

# GUÍA DE PRESENTACIÓN

## Aplicación Móvil Level-Up Gamer

Desarrollo de Soluciones Móviles

Evaluación Final Transversal

Fecha: 28 de noviembre de 2025

# ÍNDICE

1. Arquitectura del Proyecto
2. Tecnologías Implementadas
3. Estructura de Carpetas y Rutas
4. Pantallas Principales
5. Funcionalidades Clave
6. Base de Datos
7. Credenciales de Prueba
8. Preguntas Frecuentes del Profesor

# 1. ARQUITECTURA DEL PROYECTO

## ¿Cómo está diseñado este proyecto?

Esta aplicación móvil está desarrollada en **Android Studio** usando **Jetpack Compose** con **Kotlin** como lenguaje principal. No utiliza React, Bootstrap ni tecnologías web - es una aplicación nativa de Android 100%.

### Patrón de Arquitectura:

- **MVVM (Model-View-ViewModel)**
- **Repository Pattern** para acceso a datos
- **Room Database** para persistencia local
- **Jetpack Compose** para UI declarativa

# 2. TECNOLOGÍAS IMPLEMENTADAS

Tecnología	Versión/Descripción	Propósito
Android Studio	2024.x	IDE de desarrollo
Kotlin	1.9.x	Lenguaje de programación
Jetpack Compose	1.5.x	Framework UI moderno
Material Design 3	1.1.x	Sistema de diseño
Room Database	2.6.x	Base de datos local
Coroutines	1.7.x	Programación asíncrona
ViewModel	2.6.x	Gestión de estado
Navigation Compose	2.7.x	Navegación entre pantallas
Coil	2.4.x	Carga de imágenes
Gson	2.10.x	Serialización JSON

### 3. ESTRUCTURA DE CARPETAS Y RUTAS

#### Ruta Base del Proyecto:

C:\Users\Asgard\Aplicacion\_Mobile\_RodDev\

#### Estructura Principal:

```
app/src/main/java/com/levelup/gamer/
■
■ ■ ■ MainActivity.kt (Actividad principal - Orquestador de toda la app)
■ ■ ■ Ruta: app/src/main/java/com/levelup/gamer/MainActivity.kt
■
■ ■ ■ ui/screens/ (Todas las pantallas de la aplicación)
■ ■ ■ ■ HomeScreen.kt (Pantalla principal con productos)
■ ■ ■ ■ LoginScreen.kt (Inicio de sesión)
■ ■ ■ ■ RegisterScreen.kt (Registro de usuarios)
■ ■ ■ ■ ProfileScreen.kt (Perfil con estadísticas)
■ ■ ■ ■ CartScreen.kt (Carrito de compras)
■ ■ ■ ■ ProductDetailScreen.kt (Detalle de producto)
■ ■ ■ ■ PedidosScreen.kt (Historial de pedidos)
■ ■ ■ ■ AdminScreen.kt (Panel de administrador)
■ ■ ■ ■ CategoriesScreen.kt (Vista de categorías)
■ ■ ■ ■ NewsScreen.kt (Noticias)
■ ■ ■ ■ ContactScreen.kt (Contacto)
■ ■ ■ ■ SettingsScreen.kt (Configuración)
■
■ ■ ■ viewmodel/ (Lógica de negocio)
■ ■ ■ ■ AuthViewModel.kt (Autenticación)
■ ■ ■ ■ HomeViewModel.kt (Productos y búsqueda)
■ ■ ■ ■ CartViewModel.kt (Carrito)
■ ■ ■ ■ ProfileViewModel.kt (Perfil de usuario)
■ ■ ■ ■ AdminViewModel.kt (Gestión administrativa)
■ ■ ■ ■ PedidosViewModel.kt (Pedidos)
■
■ ■ ■ repository/ (Acceso a datos)
■ ■ ■ ■ ProductoRepository.kt (Productos)
■ ■ ■ ■ auth/AuthRepository.kt (Autenticación)
■ ■ ■ ■ carrito/CarritoRepository.kt (Carrito)
■ ■ ■ ■ favoritos/FavoritosRepository.kt (Favoritos)
■ ■ ■ ■ pedido/PedidoRepository.kt (Pedidos)
■
■ ■ ■ data/ (Base de datos)
■ ■ ■ ■ database/AppDatabase.kt (Configuración Room)
■ ■ ■ ■ dao/
■ ■ ■ ■ ■ UserDao.kt
■ ■ ■ ■ ■ CarritoDao.kt
■ ■ ■ ■ ■ FavoritoDao.kt
■ ■ ■ ■ ■ PedidoDao.kt
■ ■ ■ ■ UserPreferences.kt (Preferencias)
■
■ ■ ■ model/ (Modelos de datos)
■ ■ ■ ■ UserEntity.kt (Usuario)
■ ■ ■ ■ Producto.kt (Producto)
■ ■ ■ ■ CarritoItem.kt (Item del carrito)
■ ■ ■ ■ Pedido.kt (Pedido)
■ ■ ■ ■ FavoritoEntity.kt (Favorito)
```

## 4. PANTALLAS PRINCIPALES - RUTAS EXACTAS

Pantalla	Archivo	Ruta Completa
Inicio/Home	HomeScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/HomeScreen.kt
Login	LoginScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/LoginScreen.kt
Registro	RegisterScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/RegisterScreen.kt
Perfil	ProfileScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/ProfileScreen.kt
Carrito	CartScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/CartScreen.kt
Detalle Producto	ProductDetailScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/ProductDetailScreen.kt
Pedidos	PedidosScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/PedidosScreen.kt
Administrador	AdminScreen.kt	app/src/main/java/com/levelup/gamer/ui/screens/AdminScreen.kt

## 5. FUNCIONALIDADES CLAVE Y SUS UBICACIONES

### A. BÚSQUEDA DE PRODUCTOS

¿Dónde está el buscador?

- **Archivo:** HomeScreen.kt (líneas 72-93)
- **Ruta:** app/src/main/java/com/levelup/gamer/ui/screens/HomeScreen.kt
- **Lógica:** HomeViewModel.kt (función `updateSearchQuery` y `getFilteredProducts`)
- **Componente:** OutlinedTextField con placeholder 'Buscar productos...'

### B. SISTEMA DE FAVORITOS

¿Dónde está el botón de favoritos?

- **Archivo:** ProductDetailScreen.kt (líneas 90-110)
- **Ruta:** app/src/main/java/com/levelup/gamer/ui/screens/ProductDetailScreen.kt
- **Repository:** FavoritosRepository.kt
- **Base de Datos:** FavoritoEntity.kt + FavoritoDao.kt
- **Ícono:** Corazón rojo (favorito) / Corazón blanco (no favorito)

### C. CARRITO DE COMPRAS

¿Dónde se maneja el carrito?

- **Pantalla:** CartScreen.kt
- **ViewModel:** CartViewModel.kt
- **Repository:**  
app/src/main/java/com/levelup/gamer/repository/carrito/CarritoRepository.kt
- **Detalle de boleta:** CartScreen.kt (líneas 233-370) - CartSummary composable
- **Descuento DUOC:** Se calcula automáticamente si el email termina en @duocuc.cl

### D. SISTEMA DE PUNTOS

¿Dónde se calculan los puntos?

- **Cálculo:** PedidoRepository.kt (línea 31) - 5% del total
- **Ruta:** app/src/main/java/com/levelup/gamer/repository/pedido/PedidoRepository.kt
- **Visualización:** ProfileScreen.kt (UserInfoCard) + CartScreen.kt (CartSummary)
- **Actualización:** Se guardan en cada pedido y se suman en el perfil

### E. PANEL DE ADMINISTRADOR

¿Dónde está el panel de admin?

- **Archivo:** AdminScreen.kt (596 líneas)
- **Ruta:** app/src/main/java/com/levelup/gamer/ui/screens/AdminScreen.kt
- **ViewModel:** AdminViewModel.kt
- **Acceso:** Solo usuarios con `isAdmin = true`
- **Funciones:** Agregar/quitar stock, crear productos, eliminar productos

### F. BOTÓN CERRAR SESIÓN

¿Dónde está el botón de cerrar sesión?

- **Ubicación:** ProfileScreen.kt (componente LogoutButton, líneas 466-520)
- **Ruta:** app/src/main/java/com/levelup/gamer/ui/screens/ProfileScreen.kt
- **Acción:** Muestra diálogo de confirmación antes de cerrar sesión
- **Nota:** Ya NO está en SettingsScreen, se movió al perfil

## 6. BASE DE DATOS - ROOM DATABASE

### Configuración Principal:

- **Archivo:** AppDatabase.kt
- **Ruta:** app/src/main/java/com/levelup/gamer/data/database/AppDatabase.kt
- **Versión:** 6

### Tablas de la Base de Datos:

Tabla	Entity	DAO	Propósito
users	UserEntity.kt	UserDao.kt	Usuarios registrados
carritos	CarritoItem.kt	CarritoDao.kt	Items del carrito
pedidos	Pedido.kt	PedidoDao.kt	Historial de pedidos
favoritos	FavoritoEntity.kt	FavoritoDao.kt	Productos favoritos

### Campos Importantes de UserEntity:

- **id:** Identificador único
- **email:** Correo electrónico (único)
- **password:** Contraseña
- **nombre:** Nombre del usuario
- **isAdmin:** Boolean - true para administradores
- **descuentoPorcentaje:** Descuento automático (20% para @duocuc.cl)
- **puntos:** Puntos acumulados
- **totalCompras:** Cantidad de compras realizadas

## 7. CREDENCIALES DE PRUEBA

### Usuario Administrador:

- **Email:** `admlvlup@lvlup.cl`
- **Password:** `admin123`
- **Permisos:** Acceso completo al panel de administrador
  - **Ubicación en código:** `AuthRepository.kt` (línea 76-84)

### Usuario Normal:

- **Email:** `usuariolvlup@lvlup.cl`
- **Password:** `user123`
- **Permisos:** Usuario estándar sin acceso admin
  - **Ubicación en código:** `AuthRepository.kt` (línea 86-93)

### Nota Importante:

Estos usuarios se crean automáticamente al iniciar la app por primera vez si no existen. El código está en `AuthRepository.kt` en la función `initializeDefaultUsers()`.



## 8. PREGUNTAS FRECUENTES DEL PROFESOR

### P1: ¿Este botón dónde está?

R: Buscar el componente en el archivo correspondiente. Por ejemplo:

- Botón 'Agregar al Carrito': ProductDetailScreen.kt (línea ~180)
- Botón 'Finalizar Compra': CartScreen.kt (línea ~140)
- Botón 'Favoritos': ProductDetailScreen.kt (línea 90-110)
- Botón 'Cerrar Sesión': ProfileScreen.kt (línea 466)

### P2: Si cambio algo aquí, ¿dónde más debo modificar?

R: Depende del tipo de cambio:

- **Cambio en UI:** Solo el archivo .kt de la pantalla (por ejemplo, HomeScreen.kt)
- **Cambio en datos:** Model (Entity) → DAO → Repository → ViewModel → Screen
- **Nueva funcionalidad:** Crear en todos los niveles (Model, DAO, Repository, ViewModel, UI)

### P3: ¿Dónde se controla este error?

R: Los errores se manejan en:

- **Validaciones de formulario:** En el Screen correspondiente (LoginScreen, RegisterScreen)
- **Errores de base de datos:** En los Repository con try-catch
- **Errores de autenticación:** AuthViewModel.kt (líneas 93-140)
- **Mensajes al usuario:** SnackbarHostState en MainActivity o cada Screen

### P4: ¿Cómo se comunican las pantallas?

R: A través de:

- **ViewModel compartido:** Para datos entre pantallas
- **Callbacks (onBack, onClick):** Para navegación
- **currentScreen variable:** En MainActivity.kt controla qué pantalla mostrar
- **StateFlow:** Para observar cambios en tiempo real

### P5: ¿Dónde están los productos?

R: Los productos están en:

- **Archivo:** ProductoRepository.kt (línea 99-360)
- **Ruta:** app/src/main/java/com/levelup/gamer/repository/ProductoRepository.kt
- **Tipo:** Lista mutable de objetos Producto
- **Cantidad:** 11 productos predefinidos

### P6: ¿Cómo funciona el descuento DUOC?

R:

- **Detección:** UserEntity.kt (línea 36-43) - función calcularDescuentoPorEmail
- **Condición:** Si el email termina en '@duocuc.cl' → 20% descuento
- **Aplicación:** CartScreen.kt al calcular el total
- **Visualización:** CartSummary muestra 'Descuento DuocUC: -\$XXX'

## MAPA RÁPIDO DE UBICACIONES

Necesito encontrar...	Archivo	Línea Aprox.
Buscador de productos	HomeScreen.kt	72-93
Botón favoritos	ProductDetailScreen.kt	90-110
Carrito de compras	CartScreen.kt	Todo el archivo
Detalle de boleta	CartScreen.kt	233-370
Panel admin	AdminScreen.kt	Todo el archivo
Sistema de puntos	PedidoRepository.kt	31-52
Cerrar sesión	ProfileScreen.kt	466-520
Login	LoginScreen.kt	Todo el archivo
Registro	RegisterScreen.kt	Todo el archivo
Productos	ProductoRepository.kt	99-360
Base de datos	AppDatabase.kt	Todo el archivo
Usuarios default	AuthRepository.kt	70-95

## CONSEJOS PARA LA PRESENTACIÓN

1. Tener Android Studio abierto con el proyecto cargado
2. Conocer la ruta completa: C:\Users\Asgard\Aplicacion\_Mobile\_RodDev
3. Mostrar la app corriendo en el emulador o dispositivo
4. Demostrar login con ambas cuentas (admin y usuario)
5. Navegar por todas las pantallas principales
6. Mostrar el código de una funcionalidad clave (favoritos o búsqueda)
7. Explicar el patrón MVVM con un ejemplo visual
8. Tener preparado abrir cualquier archivo rápidamente (Ctrl+Shift+N)

# ¡ÉXITO EN TU PRESENTACIÓN!