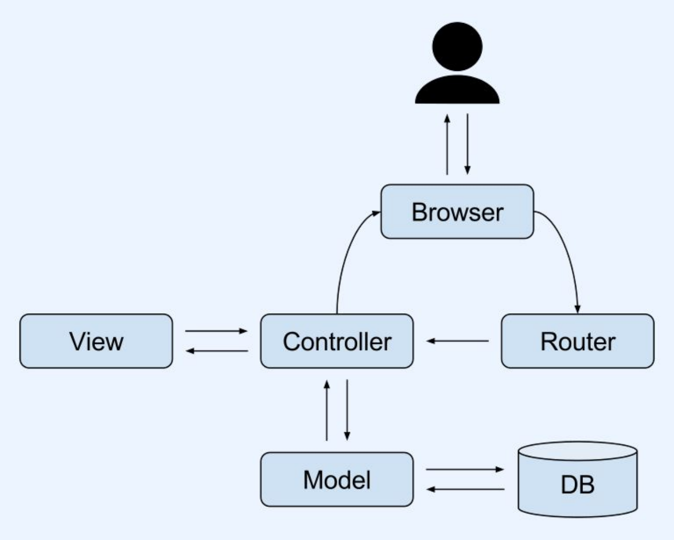
MVC

Bringing order to the coding chaos

Without good planning and organization, our work as software engineers can very quickly devolve into a chaotic mess that nobody else is going to want to have to deal with. A single file doing all of the heavy lifting may sound efficient, but this makes it very difficult to truly make sense of what’s going on or to even troubleshoot basic issues.

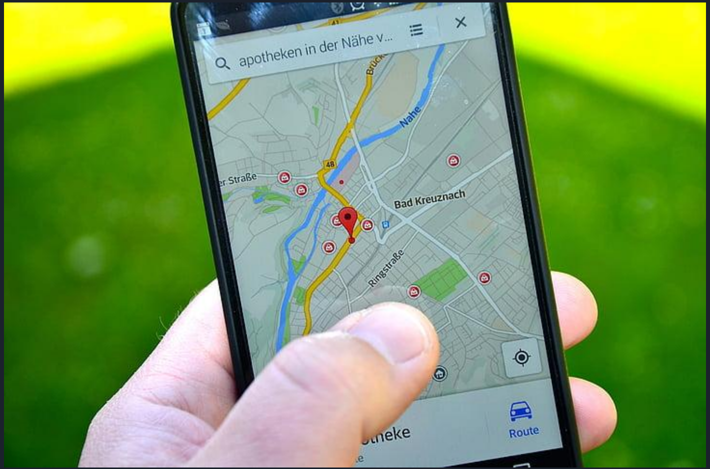
Enter MVC

MVC(Model-View-Controller) is a system that software engineers use to keep production smooth and easy to work with. Much in the same way an engineer would employ separation of concerns with their HTML, CSS and Javascript, MVC goes a step further and separates our javascript into chunks that specialize in their own specific workload. These are the Model, View and Controller.



MVC has many variations but learning the fundamentals will allow you to adapt to any of these with ease. The structure we will be talking about contains a Router, Controller, Model, and View. These are all crucial pieces of the puzzle and our apps would not function if you even remove just one of these.

Router



Once the client initiates a reaction from our javascript it falls upon our Router to make sense of the request and make sure it gets sent to the correct location so that we can properly respond. If our Router is configured incorrectly then we will end up executing the wrong code for what was requested. The Router should not be doing anything besides directing requests to their proper Controllers.

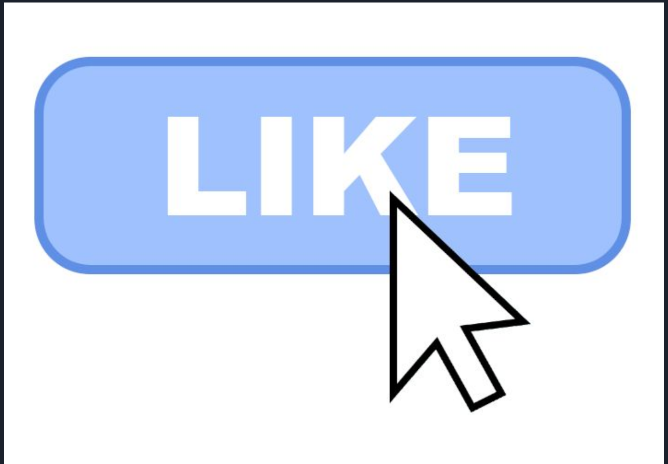
Controller

Controllers are the main muscle of our Javascript. The Router initiates the code in a specific controller to perform an action that will impact the app in some way. This can be visible to the user or not at all. If these actions require communication with the database or alteration of data, then the Controller will interact with the Model, otherwise it will go straight to the View to make these changes.

Model

The Model deals with changes in data or interactions with our database. If our user likes a post or creates something on their account, the Controller will pass these requests to the Model which will CREATE, READ, UPDATE or DELETE data on the database and then reflect these changes onto the View.

View



The View dictates what is displayed on the app. It takes the information from either the Controller or the Model and plugs that data in for the user to either see or interact with. When you like a post on social media, the View is what shows the update on the page for the user to see. This is purely for the user experience.

Order restored to codebases

As previously stated, there are numerous iterations of MVC used by software engineers, but knowing the fundamentals will prepare you for any of them. MVC makes working with a team that much easier when you can stay out of each other’s way and easily pinpoint what piece of code that the team needs to troubleshoot. It is also much easier to look at the code and see at a glance what its purpose is.