| **MVC(Model View Controller)** | **MVVM(Model View ViewModel)** |
| --- | --- |
| Oldest android app architecture. | Industry-recognized architecture pattern for applications. |
| User Inputs are handled by the **Controller**. | The **View**takes the input from the user and acts as the entry point of the application. |
| **Controller**and **View**exist with the one-to-many relationship. One **Controller** can select different **View** based upon required operation. | Multiple **View**can be mapped with single **ViewModel** and thus, the one-to-many relationship exists between **View** and **ViewModel.** |
| The **View**has no knowledge about the **Controller**. | The **View**has reference to the **ViewModel**. |
| This architecture has high dependency on the Android APIs. | Has low or no dependency on the Android APIs. |
| Difficult to make changes and modify the app features as the code layers are tightly coupoled. | Easy to make changes in application. Howerver, if data binding logic is too complex, it will be a little harder to debug the application. |
| Limited support to Unit testing. | Unit testability is highest in this architecture. |
| It does not follow modular and single responsibility principle. | Follows modular and single responsibility principle. |