

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green color. They are positioned diagonally, with the blue one in front of the green one.

The MVC (Model View Controller)

A brief summarization of Leon's big brain
knowledge of MVC

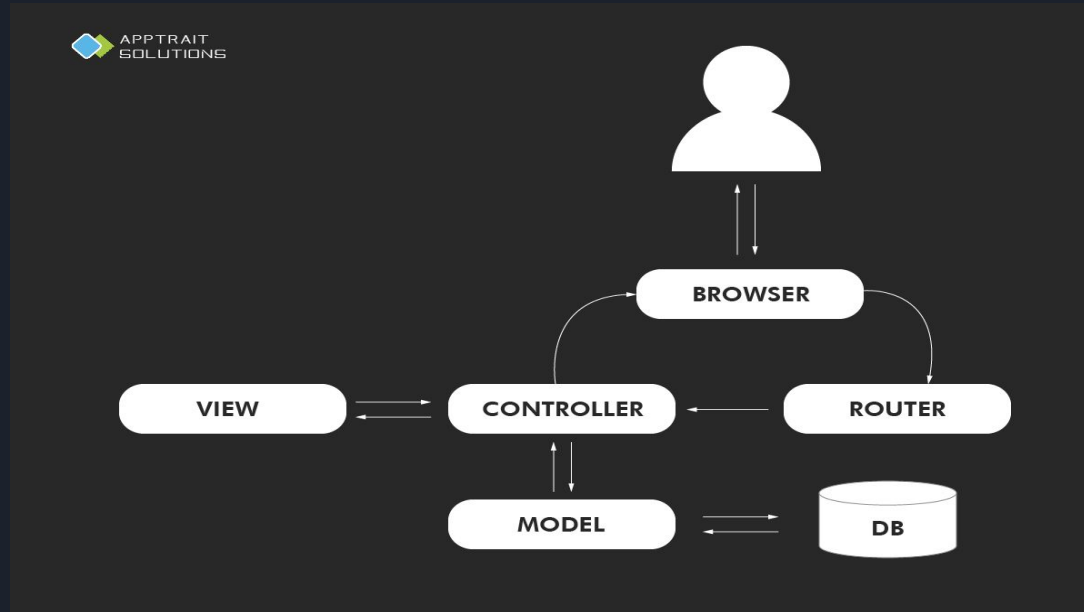


What is the MVC?

The MVC is a way that software engineers can organize their code to make it easier to read, easier to edit, and to follow the various design principles like the separation of concerns.

There are three main folders, Model, View, and Controller, that are used to make this happen in addition to a router folder that helps navigate between them.

Diagram of the flow between all folders





The Router

The router is in charge of telling the controller where to go next after getting the request from the browser through a require statement.

There can be several controllers and there is usually at least one for each page of your website.

The info it sends to the controller could be a require statement or a CRUD request.



The Controller

Once it reaches the Controller, it has three ways it could go depending on the request. It could go back to the browser, it could go to the model folder to interact with the database, or it could contact the views folder.

For example, adding an entry to the database is something the Controller can do. The model folder helps the controller interact with the database, but the Controller is the one who does the CRUD for the database. Then it may make a request to the views folder to interact with an EJS file, otherwise it would go back to the browser afterwards.



Models

The model folder is used to interact with the database so you can keep track of the specific information going in and out of it

Mongoose is an example of something that would be in the model folder since it moderates the information go to and from the database by using a schema.

A schema is like a template and you can apply rules to it. You can use it to make sure that extra info isn't added to the object being put into the database, to make sure there isn't any problematic data going to the database (a number in a name or a curse word), filtering out duplicates, etc



View

The views folder will contain your HTML, PHP, EJS, and other similar pages for your website. Each page of your website would be in here and the controller will grab the one it wants because the routes file told it what to do.



Additional Info

`app.set('view engine', 'ejs')` tells express that you will be using EJS, but also that there is a views folder and where it should be in relation to the server.js file

When we are in different paths like in the controller folder, we know where ejs is still because we use require statements starting from the server.js file.

The `module.exports` at the end of certain pages is how it returns the info to whatever contacted it through a require statement.