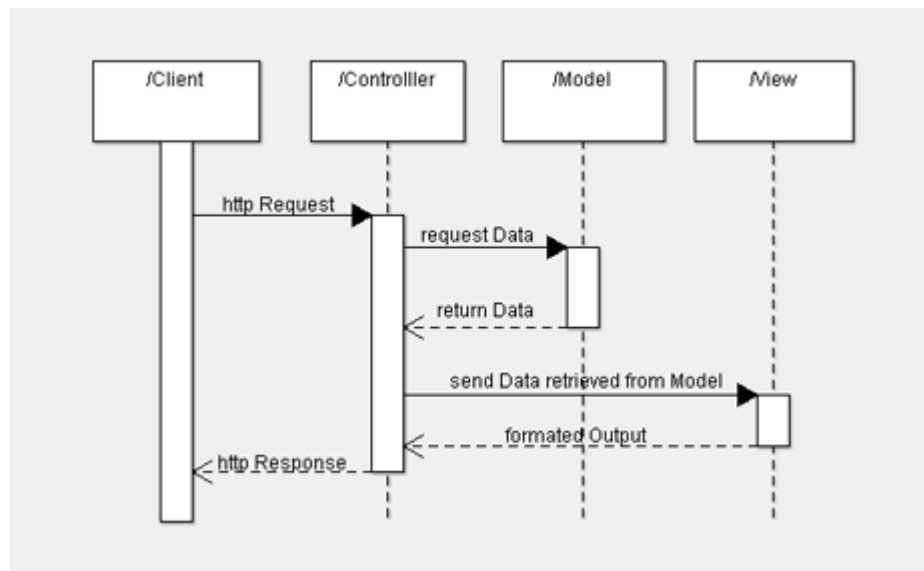
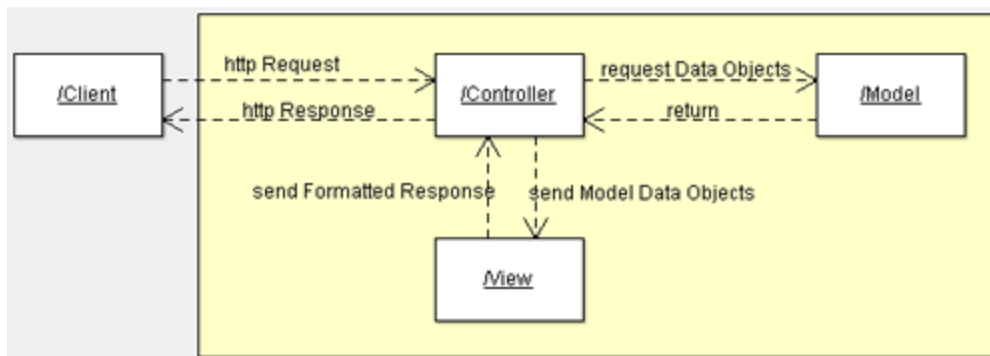


Model-View-Controller Pattern

Model-view-controller (MVC) is a frequently-used design pattern for implementing user interfaces. It divides an application into three interconnected parts, exposing the user only to the ways information is presented (View) and ways to interact with data (Controller), while hiding the internal workings of the data (Model) and their interactions. The pattern separates the modeling of the domain, the presentation, and the actions based on user input into three separate classes.



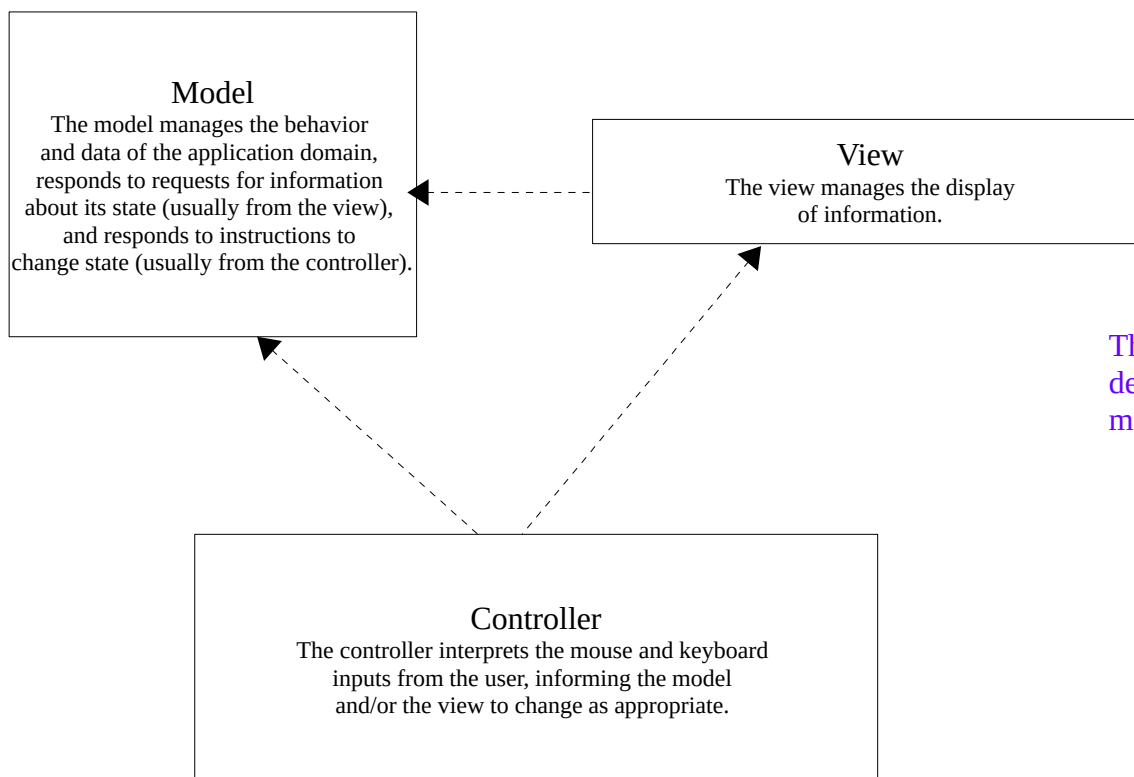
From Wikipedia:

History [\[edit\]](#)

MVC was one of the seminal insights in the early development of [graphical user interfaces](#), and one of the first approaches to describe and implement software constructs in terms of their [responsibilities](#).^[9]

[Trygve Reenskaug](#) introduced MVC into Smalltalk-76 while visiting [Xerox Parc](#)^{[10][11]} in the 1970s. In the 1980s, Jim Althoff and others implemented a version of MVC for the [Smalltalk-80](#) class library. It was only later, in a 1988 article in [The Journal of Object Technology](#), that MVC was expressed as a general concept.^[12]

The MVC pattern has subsequently evolved,^[13] giving rise to variants such as [HMVC](#), [MVA](#), [MVP](#), [MVVM](#), and others that adapted Model View Controller to different contexts.



The view and the controller depend on the model, but the model doesn't depend on either