**Modello MVC**

Nel progetto di applicazioni Web in Java, due modelli di ampio uso e di riferimento:

* ***Model 1*** è un pattern semplice in cui codice responsabile per presentazione contenuti è mescolato con logica di business, Suggerito solo per piccole applicazioni (sta diventando obsoleto nella pratica industriale)



* ***Model 2,* *Model-View-Controller (MVC)*** come design pattern più complesso e articolato che separa chiaramente il livello ***presentazione*** dei contenuti dalla ***logica*** utilizzata per manipolare e processare i contenuti stessi



1. ***Controller*** – definisce il comportamento dell’applicazione

* fa *dispatching* di richieste utente e seleziona la *view* per la presentazione
* interpreta l’input dell’utente e lo mappa su azioni che devono essere eseguite da model

1. ***Model*** – rappresenta livello dei dati, incluse operazioni per accesso e modifica.

* possibilità per la *view* di interrogare stato di model
* possibilità per il *controller* di accedere alle funzionalità incapsulate dal model

1. ***View*** – si occupa di visualizzare i contenuti del model. Accede ai dati tramite il model e specifica come dati debbano essere presentati

* aggiorna la presentazione dei dati quando il model cambia
* Fornisce i moduli per l’invio dell’input dell’utente verso il controller da parte del browser

***Esecuzione:***

In applicazioni Web conformi al ***Model 2***, richieste del browser client vengono passate al ***Controller*** (implementato da una ***Servlet***).

Il ***Controller*** si occupa di eseguire la logica business necessaria per ottenere il contenuto da mostrare. Il ***Controller*** mette il contenuto nel ***Model*** (implementato con ***JavaBean***) in un messaggio e decide a quale ***View*** (implementata da ***JSP***) passare la richiesta.

La ***View*** si occupa del rendering del contenuto.