

## **1- El patró de desenvolupament Model Vista Controlador (MVC)**

a) Què és i quines són les seves avantatges?:

- Patró de disseny per desenvolupar aplicacions web.
- El propòsit principal és separar la lògica del negoci de la interfície d'usuari.
- Això permet una millor estructuració i organització del programari de l'aplicació i facilita que múltiples usuaris treballin en un mateix projecte.
- Amb el patró MVC es poden crear projectes modulars i escalables.
- Permet facilitar el manteniment i actualització de les aplicacions.
- Fàcilment implementable utilitzant frameworks desenvolupats per treballar amb aquest patró i amb diferents llenguatges de programació com per exemple Laravel per PHP.
- Permet reutilitzar el codi.
- Facilita l'encapsulament del codi. Cada part o component del codi tindrà unes funcions ben definides que no dupliquen i es barregen amb les realitzades per altres components del codi. Això implica que també les interrelacions entre les diverses parts del codi queden també ben definides.
- Facilita evitar el codi tipus "spaguetti".

b) Separació de l'aplicació en 3 parts o capes interconnectades:

- Vista
- Model
- Controlador

c) Vista(View):

- És la part de l'aplicació que interactua amb l'usuari normalment per mitjà d'una interfície gràfica.
- Mostra a l'usuari dades rebudes des del Model en forma de taules, gràfiques, diagrames, textos, etc...
- Permet a l'usuari introduir dades o realitzar accions (esdeveniments) com fer clics sobre botons.
- La Vista es crearà en el servidor i és visualitzarà en el programa client que habitualment serà un navegador i farà servir bàsicament HTML/CSS i també pot utilitzar javascript (o altres llenguatges) si és necessari.
- L'usuari pot accedir a la Vista normalment des del Navegador.
- En general, la Vista és enviada al Navegador pel Controlador.
- En general, la Vista es comunica només amb el Controlador.

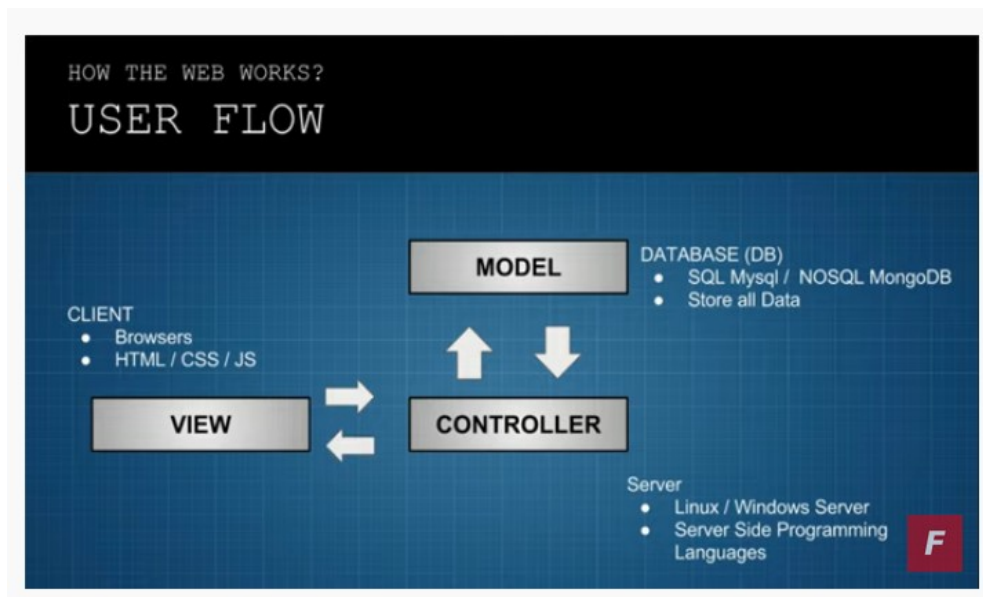
d) El Model (Model):

- La seva funció principal és obtenir i manipular les dades.
- És responsable de les interaccions amb les bases de dades fent queries com INSERT, SELECT, UPDATE, DELETE. També podria interactuar amb un fitxer XML o JSON.
- També és responsable de la gestió dels privilegis d'accés.
- En general es comunica només amb el Controlador però alguns frameworks permeten que algunes vegades actualitzi directament la Vista.
- El Model es trobarà en el costat servidor i utilitza per exemple PHP.

e) El Controlador (Controller):

- És la part de l'aplicació que es troba entre la Vista i el Model. Interactua amb totes 2 capes i en general la Vista i el Model només interactuen amb el Controlador.
- Respon als esdeveniments realitzats per l'usuari per mitjà de la Vista, i per tant rep i processa les peticions GET, POST, PUT, DELETE rebudes.
- El Controlador cridarà al Model i li demanarà que obtingui dades a partir de la BD.
- El Controlador rebrà les dades del Model i passarà a la Vista les dades obtingudes des del Model.
- El Controlador també pot ser responsable de carregar i enviar la Vista al programa client que habitualment serà un navegador.
- El Controlador es trobarà en el costat servidor i utilitza per exemple PHP.

f) En resum, la Vista mostra dades del Model a l'usuari, el usuari fa accions sobre la Vista amb l'ajut del client, es crea un esdeveniments que s'envia al Controlador, i a partir dels esdeveniments, el Controlador fa peticions al Model d'acord amb la sol·licitud, el Model realitza les operacions necessàries amb la BD i envia la resposta al Controlador que carrega una nova Vista amb les dades rebudes i l'envia al programa client perquè l'usuari pugui visualitzar la Vista.



g) Exemple molt senzill de codi utilitzant el patró MVC:

- Descarrega de <https://github.com/dacomo1920daw2/mvc.git> el codi d'una aplicació bàsica realitzada en PHP utilitzant el patró MVC.
- Primer hauràs de crear un usuari de MySQL, una base de dades i un taula amb l'ajut del fitxer **bd.sql**.
- Instal·la **vista.php**, **model.php** i **controlador.php** a l'arrel del teu servidor Web.
- Accedeix a **http://localhost/vista.php**