DESIGN PATTERNS IN PHP

Design patterns generally makes our work easier and allows others to easily understand our code.

We have different design patterns:

1. Factor pattern
2. Singleton pattern
3. Strategy pattern
4. Front controller pattern
5. MVC pattern

I will be talking about the MVC pattern.

MVC PATTERN

The MVC stands for model, view and controller. The MVC allows us to break our codes into different logical objects that serve very specific purpose. In other words, each of our files will perform specific tasks.

HOW THE MVC PATTERN INTERACT WITH EACH OTHER:

Model: It serve as a data access layer, where data is fetched and returned as a format that is usable in an application. It is the central component of the pattern.

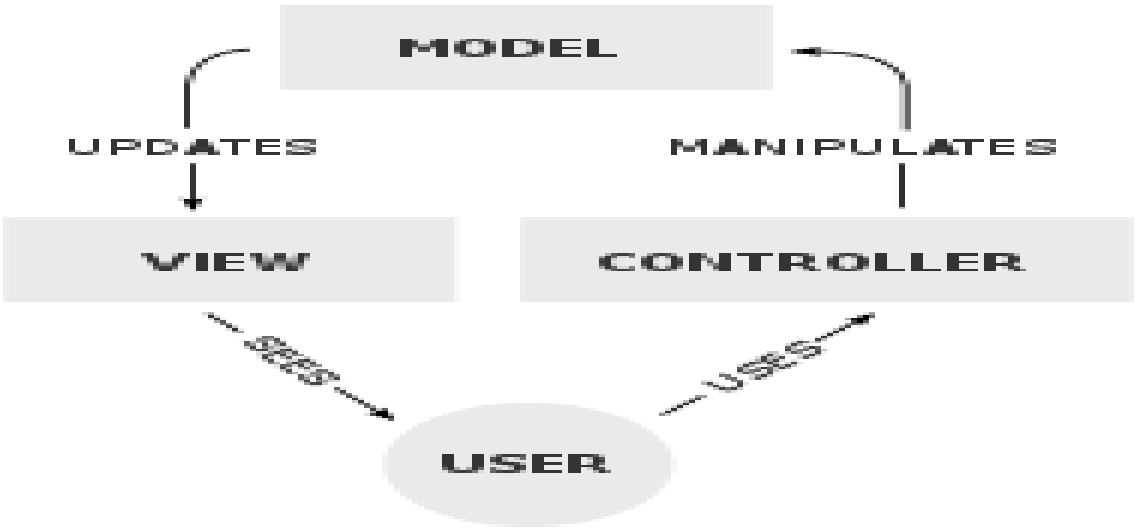
While;

The controller: It serve as a request handler that handles requests, processes data returned from the **Model** and loads the view sent to the browser.

And;

The View: is basically our display template, for example HTML, that is sent to the browsers for the users to see.

DIAGRAM OF THE INTERACTION WITHIN THE MVC PATTERN



BENEFITS OF USING THE MVC PATTERN

1. Simultaneous development: what I mean here is that it allows multiple developers to work on different parts of a project at the same time. For example, some developers may be working on model, another on controller, and another on views all at the same time.
2. Efficient testing: for example, you can test the controller without touching the model.
3. Better code maintenance.
4. You can reuse code.

POPULAR PHP FRAMEWORKS THAT USE THE MVC PATTERN

1. Codeigniter
2. Laravel
3. Cakephp
4. Symphony and many more