIX shell features:
# * functions;
# * expansions «$var», «${var}», «${var:-default}», «${var+SET}»,
#   «${var#prefix}», «${var%suffix}», and «$( cmd )»;
# * compound commands having a testable exit status, especially «case»;
# * various built-in commands including «command», «set», and «ulimit».
#
# Important for patching:
#
# (2) This script targets any POSIX shell, so it avoids extensions provided
# by Bash, Ksh, etc; in particular arrays are avoided.
#
# The "traditional" practice of packing multiple parameters into a
# space-separated string is a well documented source of bugs and security
# problems, so this is (mostly) avoided, by progressively accumulating
# options in "$@", and eventually passing that to Java.
#
# Where the inherited environment variables (DEFAULT_JVM_OPTS, JAVA_OPTS,
# and GRADLE_OPTS) rely on word-splitting, this is performed explicitly;
# see the in-line comments for details.
#
# There are tweaks for specific operating systems such as AIX, CygWin,
# Darwin, MinGW, and NonStop.
#
# (3) This script is generated from the Groovy template
# https://github.com/gradle/gradle/blob/HEAD/platforms/jvm/plugins-
application/src/main/resources/org/gradle/api/internal/plugins/unixStartScript.txt
# within the Gradle project.
#
# You can find Gradle at https://github.com/gradle/gradle/.
#
#####
#####

# Attempt to set APP_HOME

```

```

# Resolve links: $0 may be a link
app_path=$0

# Need this for daisy-chained symlinks.
while
  APP_HOME=${app_path%"${app_path##*/}"} # leaves a trailing /; empty if no leading
path
  [ -h "$app_path" ]
do
  ls=$( ls -ld "$app_path" )
  link=${ls##* -> '}
  case $link in
    /*) app_path=$link ;; #(
    *)  app_path=$APP_HOME$link ;;
  esac
done

# This is normally unused
# shellcheck disable=SC2034
APP_BASE_NAME=${0##*/}
# Discard cd standard output in case $CDPATH is set
(https://github.com/gradle/gradle/issues/25036)
APP_HOME=$( cd -P "${APP_HOME:-.}" > /dev/null && printf '%s\n' "$PWD" ) || exit

# Use the maximum available, or set MAX_FD != -1 to use that value.
MAX_FD=maximum

warn () {
  echo "$*"
} >&2

die () {
  echo
  echo "$*"
  echo
  exit 1
} >&2

# OS specific support (must be 'true' or 'false').
cygwin=false
msys=false
darwin=false
nonstop=false

```

```

case "$(uname)" in
    CYGWIN*)      cygwin=true ;;
    Darwin*)      darwin=true ;;
    MSYS* | MINGW*)  msys=true ;;
    NONSTOP*)      nonstop=true ;;
esac

```

```

CLASSPATH=""

```

```

# Determine the Java command to use to start the JVM.
if [ -n "$JAVA_HOME" ] ; then
    if [ -x "$JAVA_HOME/jre/sh/java" ] ; then
        # IBM's JDK on AIX uses strange locations for the executables
        JAVACMD=$JAVA_HOME/jre/sh/java
    else
        JAVACMD=$JAVA_HOME/bin/java
    fi
    if [ ! -x "$JAVACMD" ] ; then
        die "ERROR: JAVA_HOME is set to an invalid directory: $JAVA_HOME

```

Please set the JAVA_HOME variable in your environment to match the location of your Java installation."

```

    fi
else
    JAVACMD=java
    if ! command -v java >/dev/null 2>&1
    then
        die "ERROR: JAVA_HOME is not set and no 'java' command could be found in your
PATH.

```

Please set the JAVA_HOME variable in your environment to match the location of your Java installation."

```

    fi
fi

```

```

# Increase the maximum file descriptors if we can.
if ! "$cygwin" && ! "$darwin" && ! "$nonstop" ; then
    case $MAX_FD in
        max*)
            # In POSIX sh, ulimit -H is undefined. That's why the result is checked to see if it
            worked.
            # shellcheck disable=SC2039,SC3045

```

```

MAX_FD=$( ulimit -H -n ) ||
    warn "Could not query maximum file descriptor limit"
esac
case $MAX_FD in #(
    " | soft) ;; #(
    *)
        # In POSIX sh, ulimit -n is undefined. That's why the result is checked to see if it worked.
        # shellcheck disable=SC2039,SC3045
        ulimit -n "$MAX_FD" ||
            warn "Could not set maximum file descriptor limit to $MAX_FD"
esac
fi

# Collect all arguments for the java command, stacking in reverse order:
# * args from the command line
# * the main class name
# * -classpath
# * -D...appname settings
# * --module-path (only if needed)
# * DEFAULT_JVM_OPTS, JAVA_OPTS, and GRADLE_OPTS environment variables.

# For Cygwin or MSYS, switch paths to Windows format before running java
if "$cygwin" || "$msys" ; then
    APP_HOME=$( cygpath --path --mixed "$APP_HOME" )
    CLASSPATH=$( cygpath --path --mixed "$CLASSPATH" )

    JAVACMD=$( cygpath --unix "$JAVACMD" )

    # Now convert the arguments - kludge to limit ourselves to /bin/sh
    for arg do
        if
            case $arg in
                -*) false ;;
                /*) t=${arg#/} t=/${t%%/*} # looks like a POSIX filepath
                    [ -e "$t" ] ;;
                *) false ;;
            esac
        then
            arg=$( cygpath --path --ignore --mixed "$arg" )
        fi
        # Roll the args list around exactly as many times as the number of
        # args, so each arg winds up back in the position where it started, but
        # possibly modified.

```

```

#
# NB: a `for` loop captures its iteration list before it begins, so
# changing the positional parameters here affects neither the number of
# iterations, nor the values presented in `arg`.
shift          # remove old arg
set -- "$@" "$arg"  # push replacement arg
done
fi

# Add default JVM options here. You can also use JAVA_OPTS and GRADLE_OPTS to pass
JVM options to this script.
DEFAULT_JVM_OPTS="-Xmx64m" "-Xms64m"

# Collect all arguments for the java command:
# * DEFAULT_JVM_OPTS, JAVA_OPTS, and optsEnvironmentVar are not allowed to contain
shell fragments,
# and any embedded shellness will be escaped.
# * For example: A user cannot expect ${Hostname} to be expanded, as it is an
environment variable and will be
# treated as '${Hostname}' itself on the command line.

set -- \
    "-Dorg.gradle.appname=$APP_BASE_NAME" \
    -classpath "$CLASSPATH" \
    -jar "$APP_HOME/gradle/wrapper/gradle-wrapper.jar" \
    "$@"

# Stop when "xargs" is not available.
if ! command -v xargs >/dev/null 2>&1
then
    die "xargs is not available"
fi

# Use "xargs" to parse quoted args.
#
# With -n1 it outputs one arg per line, with the quotes and backslashes removed.
#
# In Bash we could simply go:
#
# readarray ARGS < <( xargs -n1 <<<"$var" ) &&
# set -- "${ARGS[@]}" "$@"
#

```

```

# but POSIX shell has neither arrays nor command substitution, so instead we
# post-process each arg (as a line of input to sed) to backslash-escape any
# character that might be a shell metacharacter, then use eval to reverse
# that process (while maintaining the separation between arguments), and wrap
# the whole thing up as a single "set" statement.
#
# This will of course break if any of these variables contains a newline or
# an unmatched quote.
#

eval "set -- ${
    printf '%s\n' "$DEFAULT_JVM_OPTS $JAVA_OPTS $GRADLE_OPTS" |
    xargs -n1 |
    sed 's~[^-[:alnum:]+,./:=@_]~\\&~g; ' |
    tr '\n' ' '
}" "$@"

exec "$JAVACMD" "$@"

```

gradlew.bat

```

@rem
@rem Copyright 2015 the original author or authors.
@rem
@rem Licensed under the Apache License, Version 2.0 (the "License");
@rem you may not use this file except in compliance with the License.
@rem You may obtain a copy of the License at
@rem
@rem    https://www.apache.org/licenses/LICENSE-2.0
@rem
@rem Unless required by applicable law or agreed to in writing, software
@rem distributed under the License is distributed on an "AS IS" BASIS,
@rem WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
@rem See the License for the specific language governing permissions and
@rem limitations under the License.
@rem
@rem SPDX-License-Identifier: Apache-2.0
@rem

@if "%DEBUG%"=="" @echo off
@rem
#####
#####

```

```

@rem
@rem Gradle startup script for Windows
@rem
@rem
#####

#####

@rem Set local scope for the variables with windows NT shell
if "%OS%"=="Windows_NT" setlocal

set DIRNAME=%~dp0
if "%DIRNAME%"=="" set DIRNAME=.
@rem This is normally unused
set APP_BASE_NAME=%~n0
set APP_HOME=%DIRNAME%

@rem Resolve any "." and ".." in APP_HOME to make it shorter.
for %%i in ("%APP_HOME%") do set APP_HOME=%%~fi

@rem Add default JVM options here. You can also use JAVA_OPTS and GRADLE_OPTS to
pass JVM options to this script.
set DEFAULT_JVM_OPTS="-Xmx64m" "-Xms64m"

@rem Find java.exe
if defined JAVA_HOME goto findJavaFromJavaHome

set JAVA_EXE=java.exe
%JAVA_EXE% -version >NUL 2>&1
if %ERRORLEVEL% equ 0 goto execute

echo. 1>&2
echo ERROR: JAVA_HOME is not set and no 'java' command could be found in your PATH.
1>&2
echo. 1>&2
echo Please set the JAVA_HOME variable in your environment to match the 1>&2
echo location of your Java installation. 1>&2

goto fail

:findJavaFromJavaHome
set JAVA_HOME=%JAVA_HOME:"=%
set JAVA_EXE=%JAVA_HOME%/bin/java.exe

```

```
if exist "%JAVA_EXE%" goto execute
```

```
echo. 1>&2
```

```
echo ERROR: JAVA_HOME is set to an invalid directory: %JAVA_HOME% 1>&2
```

```
echo. 1>&2
```

```
echo Please set the JAVA_HOME variable in your environment to match the 1>&2
```

```
echo location of your Java installation. 1>&2
```

```
goto fail
```

```
:execute
```

```
@rem Setup the command line
```

```
set CLASSPATH=
```

```
@rem Execute Gradle
```

```
"%JAVA_EXE%" %DEFAULT_JVM_OPTS% %JAVA_OPTS% %GRADLE_OPTS% "-  
Dorg.gradle.appname=%APP_BASE_NAME%" -classpath "%CLASSPATH%" -jar  
"%APP_HOME%\gradle\wrapper\gradle-wrapper.jar" %*
```

```
:end
```

```
@rem End local scope for the variables with windows NT shell
```

```
if %ERRORLEVEL% equ 0 goto mainEnd
```

```
:fail
```

```
rem Set variable GRADLE_EXIT_CONSOLE if you need the _script_ return code instead of  
rem the _cmd.exe /c_ return code!
```

```
set EXIT_CODE=%ERRORLEVEL%
```

```
if %EXIT_CODE% equ 0 set EXIT_CODE=1
```

```
if not ""=="%GRADLE_EXIT_CONSOLE%" exit %EXIT_CODE%
```

```
exit /b %EXIT_CODE%
```

```
:mainEnd
```

```
if "%OS%"=="Windows_NT" endlocal
```

```
:omega
```

local.properties

```
## This file is automatically generated by Android Studio.
```

```
# Do not modify this file -- YOUR CHANGES WILL BE ERASED!
```

```
#
```

```
# This file should *NOT* be checked into Version Control Systems,  
# as it contains information specific to your local configuration.  
#  
# Location of the SDK. This is only used by Gradle.  
# For customization when using a Version Control System, please read the  
# header note.  
sdk.dir=C:\\Users\\ADMIN\\AppData\\Local\\Android\\Sdk  
CLOUDINARY_CLOUD_NAME=diutnceax  
CLOUDINARY_API_KEY=852589725185877  
CLOUDINARY_API_SECRET=dyYSiYgvUsn3Tx6HWe47RD1WJNA
```

settings.gradle.kts

```
pluginManagement {  
    repositories {  
        google {  
            content {  
                includeGroupByRegex("com\\.\\.android.*")  
                includeGroupByRegex("com\\.\\.google.*")  
                includeGroupByRegex("androidx.*")  
            }  
        }  
        mavenCentral()  
        gradlePluginPortal()  
    }  
}  
dependencyResolutionManagement {  
    repositoriesMode.set(RepositoriesMode.FAIL_ON_PROJECT_REPOS)  
    repositories {  
        google()  
        mavenCentral()  
    }  
}  
  
rootProject.name = "AnTamViecLam"  
include(":app")
```

app/.gitignore

```
/build
```

app/build.gradle.kts

```
// Đảm bảo 2 dòng import này ở đầu file  
import java.util.Properties  
import java.io.FileInputStream
```

```

// ĐOẠN CODE LOGIC MỚI
val localProperties = Properties()
val localPropertiesFile = rootProject.file("local.properties")
if (localPropertiesFile.exists()) {
    localProperties.load(FileInputStream(localPropertiesFile))
}
// Đọc 3 giá trị mới
val cloudName = localProperties.getProperty("CLOUDINARY_CLOUD_NAME") ?: ""
val apiKey = localProperties.getProperty("CLOUDINARY_API_KEY") ?: ""
val apiSecret = localProperties.getProperty("CLOUDINARY_API_SECRET") ?: ""

plugins {
    alias(libs.plugins.android.application)
    alias(libs.plugins.kotlin.android)
    alias(libs.plugins.ksp)
    alias(libs.plugins.firebase.crashlytics)
    alias(libs.plugins.kotlin.compose.compiler)
    alias(libs.plugins.android.secrets.gradle.plugin)
    id("com.google.dagger.hilt.android")
    id("com.google.gms.google-services")
}

android {
    namespace = "com.example.antamvieclam"
    compileSdk = 34

    defaultConfig {
        applicationId = "com.example.antamvieclam"
        minSdk = 24
        targetSdkVersion(34) // Sửa bằng cú pháp gọi hàm
        versionCode = 1
        versionName = "1.0"
        testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner"

        // Chỉ giữ lại dòng buildConfigField ở đây
        buildConfigField("String", "CLOUDINARY_CLOUD_NAME", "\"\$cloudName\"")
        buildConfigField("String", "CLOUDINARY_API_KEY", "\"\$apiKey\"")
        buildConfigField("String", "CLOUDINARY_API_SECRET", "\"\$apiSecret\"") }

    buildTypes {
        release {
            isMinifyEnabled = false
            proguardFiles(

```

```

        getDefaultProguardFile("proguard-android-optimize.txt"),
        "proguard-rules.pro"
    )
}

compileOptions {
    sourceCompatibility = JavaVersion.VERSION_1_8
    targetCompatibility = JavaVersion.VERSION_1_8
}

kotlinOptions {
    jvmTarget = "1.8"
}

buildFeatures {
    compose = true
    buildConfig = true
}

dependencies {

    // Jetpack Compose BoM (Bill of Materials) - Giúp quản lý phiên bản các thư viện
    Compose
    val composeBom = platform("androidx.compose:compose-bom:2024.05.00") // Kiểm tra
    phiên bản mới nhất
    implementation(composeBom)
    androidTestImplementation(composeBom)

    // Các thư viện Compose cần thiết
    implementation("androidx.compose.ui:ui")
    implementation("androidx.compose.ui:ui-graphics")
    implementation("androidx.compose.ui:ui-tooling-preview")
    implementation("androidx.compose.material3:material3")

    // Thư viện cần thiết cho setContent
    implementation("androidx.activity:activity-compose:1.9.0") // Kiểm tra phiên bản mới
    nhất

```

```
// Phần này của bạn đã đúng, giữ nguyên
implementation(libs.cloudinary.android)
implementation(libs.androidx.core.ktx)
implementation(platform(libs.androidx.compose.bom))
implementation(libs.androidx.compose.ui)
implementation(libs.androidx.compose.ui.graphics)
implementation(libs.androidx.compose.ui.tooling.preview)
implementation(libs.androidx.compose.material3)
implementation(libs.hilt.android)
ksp(libs.hilt.compiler)
implementation(libs.androidx.room.runtime)
implementation(libs.androidx.room.ktx)
ksp(libs.androidx.room.compiler)
implementation(platform(libs.firebase.bom))
implementation(libs.firebase.auth.ktx)
implementation(libs.firebase.firestore.ktx)
implementation(libs.firebase.storage.ktx)
implementation(libs.firebase.messaging.ktx)
implementation(libs.firebase.crashlytics.ktx)
implementation(libs.play.services.auth)
implementation(libs.kotlinx.coroutines.core)
implementation(libs.kotlinx.coroutines.android)
implementation(libs.retrofit)
implementation(libs.converter.gson)
implementation(libs.logging.interceptor)
testImplementation(libs.junit)
androidTestImplementation(libs.androidx.junit)
androidTestImplementation(libs.androidx.espresso.core)
androidTestImplementation(platform(libs.androidx.compose.bom))
debugImplementation(libs.androidx.compose.ui.tooling)
}

ksp {
    arg("hilt.CorrectErrorTypes", "true")
}
```

app/google-services.json

```
{
  "project_info": {
    "project_number": "221272132411",
    "project_id": "antamvieclam",
    "storage_bucket": "antamvieclam.firebaseioapp"
  },
```

```

"client": [
  {
    "client_info": {
      "mobilesdk_app_id": "1:221272132411:android:c432fab2ad267434c8451e",
      "android_client_info": {
        "package_name": "com.example.antamvieclam"
      }
    },
    "oauth_client": [],
    "api_key": [
      {
        "current_key": "AlzaSyCuyB9Bi3hWhR3JCuaZ4kZPq1NZ8u6-vEs"
      }
    ],
    "services": {
      "appinvite_service": {
        "other_platform_oauth_client": []
      }
    }
  }
],
"configuration_version": "1"
}

```

app/proguard-rules.pro

```

# Add project specific ProGuard rules here.
# You can control the set of applied configuration files using the
# proguardFiles setting in build.gradle.
#
# For more details, see
# http://developer.android.com/guide/developing/tools/proguard.html

# If your project uses WebView with JS, uncomment the following
# and specify the fully qualified class name to the JavaScript interface
# class:
#-keepclassmembers class fqcn.of.javascript.interface.for.webview {
#   public *;
#}

# Uncomment this to preserve the line number information for
# debugging stack traces.
#-keepattributes SourceFile,LineNumberTable

```

```
# If you keep the line number information, uncomment this to
# hide the original source file name.
#-renamesourcefileattribute SourceFile
```

app/src/androidTest/java/com/example/antamvieclam/ExampleInstrumentedTest.kt

```
package com.example.antamvieclam
```

```
import androidx.test.platform.app.InstrumentationRegistry
import androidx.test.ext.junit.runners.AndroidJUnit4
```

```
import org.junit.Test
import org.junit.runner.RunWith
```

```
import org.junit.Assert.*
```

```
/**
 * Instrumented test, which will execute on an Android device.
 *
 * See [testing documentation](http://d.android.com/tools/testing).
 */
@RunWith(AndroidJUnit4::class)
class ExampleInstrumentedTest {
    @Test
    fun useAppContext() {
        // Context of the app under test.
        val appContext = InstrumentationRegistry.getInstrumentation().targetContext
        assertEquals("com.example.antamvieclam", appContext.packageName)
    }
}
```

app/src/main/AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:name="com.example.antamvieclam.AnVuiViecApplication"
        android:label="@string/app_name"
```

```

        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.AnTamViecLam">
        <activity
            android:name="com.example.antamvieclam.MainActivity"
            android:exported="true"
            android:label="@string/app_name"
            android:theme="@style/Theme.AnTamViecLam">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>

```

[app/src/main/java/com/example/antamvieclam/AnVuiViecApplication.kt](#)

```

package com.example.antamvieclam

import android.app.Application
import com.cloudinary.android.MediaManager
import dagger.hilt.android.HiltAndroidApp
import com.example.antamvieclam.BuildConfig

@HiltAndroidApp
class AnVuiViecApplication : Application() {

    override fun onCreate() {
        super.onCreate()

        // Tạo một Map để chứa thông tin cấu hình
        val config = mutableMapOf<String, String>()
        config["cloud_name"] = BuildConfig.CLOUDINARY_CLOUD_NAME
        config["api_key"] = BuildConfig.CLOUDINARY_API_KEY
        config["api_secret"] = BuildConfig.CLOUDINARY_API_SECRET

        // Gọi hàm init với Context và Map, đây là phiên bản hợp lệ
        MediaManager.init(this, config)
    }
}

```

[app/src/main/java/com/example/antamvieclam/MainActivity.kt](#)

```
package com.example.antamvieclam
```

```
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
import androidx.activity.enableEdgeToEdge
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.padding
import androidx.compose.material3.Scaffold
import androidx.compose.material3.Text
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.tooling.preview.Preview
import com.example.antamvieclam.ui.theme.AnTamViecLamTheme
```

```
class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContent {
            AnTamViecLamTheme {
                Scaffold(modifier = Modifier.fillMaxSize()) { innerPadding ->
                    Greeting(
                        name = "Android",
                        modifier = Modifier.padding(innerPadding)
                    )
                }
            }
        }
    }
}
```

```
@Composable
fun Greeting(name: String, modifier: Modifier = Modifier) {
    Text(
        text = "Hello $name!",
        modifier = modifier
    )
}
```

```
@Preview(showBackground = true)
@Composable
```

```

fun GreetingPreview() {
    AnTamViecLamTheme {
        Greeting("Android")
    }
}

```

[app/src/main/java/com/example/antamvieclam/di/AppModule.kt](#)

```

package com.example.antamvieclam.di

```

```

import com.cloudinary.android.MediaManager
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore
import dagger.Module
import dagger.Provides
import dagger.hilt.InstallIn
import dagger.hilt.components.SingletonComponent
import javax.inject.Singleton

```

```

@Module

```

```

@InstallIn(SingletonComponent::class) // Dependencies sẽ sống sót suốt vòng đời ứng dụng

```

```

object AppModule {

```

```

    @Provides

```

```

    @Singleton // Chỉ tạo một instance duy nhất

```

```

    fun provideFirebaseAuth(): FirebaseAuth {

```

```

        return FirebaseAuth.getInstance()

```

```

    }

```

```

    @Provides

```

```

    @Singleton

```

```

    fun provideFirestore(): FirebaseFirestore {

```

```

        return FirebaseFirestore.getInstance()

```

```

    }

```

```

    @Provides

```

```

    @Singleton

```

```

    fun provideCloudinary(): MediaManager {

```

```

        // Vì MediaManager đã được khởi tạo trong Application class,

```

```

        // ở đây chúng ta chỉ cần lấy ra instance đã tồn tại đó.

```

```

        return MediaManager.get()

```

```

    }

```

```

}

```

[app/src/main/java/com/example/antamvieclam/ui/theme/Color.kt](#)

```
package com.example.antamvieclam.ui.theme
```

```
import androidx.compose.ui.graphics.Color
```

```
val Purple80 = Color(0xFFD0BCFF)  
val PurpleGrey80 = Color(0xFFCCC2DC)  
val Pink80 = Color(0xFFE8B8C8)
```

```
val Purple40 = Color(0xFF6650a4)  
val PurpleGrey40 = Color(0xFF625b71)  
val Pink40 = Color(0xFF7D5260)
```

[app/src/main/java/com/example/antamvieclam/ui/theme/Theme.kt](#)

```
package com.example.antamvieclam.ui.theme
```

```
import android.app.Activity  
import android.os.Build  
import androidx.compose.foundation.isSystemInDarkTheme  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.darkColorScheme  
import androidx.compose.material3.dynamicDarkColorScheme  
import androidx.compose.material3.dynamicLightColorScheme  
import androidx.compose.material3.lightColorScheme  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.platform.LocalContext
```

```
private val DarkColorScheme = darkColorScheme(  
    primary = Purple80,  
    secondary = PurpleGrey80,  
    tertiary = Pink80  
)
```

```
private val LightColorScheme = lightColorScheme(  
    primary = Purple40,  
    secondary = PurpleGrey40,  
    tertiary = Pink40
```

```
/* Other default colors to override  
background = Color(0xFFFFFBFE),  
surface = Color(0xFFFFFBFE),  
onPrimary = Color.White,  
onSecondary = Color.White,
```

```

        onTertiary = Color.White,
        onBackground = Color(0xFF1C1B1F),
        onSurface = Color(0xFF1C1B1F),
        */
    )

@Composable
fun AnTamViecLamTheme(
    darkTheme: Boolean = isSystemInDarkTheme(),
    // Dynamic color is available on Android 12+
    dynamicColor: Boolean = true,
    content: @Composable () -> Unit
) {
    val colorScheme = when {
        dynamicColor && Build.VERSION.SDK_INT >= Build.VERSION_CODES.S -> {
            val context = LocalContext.current
            if (darkTheme) dynamicDarkColorScheme(context) else
dynamicLightColorScheme(context)
        }

        darkTheme -> DarkColorScheme
        else -> LightColorScheme
    }

    MaterialTheme(
        colorScheme = colorScheme,
        typography = Typography,
        content = content
    )
}

```

<app/src/main/java/com/example/antamvieclam/ui/theme/Type.kt>

```
package com.example.antamvieclam.ui.theme
```

```

import androidx.compose.material3.Typography
import androidx.compose.ui.text.TextStyle
import androidx.compose.ui.text.font.FontFamily
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.unit.sp

```

```

// Set of Material typography styles to start with
val Typography = Typography(
    bodyLarge = TextStyle(

```

```

        fontFamily = FontFamily.Default,
        fontWeight = FontWeight.Normal,
        fontSize = 16.sp,
        lineHeight = 24.sp,
        letterSpacing = 0.5.sp
    )
    /* Other default text styles to override
    titleLarge = TextStyle(
        fontFamily = FontFamily.Default,
        fontWeight = FontWeight.Normal,
        fontSize = 22.sp,
        lineHeight = 28.sp,
        letterSpacing = 0.sp
    ),
    labelSmall = TextStyle(
        fontFamily = FontFamily.Default,
        fontWeight = FontWeight.Medium,
        fontSize = 11.sp,
        lineHeight = 16.sp,
        letterSpacing = 0.5.sp
    )
    */
}

```

[app/src/main/res/drawable/ic_launcher_background.xml](#)

```

<?xml version="1.0" encoding="utf-8"?>
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="108dp"
    android:height="108dp"
    android:viewportWidth="108"
    android:viewportHeight="108">
    <path
        android:fillColor="#3DDC84"
        android:pathData="M0,0h108v108h-108z" />
    <path
        android:fillColor="#00000000"
        android:pathData="M9,0L9,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M19,0L19,108"
        android:strokeWidth="0.8"

```

```

        android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M29,0L29,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M39,0L39,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M49,0L49,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M59,0L59,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M69,0L69,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M79,0L79,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M89,0L89,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M99,0L99,108"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"

```

```
        android:pathData="M0,9L108,9"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,19L108,19"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,29L108,29"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,39L108,39"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,49L108,49"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,59L108,59"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,69L108,69"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,79L108,79"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,89L108,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
```

```
<path
  android:fillColor="#00000000"
  android:pathData="M0,99L108,99"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,29L89,29"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,39L89,39"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,49L89,49"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,59L89,59"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,69L89,69"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M19,79L89,79"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M29,19L29,89"
  android:strokeWidth="0.8"
  android:strokeColor="#33FFFFFF" />
<path
  android:fillColor="#00000000"
  android:pathData="M39,19L39,89"
```

```

        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M49,19L49,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M59,19L59,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M69,19L69,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M79,19L79,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
</vector>

```

[app/src/main/res/drawable/ic_launcher_foreground.xml](#)

```

<vector xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:aapt="http://schemas.android.com/aapt"
    android:width="108dp"
    android:height="108dp"
    android:viewportWidth="108"
    android:viewportHeight="108">
    <path android:pathData="M31,63.928c0,0 6.4,-11 12.1,-13.1c7.2,-2.6 26,-1.4 26,-1.4l38.1,38.1L107,108.928l-32,-1L31,63.928z">
        <aapt:attr name="android:fillColor">
            <gradient
                android:endX="85.84757"
                android:endY="92.4963"
                android:startX="42.9492"
                android:startY="49.59793"
                android:type="linear">
                <item
                    android:color="#44000000"

```

```

        android:offset="0.0" />
    <item
        android:color="#00000000"
        android:offset="1.0" />
    </gradient>
</aapt:attr>
</path>
<path
    android:fillColor="#FFFFFF"
    android:fillType="nonZero"
    android:pathData="M65.3,45.828l3.8,-6.6c0.2,-0.4 0.1,-0.9 -0.3,-1.1c-0.4,-0.2 -0.9,-0.1 -
1.1,0.3l-3.9,6.7c-6.3,-2.8 -13.4,-2.8 -19.7,0l-3.9,-6.7c-0.2,-0.4 -0.7,-0.5 -1.1,-0.3C38.8,38.328
38.7,38.828 38.9,39.228l3.8,6.6C36.2,49.428 31.7,56.028 31,63.928h46C76.3,56.028
71.8,49.428 65.3,45.828zM43.4,57.328c-0.8,0 -1.5,-0.5 -1.8,-1.2c-0.3,-0.7 -0.1,-1.5 0.4,-
2.1c0.5,-0.5 1.4,-0.7 2.1,-0.4c0.7,0.3 1.2,1 1.2,1.8C45.3,56.528 44.5,57.328
43.4,57.328L43.4,57.328zM64.6,57.328c-0.8,0 -1.5,-0.5 -1.8,-1.2s-0.1,-1.5 0.4,-2.1c0.5,-0.5
1.4,-0.7 2.1,-0.4c0.7,0.3 1.2,1 1.2,1.8C66.5,56.528 65.6,57.328 64.6,57.328L64.6,57.328z"
    android:strokeWidth="1"
    android:strokeColor="#00000000" />
</vector>

```

[app/src/main/res/mipmap-anydpi-v26/ic_launcher.xml](#)

```

<?xml version="1.0" encoding="utf-8"?>
<adaptive-icon xmlns:android="http://schemas.android.com/apk/res/android">
    <background android:drawable="@drawable/ic_launcher_background" />
    <foreground android:drawable="@drawable/ic_launcher_foreground" />
    <monochrome android:drawable="@drawable/ic_launcher_foreground" />
</adaptive-icon>

```

[app/src/main/res/mipmap-anydpi-v26/ic_launcher_round.xml](#)

```

<?xml version="1.0" encoding="utf-8"?>
<adaptive-icon xmlns:android="http://schemas.android.com/apk/res/android">
    <background android:drawable="@drawable/ic_launcher_background" />
    <foreground android:drawable="@drawable/ic_launcher_foreground" />
    <monochrome android:drawable="@drawable/ic_launcher_foreground" />
</adaptive-icon>

```

[app/src/main/res/values/colors.xml](#)

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>

```

```

    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>

```

app/src/main/res/values/strings.xml

```

<resources>
    <string name="app_name">AnTamViecLam</string>
</resources>

```

app/src/main/res/values/themes.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>

    <style name="Theme.AnTamViecLam"
parent="android:Theme.Material.Light.NoActionBar" />
</resources>

```

app/src/main/res/xml/backup_rules.xml

```

<?xml version="1.0" encoding="utf-8"?><!--
    Sample backup rules file; uncomment and customize as necessary.
    See https://developer.android.com/guide/topics/data/autobackup
    for details.
    Note: This file is ignored for devices older than API 31
    See https://developer.android.com/about/versions/12/backup-restore
-->
<full-backup-content>
    <!--
    <include domain="sharedpref" path="."/>
    <exclude domain="sharedpref" path="device.xml"/>
    -->
</full-backup-content>

```

app/src/main/res/xml/data_extraction_rules.xml

```

<?xml version="1.0" encoding="utf-8"?><!--
    Sample data extraction rules file; uncomment and customize as necessary.
    See https://developer.android.com/about/versions/12/backup-restore#xml-changes
    for details.
-->
<data-extraction-rules>
    <cloud-backup>
        <!-- TODO: Use <include> and <exclude> to control what is backed up.
        <include .../>
        <exclude .../>

```

```

-->
</cloud-backup>
<!--
<device-transfer>
  <include .../>
  <exclude .../>
</device-transfer>
-->
</data-extraction-rules>

```

[app/src/test/java/com/example/antamvieclam/ExampleUnitTest.kt](#)

```
package com.example.antamvieclam
```

```
import org.junit.Test
```

```
import org.junit.Assert.*
```

```

/**
 * Example local unit test, which will execute on the development machine (host).
 *
 * See [testing documentation](http://d.android.com/tools/testing).
 */
class ExampleUnitTest {
    @Test
    fun addition_isCorrect() {
        assertEquals(4, 2 + 2)
    }
}

```

[gradle/libs.versions.toml](#)

```
# gradle/libs.versions.toml
```

```

[versions]
# Plugins
crashlyticsPlugin = "3.0.1"
androidGradlePlugin = "8.4.1"
kotlin = "2.0.0"
ksp = "2.0.0-1.0.21"
googleServices = "4.4.2"
mapsSecretsPlugin = "2.0.1"
materialIconsExtended = "1.6.8"
trackasia = "2.0.2"
accompanist = "0.32.0"
secrets-gradle-plugin = "2.0.1"

```

cloudinary = "2.4.0"

SDKs

compileSdk = "34"

minSdk = "24"

targetSdk = "34"

THÊM CÁC THƯ VIỆN TRACKASIA

trackasia-sdk = { group = "com.track-asia", name = "trackasia-android-sdk", version.ref = "trackasia" }

trackasia-annotation-plugin = { group = "com.track-asia", name = "trackasia-android-plugin-annotation-v9", version.ref = "trackasia" }

Libraries

coreKtx = "1.13.1"

activityCompose = "1.9.0"

composeBom = "2024.06.00"

composeCompiler = "1.5.14"

hilt = "2.51.1"

lifecycle = "2.8.2"

room = "2.6.1"

firebaseBom = "33.1.1"

playServicesAuth = "21.2.0"

playServicesCoroutines = "1.8.1"

coroutines = "1.8.0"

retrofit = "2.9.0"

okhttp = "4.12.0"

navigation = "2.7.7"

hiltNavigation = "1.2.0"

coil = "2.6.0"

mapsCompose = "4.3.3" # <--- THÊM MỚI: Thư viện Maps cho Compose

playServicesMaps = "18.2.0" # <--- THÊM MỚI: Thư viện Google Maps SDK

playServicesLocation = "21.3.0" # <--- THÊM MỚI: Thư viện Location Services

Testing

junit = "4.13.2"

androidxJunit = "1.1.5"

espressoCore = "3.5.1"

firebaseAuth = "23.0.0"

viewbinding = "7.4.2"

transportBackendCct = "3.1.9"

transportApi = "3.0.0"

[libraries]

AndroidX Core

androidx-core-ktx = { group = "androidx.core", name = "core-ktx", version.ref = "coreKtx" }

androidx-lifecycle-runtime-ktx = { group = "androidx.lifecycle", name = "lifecycle-runtime-ktx", version.ref = "lifecycle" }

androidx-activity-compose = { group = "androidx.activity", name = "activity-compose", version.ref = "activityCompose" }

accompanist-navigation-animation = { group = "com.google.accompanist", name = "accompanist-navigation-animation", version.ref = "accompanist" }

Jetpack Compose

androidx-compose-bom = { group = "androidx.compose", name = "compose-bom", version.ref = "composeBom" }

androidx-compose-ui = { group = "androidx.compose.ui", name = "ui" }

androidx-compose-ui-graphics = { group = "androidx.compose.ui", name = "ui-graphics" }

androidx-compose-ui-tooling = { group = "androidx.compose.ui", name = "ui-tooling" }

androidx-compose-ui-tooling-preview = { group = "androidx.compose.ui", name = "ui-tooling-preview" }

androidx-compose-material3 = { group = "androidx.compose.material3", name = "material3" }

androidx-lifecycle-viewmodel-compose = { group = "androidx.lifecycle", name = "lifecycle-viewmodel-compose", version.ref = "lifecycle" }

androidx-lifecycle-runtime-compose = { group = "androidx.lifecycle", name = "lifecycle-runtime-compose", version.ref = "lifecycle" }

androidx-compose-material-icons-extended = { group = "androidx.compose.material", name = "material-icons-extended", version.ref = "materialIconsExtended" }

Image Loading - Coil

coil-compose = { module = "io.coil-kt:coil-compose", version.ref = "coil" }

Navigation

androidx-navigation-compose = { group = "androidx.navigation", name = "navigation-compose", version.ref = "navigation" }

androidx-hilt-navigation-compose = { group = "androidx.hilt", name = "hilt-navigation-compose", version.ref = "hiltNavigation" }

Dependency Injection - Hilt

hilt-android = { group = "com.google.dagger", name = "hilt-android", version.ref = "hilt" }

hilt-compiler = { group = "com.google.dagger", name = "hilt-compiler", version.ref = "hilt" }

Local Database - Room

```
androidx-room-runtime = { group = "androidx.room", name = "room-runtime", version.ref = "room" }
androidx-room-ktx = { group = "androidx.room", name = "room-ktx", version.ref = "room" }
androidx-room-compiler = { group = "androidx.room", name = "room-compiler", version.ref = "room" }
```

Remote - Firebase

```
cloudinary-android = { group = "com.cloudinary", name = "cloudinary-android", version.ref = "cloudinary" }
firebase-bom = { group = "com.google.firebase", name = "firebase-bom", version.ref = "firebaseBom" }
firebase-auth-ktx = { group = "com.google.firebase", name = "firebase-auth-ktx" }
firebase-firestore-ktx = { group = "com.google.firebase", name = "firebase-firestore-ktx" }
firebase-storage-ktx = { group = "com.google.firebase", name = "firebase-storage-ktx" }
firebase-messaging-ktx = { group = "com.google.firebase", name = "firebase-messaging-ktx" }
firebase-crashlytics-ktx = { group = "com.google.firebase", name = "firebase-crashlytics-ktx" }
play-services-auth = { group = "com.google.android.gms", name = "play-services-auth", version.ref = "playServicesAuth" }
```

THÊM MÓI: Google Maps & Location

```
maps-compose = { group = "com.google.maps.android", name = "maps-compose", version.ref = "mapsCompose" }
play-services-maps = { group = "com.google.android.gms", name = "play-services-maps", version.ref = "playServicesMaps" }
play-services-location = { group = "com.google.android.gms", name = "play-services-location", version.ref = "playServicesLocation" }
trackasia-sdk = { group = "io.github.track-asia", name = "android-sdk", version.ref = "trackasia" }
trackasia-annotation-plugin = { group = "io.github.track-asia", name = "android-plugin-annotation-v9", version = "2.0.1" }
```

Asynchronous - Coroutines

```
kotlinx-coroutines-core = { group = "org.jetbrains.kotlinx", name = "kotlinx-coroutines-core", version.ref = "coroutines" }
kotlinx-coroutines-android = { group = "org.jetbrains.kotlinx", name = "kotlinx-coroutines-android", version.ref = "coroutines" }
kotlinx-coroutines-play-services = { group = "org.jetbrains.kotlinx", name = "kotlinx-coroutines-play-services", version.ref = "playServicesCoroutines" }
```

Networking - Retrofit & OkHttp

```
retrofit = { group = "com.squareup.retrofit2", name = "retrofit", version.ref = "retrofit" }
converter-gson = { group = "com.squareup.retrofit2", name = "converter-gson", version.ref = "retrofit" }
logging-interceptor = { group = "com.squareup.okhttp3", name = "logging-interceptor",
version.ref = "okhttp" }
```

Testing

```
junit = { group = "junit", name = "junit", version.ref = "junit" }
androidx-junit = { group = "androidx.test.ext", name = "junit", version.ref = "androidxJunit" }
androidx-espresso-core = { group = "androidx.test.espresso", name = "espresso-core",
version.ref = "espressoCore" }
androidx-compose-ui-test-junit4 = { group = "androidx.compose.ui", name = "ui-test-junit4"
}
androidx-compose-ui-test-manifest = { group = "androidx.compose.ui", name = "ui-test-
manifest" }
firebase-auth = { group = "com.google.firebase", name = "firebase-auth", version.ref =
"firebaseAuth" }
androidx-viewbinding = { group = "androidx.databinding", name = "viewbinding",
version.ref = "viewbinding" }
transport-backend-cct = { group = "com.google.android.datatransport", name = "transport-
backend-cct", version.ref = "transportBackendCct" }
transport-api = { group = "com.google.android.datatransport", name = "transport-api",
version.ref = "transportApi" }
```

[plugins]

Khai báo các plugin của dự án

```
android-secrets-gradle-plugin = { id = "com.google.android.libraries.mapsplatform.secrets-
gradle-plugin", version.ref = "secrets-gradle-plugin" }
android-application = { id = "com.android.application", version.ref = "androidGradlePlugin"
}
android-library = { id = "com.android.library", version.ref = "androidGradlePlugin" }
kotlin-android = { id = "org.jetbrains.kotlin.android", version.ref = "kotlin" }
ksp = { id = "com.google.devtools.ksp", version.ref = "ksp" }
hilt = { id = "com.google.dagger.hilt.android", version.ref = "hilt" }
google-services = { id = "com.google.gms.google-services", version.ref = "googleServices" }
kotlin-compose-compiler = { id = "org.jetbrains.kotlin.plugin.compose", version.ref =
"kotlin" }
firebase-crashlytics = { id = "com.google.firebase.crashlytics", version.ref =
"crashlyticsPlugin" }
maps-secrets = { id = "com.google.android.libraries.mapsplatform.secrets-gradle-plugin",
version.ref = "mapsSecretsPlugin" } # <--- THÊM MỚI
```

[bundles]

Nhóm các thư viện thường đi chung với nhau để gọi cho gọn

compose = ["androidx-compose-ui", "androidx-compose-ui-graphics", "androidx-compose-ui-tooling-preview", "androidx-compose-material3"]

room = ["androidx-room-runtime", "androidx-room-ktx"]

coroutines = ["kotlinx-coroutines-core", "kotlinx-coroutines-android"]

[gradle/wrapper/gradle-wrapper.properties](#)

#Sun Nov 09 13:27:50 ICT 2025

distributionBase=GRADLE_USER_HOME

distributionPath=wrapper/dists

distributionUrl=https\://services.gradle.org/distributions/gradle-8.13-bin.zip

networkTimeout=10000

validateDistributionUrl=true

zipStoreBase=GRADLE_USER_HOME

zipStorePath=wrapper/dists