Week 6 Commentary

What is the aim of an MVC Framework?

The aim and purpose of an MVC Framework is to separate your code by its function i.e. by what it does. It is separated into 3 main categories; Model, View, and Controller, whereby Model is for the data for the program, such as a database or a file. View is the means of displaying the program/web, and Controller is the code that contains the actioning and using of this data. In short, MVC encourages decoupling, meaning that if you need to change the code in one area, this can be done without affecting another area of the website.

What are the advantages of an MVC Framework?

An advantage of using an MVC Framework is as mentioned before, it's decoupling, in that one can modify one a function of a website without affecting the functionality of another unrelated section. In terms of web development, MVC constitutes the Microservices architecture, as opposed to the Monolithic architecture, wherein all components of a web page are all in the one .html or .php file.

Another advantage is that because of its duplicity by design, there can be multiple ways in which one can view the model for the data, whilst using the same code from the controller if applicable. This calling of code methods means that the developer doesn't have to write out the same function again; they can just call the function and use it for another purpose.

Another advantage of MVC is that a web developer could for example, change the entire layout of a website, with little to none edits on the pages themselves, regarding web content on their site. In short, MVC in web development allows for greater flexibility regarding the creation and altering of the web pages themselves, compared to having all the code for one web page all on one file.

Using a diagram, show how your code from the RPC works, using the example supplied.

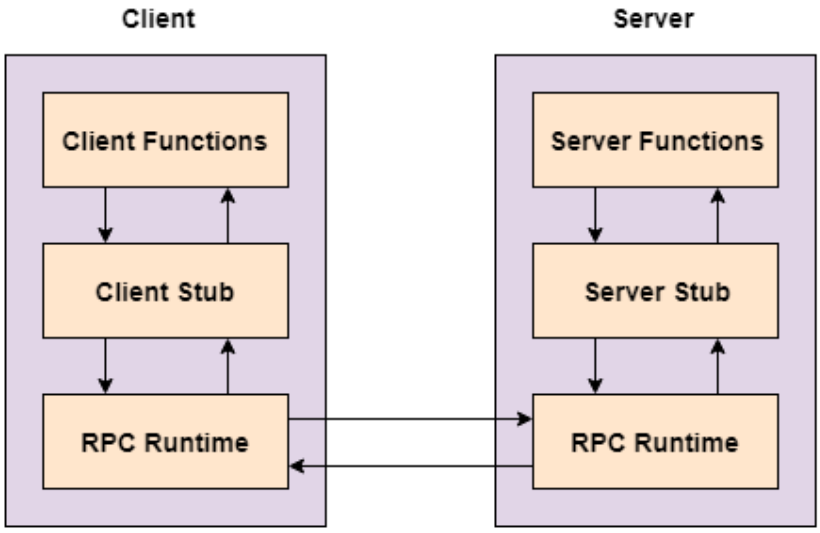

Image from https://www.tutorialspoint.com/assets/questions/media/12686/RPC.PNG

Remote Procedure Call (RPC) is a technique of distributed computing that allows web developers to construct client-server-based web applications. It is based on extending the conventional local procedure calling so that the called procedure doesn't have to exist in the same address space as the calling procedure and can be called remotely. The two processes may be on the same system, or they may be on different systems with a network connecting them. RPC is generally more about the latter, as it is primarily about remote data, hence its name.
The process for the remote procedure call is as follows in the sequence below:

1. The client stub is called by the client.
2. The client stub then makes a system call to send the message to the server and puts the parameters in the message
3. The message is sent from the client to the server by the client's operating system.
4. The parameters are deleted from the respective message by the server stub.
5. Finally, the server procedure is called by the server stub.

In short, RPC allows for one program to get and call services and/or data from another computer on a network on the same, or a different network, than which the computer calling the data is on.

As to what knowledge was required to work on the code as provided in the 'week06.zip' package of files from MyLearningSpace, I needed to know that the data was being sent as JSON array, and what each attribute was named, as this is essential for the editing and expansion of the pages.