Jonathan Disla

Discussion 9.1

Web 330

Prof. Krasso

MVC vs MVVM

Modern web applications have become very large throughout time. Not only are the applications large but so are the development teams managing them. The model view controller pattern is aimed at separating the application into sections that have unique purposes. Under MVC, when a user makes a request, it is handled by the controller. Once the data has been required, the model handles the data logic as well as the interaction with a database. The controller never handles the data and the model never handles user requests or validation of the request. After the model sends the data to the controller, the controller then interacts with the view to render the data which is returned to the controller and then to the user. With MVVM, the view model acts similar to a controller as a middle man. The model handles business logic such as classes and structure that represents collections. The view in MVVC is what is seen on the browser, it is the visual representation. According to Eric Maxwell, product engineer at Realm,“Both MVP and MVVM do a better job than MVC in breaking down your app into modular, single purpose components, but they also add more complexity to your app. For a very simple application with only one or two screens, MVC may work just fine. MVVM with data binding is attractive as it follows a more reactive programming model and produces less code.”

Reference List

Maxwell, Eric. “MVC vs MVC vs MVVM on Android.” Realm. 26 January 2017.

<https://academy.realm.io/posts/eric-maxwell-mvc-mvp-and-mvvm-on-android/>