

JAVA/JEE

Le pattern MVC

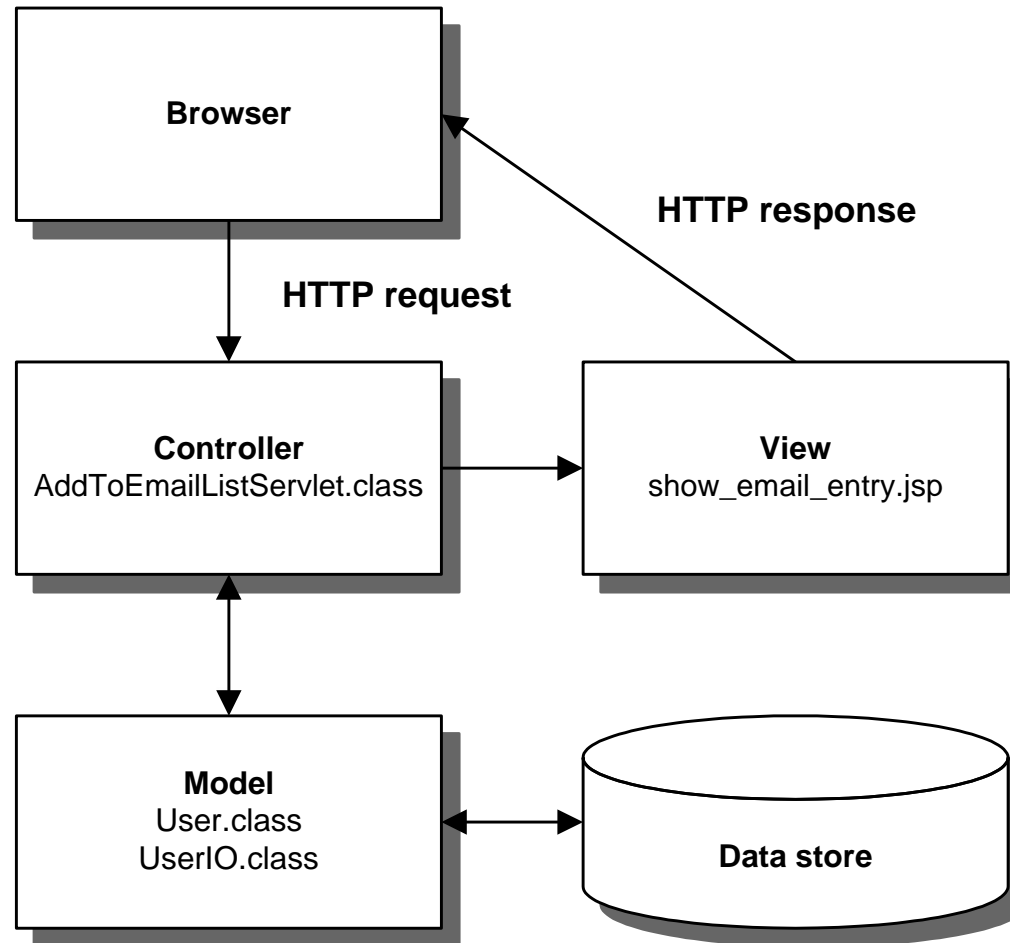


Pattern MVC

- **Modèle:** les données de l'application
- **Vue:** la manière de représenter ces données
- **Contrôleur:** comment répondre aux actions de l'utilisateur

- Pattern ultra classique pour la couche de présentation

Pattern MVC



Les principaux Framework

En JAVA:

- JSF (basé sur des actions comme swing)
- Tapestry
- Spring MVC

Hors du monde JAVA: RoR, Zend, Django, Symfony, AngularJS...

Implémentation

- La mise en place du pattern MVC ne nécessite pas forcément la mise en place d'un Framework additionnel en Java EE
 - La partie modèle est implémenté via les POJO retournés par la DAL
 - La partie contrôleur est implémenté par des Servlet
 - La partie vue est implémentée par des JSP
- Utilisation d'un dispatcher pour lier vues et contrôleurs
 - L'objet HttpRequest permet d'obtenir un dispatcher qui permet à un servlet de renvoyer une requête soit à un autre Servlet soit à une JSP
 - On pourra ainsi passer des paramètres directement à la JSP à travers l'objet HttpRequest

Exemple

```
protected void doPost(HttpServletRequest request, HttpServletResponse  
response) {  
    request.setAttribute("itemList", itemDao.find());  
  
    RequestDispatcher requestDispatcher =  
    request.getRequestDispatcher("/itemDisplay.jsp");  
  
    dispatcher.forward(request, response);  
}
```

Note: la classe dispatcher propose les méthodes forward et include

- La méthode forward permet de transmettre la requête vers un autre point d'entrée de manière transparente « comme si elle venait d'arriver »
- La méthode include conserve les modifications faites à la réponse et notamment au buffer d'écriture