**Architectural Pattern: MVVM Model**

Defining an architectural pattern for the application is important to maintain scalability and maintainability. The architecture acts as the backbone for developing the project as it defines a set of guidelines that need to be followed. Defining an architecture makes it easy to incorporate changes in the project as it reduces tight coupling.

There are three main architectural patters available for developing android applications. The Model View Controller (MVC), Model View Presenter (MVP) and the most recent Model View ViewModel (MVVM). On further studying these models we found MVVM Model to be the one that would be best suit our requirements.

The MVVM architectural pattern is used to simplify Event- Based implementation of the user interface. The three components of the MVVM Model is the Model, View and ViewModel. The Model is responsible for providing the data to whoever needs it. The View is responsible for displaying the data. ViewModel gets the data from the model, applies the UI related logic and provides it to either a View or any Class that needs it. It provides a clear separation between the domain logic and the business layer

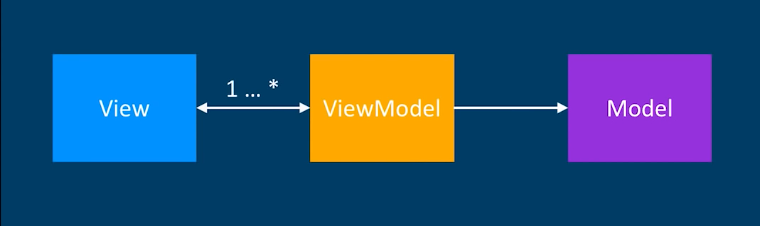


Figure : MVVM (Source: https://cdn-images-1.medium.com/max/1000/1\*txL-xUrnT8ga3zx9K4Xl2g.png)

As the To Do application would require the task list to be updated every time there is a new task added or edited. The event driven MVVM Model would work best.

**References**

[1] Kumar, V. (2018). *Android Architecture Patterns : MV( C | P | VM ) – MindOrks – Medium*. [online] Medium. Available at: https://medium.com/mindorks/android-architecture-patterns-mv-c-p-vm-4594574eeaa1 [Accessed 31 Oct. 2018].

[2] Glebocki, P. (2018). *Considering architecture for Android app – stepstone-tech – Medium*. [online] Medium. Available at: https://medium.com/stepstone-tech/considering-architecture-for-android-app-f7f0fabf680a [Accessed 31 Oct. 2018].