

**Nama : Muhammad Adh-Dhiya'Us Salim**

**NIM : 2310817210022**

**KuisMobile-1 / Pemrograman Mobile**

### **Soal**

1. Jelaskan dengan pemahaman kalian mengenai clean architecture di Android!
2. Buatlah struktur file menggunakan data dummy di Android Studio pada Clean Architecture
3. Menurut kalian, apakah wajib menerapkan clean architecture pada penetapan Android dan case apa kita harus menerapkan clean architecture

### **Jawaban**

1. Clean Architecture adalah desain arsitektur perangkat lunak yang menekankan pemilahan kode ke dalam lapisan-lapisan terstruktur agar arah ketergantungan kode hanya mengarah ke dalam (inward). Ini mengikuti prinsip Dependency Rule, di mana kode di lapisan luar bergantung pada lapisan yang lebih dalam, bukan sebaliknya.
2. Menggunakan cmd cd D:\location file lalu tree /F untuk menampilkan struktur file Clean Architecture :

D:.

```
├──data
│   ├──model
│   │   MovieEntity.kt
│   │
│   ├──repository
│   │   MovieRepository.kt
│   │
│   └──source
│       ApiService.kt
│       AppDatabase.kt
│       MovieDao.kt
│       MovieDto.kt
│
```

```
├── di
│   ├── NetworkModule.kt
│   └── RepositroyModule.kt
│
├── domain
│   ├── model
│   │   └── Movie.kt
│   │
│   ├── repository
│   │   └── MovieRepository.kt
│   │
│   └── usecase
│       └── GetPopularMovieUseCase.kt
│
└── presentation
    ├── MainActivity.kt
    │
    ├── navigation
    │   └── AppNavigation.kt
    │
    ├── screen
    │   ├── MovieDetailScreen.kt
    │   └── MovieListScreen.kt
    │
    ├── theme
    │   ├── Color.kt
    │   ├── Theme.kt
    │   └── Type.kt
    │
    ├── ui
    └── viewmodel
```

Link Github : <https://github.com/Adiiaus/KuisMobile-1.git>

3. Clean Architecture tidak wajib diterapkan dalam pengembangan aplikasi Android, namun sangat disarankan pada proyek yang berskala menengah hingga besar, melibatkan banyak developer, memiliki logika bisnis yang kompleks, atau direncanakan untuk jangka panjang karena arsitektur ini memisahkan tanggung jawab antar lapisan (UI, domain, data) sehingga memudahkan pengujian, pemeliharaan, dan pengembangan fitur baru. Sebaliknya, untuk proyek kecil, aplikasi demo, tugas kuliah, atau pengembangan cepat oleh satu orang, penerapan Clean Architecture bisa menjadi berlebihan karena menambah kompleksitas dan waktu pengembangan yang tidak proporsional dengan kebutuhan proyek.