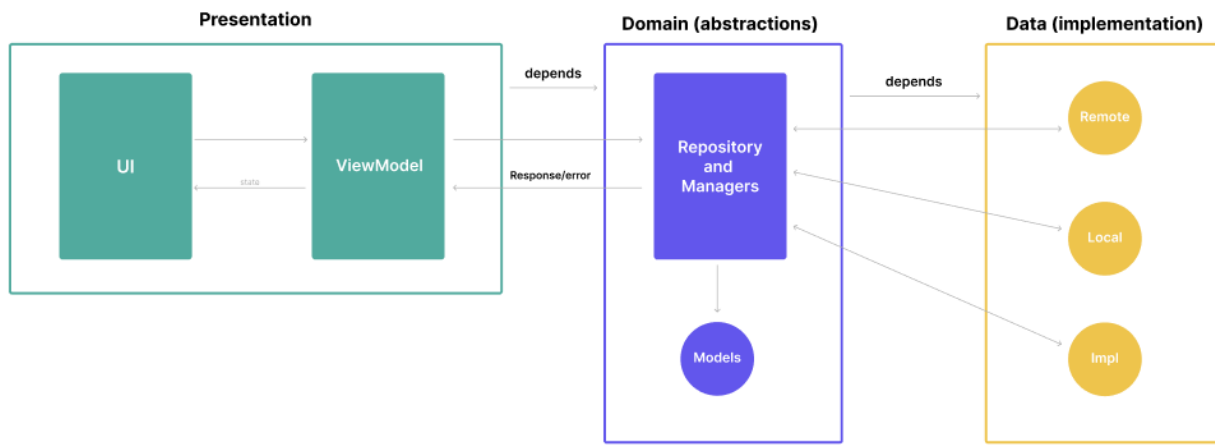# Clean Architecture Guide

1. **Presentation Layer**:
   - **UI**: Interacts with `ViewModel` to display data.
   - **ViewModel**: Depends on abstractions defined in the `Domain` layer (repositories).
2. **Domain Layer**:
   - **Repository and Managers**: Defines interfaces (protocols) and business logic.
   - **Models**: Data structures used within the domain layer.
3. **Data Layer**:
   - **Remote, Local, Implementation**: Concrete implementations of the interfaces defined in the `Domain` layer (e.g., network requests, local database operations).
4. **DIContainer**:
   - Manages the creation and provision of dependencies.
   - Ensures that the `ViewModel` gets the appropriate services from the `Domain` layer.
   - Provides the `Domain` layer with implementations from the `Data` layer