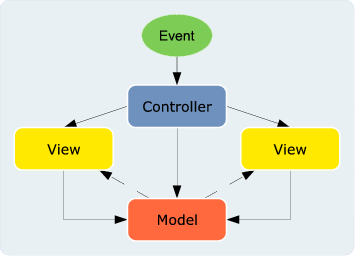
Arhitectura aplicatiei

MVC, sau Model-View-Controller este un sablon arhitectural folosit in industria de software development(inclusiv web development). Aceasta modalitate de lucru reuseste cu succes izolarea partii logice de interfata proiectului, rezultand in aplicatii extrem de usor de modificat. In organizarea MVC, modelul reprezinta informatia (datele) de care are nevoie aplicatia, viewerul corespunde cu elementele de interfata iar controller-ul reprezinta sistemul comunicativ si decizional ce proceseaza datele informationale, facand legatura intre model si view.



**Modelul** reprezinta partea de hard-programming, partea logica a aplicatiei. El are in responsabilitate actiunile si operatiile asupra datelor, autentificarea utilizatorilor, integrarea diverselor clase ce permit procesarea informatiilor din diverse baze de date.

In cadrul aplicatiei, modelul este reprezentat de clasa EntityController care se foloseste de clasa singleton Database care ii creeaza o conexiune cu baza de date . Prin aceasta conexiune, EntityController insereaza si extrage informatii din baza de date, acestea fiind transmise mai departe catre Controller.

**Controller**-ul reprezinta creierul aplicatiei. Aceasta face legatura intre model si view, intre actiunile userului si partea decizionala a aplicatiei. In functie de nevoile utilizatorului, controllerul apeleaza diverse functii definite special pentru sectiunea de site in care se afla userul. Functia se va folosi de model pentru a prelucra (extrage, actualiza) datele, dupa care informatiile noi vor fi trimise catre view.

Controller-ul este format din mai multe clase Servlet care trateaza evenimente distincte (Ex: in momentul in care un utilizator da submit unui formular de adaugare pentru un post, Servletul FormHandler va fi apelat, iar acesta va extrage datele din formularul trimis si le va insera in baza de date). De asemenea, controller-ul se foloseste de model pentru a extrage datele din baza de date care vor fi procesate adecvat si apoi trimise catre view.

**View**-ul se ocupa de afisarea datelor, practic aceasta parte a programului va avea grija de cum vede end-userul informatia procesata de controller. O data ce functiile sunt executate de model, viewului ii sunt oferite rezultatele, iar acesta le va trimite catre browser.