

## WK - 03 : MVC Architecture

- **What is MVC architecture?**

Model View Controller is a software design pattern. The main idea is to separate business logic from the applications view or UI.

Application logic is located in controllers. Views are responsible for the rendering of the appropriate UI. Models are usually used to transfer data back and forth between controllers and views.

- **If you fill in a form and then click on submit, how does the flow work according to MVC?**

**Model** - The class or object which defines the structure of form data. This part used to store and retrieve the data from the database.

**View** - The part in the javascript responsible for rendering the UI / after getting the data from the controllers.

**Controllers** - The functions in which controls the routes in which data is stored and retrieved from.

- **Name the three layers of MVC architecture.**

Model View Controller

- **What is the role of model, view and controller in MVC architecture?**

**Model** - Model represents the shape of the data and business logic. It maintains the data of the application. Model objects retrieve and store model state in a database.

**View** - View is a user interface. View displays data using models to the user and also enables them to modify the data.

**Controllers** - Controller handles the user request. Users interact with View, which in-turn raises appropriate URL requests, this request will be handled by a controller. The controller renders the appropriate view with the model data as a response.

- **Explain the mutual dependence between the model, view and controller.**

Model is the business layer of an application. It contains classes and application logic. View is the front-end or interface through which a user interacts with our application. Controller is the bridge between Model and View. It is used to handle requests.

- **What is the fundamental difference between MVC and MVVM architecture?**

The difference between the MVC and MVVM is View and Controller are responsible for calculating the value and assigning the value so the load is more on View and Controller where in MVVM View and Controller are only responsible for assigning the value not calculating the value.

- **What are the advantages of MVC architecture?**

Separation of responsibilities help in building scalable and manageable large applications.