

Présentation générale

Le terme « Java EE » signifie Java Enterprise Edition, et était anciennement raccourci en J2EE. La plate-forme Java EE est construite sur le langage Java et sur la plate-forme Java SE (Java Standard Edition) contenant un grand nombre de bibliothèques en plus de la bibliothèque d'origine. L'objectif majeur de Java EE est de faciliter **le développement d'applications web** robustes et distribuées, déployées et exécutées sur un **serveur d'applications** (Tomcat).

Le Java Enterprise Edition a été créé pour le développement d'applications d'entreprises. Ses spécifications ont été pensées afin, notamment, de faciliter le travail en équipe sur un même projet : l'application est découpée en couches.

Rappels architecture web simple

La communication entre le client et le serveur s'effectue au travers du protocole HTTP (voir la figure ci-dessus) :



1. l'utilisateur saisit une URL dans la barre d'adresses de son navigateur ;
2. le navigateur envoie alors une requête HTTP au serveur pour lui demander la page correspondante ;
3. le serveur reçoit cette requête, l'interprète et génère alors une page web qu'il va renvoyer au client au travers d'une réponse HTTP ;
4. le navigateur reçoit, via cette réponse, la page web finale, qu'il affiche alors à l'utilisateur dans son navigateur.

Ainsi :

- les données sont échangées entre le client et le serveur via le protocole HTTP ;
- le client (navigateur) ne comprend que les langages de présentation de l'information : HTML, CSS et Javascript ;
- les pages sont générées sur le serveur de manière dynamique, à partir du code source du site.
- Le serveur HTTP (Apache) écoute tout ce qui arrive sur le port utilisé par le protocole HTTP, le port 80 par défaut, et scrute chaque requête entrante.

Serveur d'applications

Mais un serveur doit être capable d'effectuer d'autres tâches, notamment exécuter des applications. Or un simple serveur HTTP n'en est pas capable. Il faut donc un autre composant qui va se charger d'exécuter du code en plus de faire le travail du serveur HTTP ; c'est le serveur d'applications. Nous utiliserons ici Tomcat de la fondation Apache. Un tel serveur inclut un serveur HTTP, et permet l'exécution de traitements dans des langages spécifiques (ici Java).

Le serveur d'applications aura pour rôle de :

- récupérer les requêtes HTTP issues des clients ;
- créer des objets à partir des données de la requête http;
- se connecter à la base de données et y exécuter des instructions
- renvoyer des réponses HTTP aux clients, en se basant sur les objets retournés par le code.

Il existe plusieurs serveurs d'application sur le marché :

- les solutions propriétaires et payantes : WebLogic (Oracle) et WebSphere (IBM),
- les solutions libres et gratuites : Apache Tomcat, JBoss, GlassFish et Jonas.

Le Modèle MVC

L'architecture JEE structure l'application en couches distinctes.

- tout ce qui concerne le traitement, le stockage et la mise à jour des données de l'application doit être contenu dans la couche nommée "Modèle" (le M de MVC) ;
- tout ce qui concerne l'interaction avec l'utilisateur et la présentation des données (mise en forme, affichage) doit être contenu dans la couche nommée "Vue" (le V de MVC) ;
- tout ce qui concerne le contrôle des actions de l'utilisateur et des données doit être contenu dans la couche nommée "Contrôle" (le C de MVC).

Modèle : des traitements et des données

Dans le modèle, on trouve à la fois les données et les traitements à appliquer à ces données. Ce bloc contient donc des objets Java d'une part, qui peuvent contenir des attributs (données) et des méthodes (traitements) qui leur sont propres, et un système capable de stocker des données d'autre part.

La couche DAO : Cette couche applicative permet d'accéder à la base de données et d'exécuter des instructions sql.

Vue : des pages JSP

Une page JSP est une page Html dans laquelle sont insérés des morceaux de code java. Elle est exécutée côté serveur.

Contrôleur : les servlets

La servlet permet de récupérer les informations stockées dans la requête http. Elle utilise la couche Modèle et DAO pour effectuer des traitements et récupérer des données de la base.

Cet objet n'agit jamais directement sur les données, il faut le voir comme un simple aiguilleur : il intercepte une requête issue d'un client, appelle éventuellement des traitements effectués par le modèle, et ordonne en retour à la vue d'afficher le résultat au client.

From:

<https://wiki.sio.bts/> - **WIKI SIO : DEPUIS 2017**

Permanent link:

<https://wiki.sio.bts/doku.php?id=presentjee>

Last update: **2020/07/26 16:27**

