SOURCE CODE

=== File: build.gradle (Project level) ===

```
// Top-level build file
buildscript {
    ext.kotlin_version = '1.8.21'
    repositories {
        google()
        mavenCentral()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:8.1.0'
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"
    }
}

allprojects {
    repositories {
        google()
        mavenCentral()
    }
}

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

=== File: app/build.gradle (Module: app) ===

```
plugins {
    id 'com.android.application'
    id 'org.jetbrains.kotlin.android'
    id 'kotlin-kapt'
}

android {
    namespace 'com.example.digitalwardrobe'
    compileSdk 34

    defaultConfig {
        applicationId "com.example.digitalwardrobe"
        minSdk 24
        targetSdk 34
        versionCode 1
        versionName "1.0"
```

```
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'),
'proguard-rules.pro'
        }
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_17
        targetCompatibility JavaVersion.VERSION_17
    }
    kotlinOptions {
        jvmTarget = '17'
    }
}

dependencies {
    implementation "org.jetbrains.kotlin:kotlin-stdlib:1.8.21"
    implementation 'androidx.core:core-ktx:1.10.1'
    implementation 'androidx.appcompat:appcompat:1.6.1'
    implementation 'com.google.android.material:material:1.9.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'

    // Room
    implementation 'androidx.room:room-runtime:2.5.2'
    kapt 'androidx.room:room-compiler:2.5.2'
    implementation 'androidx.room:room-ktx:2.5.2'

    // Lifecycle / coroutines
    implementation 'androidx.lifecycle:lifecycle-runtime-ktx:2.6.1'
    implementation 'org.jetbrains.kotlinx:kotlinx-coroutines-android:1.7.3'

    // RecyclerView
    implementation 'androidx.recyclerview:recyclerview:1.3.1'

    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.5'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'
}
```

=== File: AndroidManifest.xml ===

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

    <uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE" />

    <application
        android:allowBackup="true"
        android:label="Digital Wardrobe"
        android:icon="@mipmap/ic_launcher"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.DigitalWardrobe">
        <activity android:name="com.example.digitalwardrobe.SuggestionsActivity" />
        <activity android:name="com.example.digitalwardrobe.AnalyticsActivity" />
        <activity android:name="com.example.digitalwardrobe.AddClothingActivity" />
        <activity android:name="com.example.digitalwardrobe.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

=== File: /res/values/strings.xml ===

```
<resources>
    <string name="app_name">Digital Wardrobe</string>
</resources>
```

=== File: /res/values/themes.xml ===

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <style name="Theme.DigitalWardrobe"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <item name="colorPrimary">@color/purple_500</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@android:color/white</item>
    </style>
</resources>
```

=== File: /res/layout/activity_main.xml ===

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/tvTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Digital Wardrobe"
        android:textSize="24sp"
        android:layout_gravity="center_horizontal"
        android:padding="8dp" />

    <Button
        android:id="@+id/btnAdd"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Add Clothing"/>

    <Button
        android:id="@+id/btnView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="View Wardrobe"
        android:layout_marginTop="8dp"/>

    <Button
        android:id="@+id/btnAnalytics"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Analytics"
        android:layout_marginTop="8dp"/>

    <Button
        android:id="@+id/btnSuggest"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Get Outfit Suggestion"
```

```
        android:layout_marginTop="8dp"/>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/rvWardrobe"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:layout_marginTop="16dp"/>

</LinearLayout>
```

=== File: /res/layout/item_clothing.xml ===

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="8dp">

    <ImageView
        android:id="@+id/imgItem"
        android:layout_width="64dp"
        android:layout_height="64dp"
        android:scaleType="centerCrop"
        android:src="@mipmap/ic_launcher" />

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:orientation="vertical"
        android:paddingStart="8dp">

        <TextView
            android:id="@+id/tvName"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Name"/>

        <TextView
            android:id="@+id/tvDetails"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Details"/>
```

```
    </LinearLayout>

    <Button
        android:id="@+id/btnWear"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Wear" />

</LinearLayout>
```

=== File: /res/layout/activity_add_clothing.xml ===

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="16dp">

        <EditText
            android:id="@+id/etName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Name"/>

        <EditText
            android:id="@+id/etType"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Type (Top/Bottom/Outer)"/>

        <EditText
            android:id="@+id/etColor"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Color"/>

        <EditText
            android:id="@+id/etSeason"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
```

```
        android:hint="Season (Summer/Winter)"/>

    <ImageView
        android:id="@+id/imgPreview"
        android:layout_width="120dp"
        android:layout_height="120dp"
        android:layout_marginTop="8dp"
        android:src="@mipmap/ic_launcher"/>

    <Button
        android:id="@+id/btnPick"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Pick Image"/>

    <Button
        android:id="@+id/btnSave"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Save"
        android:layout_marginTop="12dp"/>

  </LinearLayout>
</ScrollView>
```

=== File: /src/main/java/com/example/digitalwardrobe/model/Clothing.kt ===

```
package com.example.digitalwardrobe.model

import androidx.room.Entity
import androidx.room.PrimaryKey

@Entity(tableName = "clothes")
data class Clothing(
  @PrimaryKey(autoGenerate = true) val id: Int = 0,
  val name: String,
  val type: String,
  val color: String,
  val season: String,
  val imageUri: String,
  var usageCount: Int = 0
)
```

=== File: /src/main/java/com/example/digitalwardrobe/db/ClothingDao.kt ===

```
package com.example.digitalwardrobe.db
```

```
import androidx.room.*
import com.example.digitalwardrobe.model.Clothing

@Dao
interface ClothingDao {
    @Insert
    suspend fun insert(clothing: Clothing): Long

    @Query("SELECT * FROM clothes ORDER BY id DESC")
    suspend fun getAll(): List<Clothing>

    @Update
    suspend fun update(clothing: Clothing)

    @Delete
    suspend fun delete(clothing: Clothing)

    @Query("UPDATE clothes SET usageCount = usageCount + 1 WHERE id = :id")
    suspend fun incrementUsage(id: Int)

    @Query("SELECT * FROM clothes WHERE id = :id")
    suspend fun getById(id: Int): Clothing?
}
```

=== File: /src/main/java/com/example/digitalwardrobe/db/AppDatabase.kt ===

```
package com.example.digitalwardrobe.db

import android.content.Context
import androidx.room.Database
import androidx.room.Room
import androidx.room.RoomDatabase
import com.example.digitalwardrobe.model.Clothing

@Database(entities = [Clothing::class], version = 1)
abstract class AppDatabase : RoomDatabase() {
    abstract fun clothingDao(): ClothingDao

    companion object {
        @Volatile
        private var INSTANCE: AppDatabase? = null

        fun getDatabase(context: Context): AppDatabase {
            return INSTANCE ?: synchronized(this) {
```

```
            val instance = Room.databaseBuilder(
                context.applicationContext,
                AppDatabase::class.java,
                "digital_wardrobe_db"
            ).build()
            INSTANCE = instance
            instance
        }
    }
}
```

=== File: /src/main/java/com/example/digitalwardrobe/MainActivity.kt ===

```
package com.example.digitalwardrobe

import android.app.Activity
import android.content.Intent
import android.net.Uri
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import androidx.recyclerview.widget.LinearLayoutManager
import com.example.digitalwardrobe.db.AppDatabase
import com.example.digitalwardrobe.model.Clothing
import kotlinx.android.synthetic.main.activity_main.*
import kotlinx.coroutines.launch

class MainActivity : AppCompatActivity() {

    private lateinit var db: AppDatabase
    private lateinit var adapter: ClothingAdapter
    private var clothes = mutableListOf<Clothing>()

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        db = AppDatabase.getDatabase(this)

        adapter = ClothingAdapter(clothes) { clothing ->
            // onWear clicked
            lifecycleScope.launch {
                db.clothingDao().incrementUsage(clothing.id)
                loadClothes()
```

```
            }
        }

        rvWardrobe.layoutManager = LinearLayoutManager(this)
        rvWardrobe.adapter = adapter

        btnAdd.setOnClickListener {
            startActivity(Intent(this, AddClothingActivity::class.java))
        }
        btnView.setOnClickListener {
            loadClothes()
        }
        btnAnalytics.setOnClickListener {
            startActivity(Intent(this, AnalyticsActivity::class.java))
        }
        btnSuggest.setOnClickListener {
            startActivity(Intent(this, SuggestionsActivity::class.java))
        }

        // initial load
        loadClothes()
    }

    override fun onResume() {
        super.onResume()
        loadClothes()
    }

    private fun loadClothes() {
        lifecycleScope.launch {
            val list = db.clothingDao().getAll()
            clothes.clear()
            clothes.addAll(list)
            adapter.notifyDataSetChanged()
        }
    }
}
```

=== File: /src/main/java/com/example/digitalwardrobe/ClothingAdapter.kt ===

```
package com.example.digitalwardrobe

import android.net.Uri
import android.view.LayoutInflater
import android.view.View
```

```
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.example.digitalwardrobe.model.Clothing
import kotlinx.android.synthetic.main.item_clothing.view.*

class ClothingAdapter(
    private val items: List<Clothing>,
    private val onWearClicked: (Clothing) -> Unit
) : RecyclerView.Adapter<ClothingAdapter.VH>() {

    inner class VH(view: View) : RecyclerView.ViewHolder(view) {
        val img: ImageView = view.imgItem
        val name: TextView = view.tvName
        val details: TextView = view.tvDetails
        val btnWear = view.btnWear
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): VH {
        val v = LayoutInflater.from(parent.context).inflate(R.layout.item_clothing,
parent, false)
        return VH(v)
    }

    override fun onBindViewHolder(holder: VH, position: Int) {
        val item = items[position]
        holder.name.text = item.name
        holder.details.text = "${item.type} • ${item.color} • Worn: ${item.usageCount}"
        try {
            if (item.imageUri.isNotEmpty())
holder.img.setImageURI(Uri.parse(item.imageUri))
        } catch (e: Exception) { /* ignore */ }

        holder.btnWear.setOnClickListener { onWearClicked(item) }
    }

    override fun getItemCount(): Int = items.size
}
```

=== File: /src/main/java/com/example/digitalwardrobe/AddClothingActivity.kt ===

```
package com.example.digitalwardrobe
```

```
import android.app.Activity
import android.content.Intent
import android.net.Uri
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import com.example.digitalwardrobe.db.AppDatabase
import com.example.digitalwardrobe.model.Clothing
import kotlinx.android.synthetic.main.activity_add_clothing.*
import kotlinx.coroutines.launch

class AddClothingActivity : AppCompatActivity() {

    private val PICK_IMAGE = 1001
    private var selectedUri: Uri? = null
    private lateinit var db: AppDatabase

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_add_clothing)

        db = AppDatabase.getDatabase(this)

        btnPick.setOnClickListener {
            val i = Intent(Intent.ACTION_PICK)
            i.type = "image/*"
            startActivityForResult(i, PICK_IMAGE)
        }

        btnSave.setOnClickListener {
            val name = etName.text.toString().trim()
            val type = etType.text.toString().trim()
            val color = etColor.text.toString().trim()
            val season = etSeason.text.toString().trim()
            val uriStr = selectedUri?.toString() ?: ""

            if (name.isEmpty() || type.isEmpty()) {
                etName.error = "Enter name"
                return@setOnClickListener
            }

            val clothing = Clothing(
                name = name,
                type = type,
```

```
                color = color,
                season = season,
                imageUri = uriStr,
                usageCount = 0
            )

            lifecycleScope.launch {
                db.clothingDao().insert(clothing)
                finish()
            }
        }
    }

    override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
        super.onActivityResult(requestCode, resultCode, data)
        if (requestCode == PICK_IMAGE && resultCode == Activity.RESULT_OK) {
            selectedUri = data?.data
            imgPreview.setImageURI(selectedUri)
        }
    }
}
```

=== File: /src/main/java/com/example/digitalwardrobe/AnalyticsActivity.kt ===

```
package com.example.digitalwardrobe

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import com.example.digitalwardrobe.db.AppDatabase
import kotlinx.android.synthetic.main.activity_main.*
import kotlinx.android.synthetic.main.activity_main.view.*
import kotlinx.android.synthetic.main.activity_main.view.rvWardrobe
import kotlinx.android.synthetic.main.activity_main.view.tvTitle
import kotlinx.android.synthetic.main.activity_main.view.btnAdd
import kotlinx.android.synthetic.main.activity_main.view.btnView
import kotlinx.coroutines.launch
import android.widget.TextView
import android.view.ViewGroup
import android.view.LayoutInflater
import com.example.digitalwardrobe.model.Clothing

class AnalyticsActivity : AppCompatActivity() {

    private lateinit var db: AppDatabase
```

```
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        val root = LayoutInflater.from(this).inflate(R.layout.activity_main, null) as
ViewGroup
        val tv = TextView(this)
        tv.textSize = 16f
        root.removeAllViews()
        root.addView(tv)
        setContentView(root)

        db = AppDatabase.getDatabase(this)

        lifecycleScope.launch {
            val list = db.clothingDao().getAll()
            if (list.isEmpty()) {
                tv.text = "No clothing items yet. Add some items to see analytics."
            } else {
                val total = list.size
                val totalWorn = list.sumOf { it.usageCount }
                val avg = if (total > 0) totalWorn.toDouble() / total else 0.0

                val least = list.minByOrNull { it.usageCount }
                val most = list.maxByOrNull { it.usageCount }

                val sb = StringBuilder()
                sb.append("Total items: $total\n")
                sb.append("Total wears recorded: $totalWorn\n")
                sb.append("Average wears per item: %.2f\n".format(avg))
                sb.append("\nLeast worn: ${least?.name ?: "-"} (worn ${least?.usageCount
?: 0} times)\n")
                sb.append("Most worn: ${most?.name ?: "-"} (worn ${most?.usageCount ?:
0} times)\n")

                sb.append("\nDetailed list:\n")
                for (c in list) {
                    sb.append("- ${c.name}: ${c.usageCount}\n")
                }

                tv.text = sb.toString()
            }
        }
    }
}
```

=== File: /src/main/java/com/example/digitalwardrobe/SuggestionsActivity.kt ===

```
package com.example.digitalwardrobe

import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import androidx.lifecycle.lifecycleScope
import com.example.digitalwardrobe.db.AppDatabase
import com.example.digitalwardrobe.model.Clothing
import kotlinx.coroutines.launch
import android.widget.TextView
import android.view.ViewGroup
import android.view.LayoutInflater

class SuggestionsActivity : AppCompatActivity() {

    private lateinit var db: AppDatabase

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        val root = LayoutInflater.from(this).inflate(R.layout.activity_main, null) as
ViewGroup
        val tv = TextView(this)
        tv.textSize = 16f
        root.removeAllViews()
        root.addView(tv)
        setContentView(root)

        db = AppDatabase.getDatabase(this)

        lifecycleScope.launch {
            val list = db.clothingDao().getAll()
            val suggestion = suggestOutfit(list)
            if (suggestion == null) {
                tv.text = "No suitable outfit found. Add more tops/bottoms."
            } else {
                tv.text = "Try: ${suggestion.first.name} (Top) + ${suggestion.second.name}
(Bottom)\n\nDetails:\nTop - ${suggestion.first.type},
${suggestion.first.color}\nBottom - ${suggestion.second.type},
${suggestion.second.color}"
            }
        }
    }
```

```
    private fun suggestOutfit(clothes: List<Clothing>): Pair<Clothing, Clothing>? {
        val tops = clothes.filter { it.type.equals("Top", true) || it.type.equals("Shirt", true)
|| it.type.equals("Tops", true) }
        val bottoms = clothes.filter { it.type.equals("Bottom", true) ||
it.type.equals("Pants", true) || it.type.equals("Jeans", true) }

        if (tops.isEmpty() || bottoms.isEmpty()) return null

        // prefer less recently worn items to rotate wardrobe
        val topsSorted = tops.sortedBy { it.usageCount }
        val bottomsSorted = bottoms.sortedBy { it.usageCount }

        for (t in topsSorted) {
            for (b in bottomsSorted) {
                if (isColorMatch(t.color, b.color)) return Pair(t, b)
            }
        }

        // fallback pair
        return Pair(topsSorted.first(), bottomsSorted.first())
    }

    private fun isColorMatch(c1: String, c2: String): Boolean {
        val a = c1.trim().lowercase()
        val b = c2.trim().lowercase()
        if (a.isEmpty() || b.isEmpty()) return true
        if (a == "black" || b == "black") return true
        if (a == "white" || b == "white") return true
        if (a == b) return true
        // some basic matches
        val matches = mapOf(
            "blue" to listOf("white", "black", "gray"),
            "gray" to listOf("black", "white"),
            "brown" to listOf("white", "beige"),
            "red" to listOf("black", "white")
        )
        return matches[a]?.contains(b) ?: false
    }
}
```