

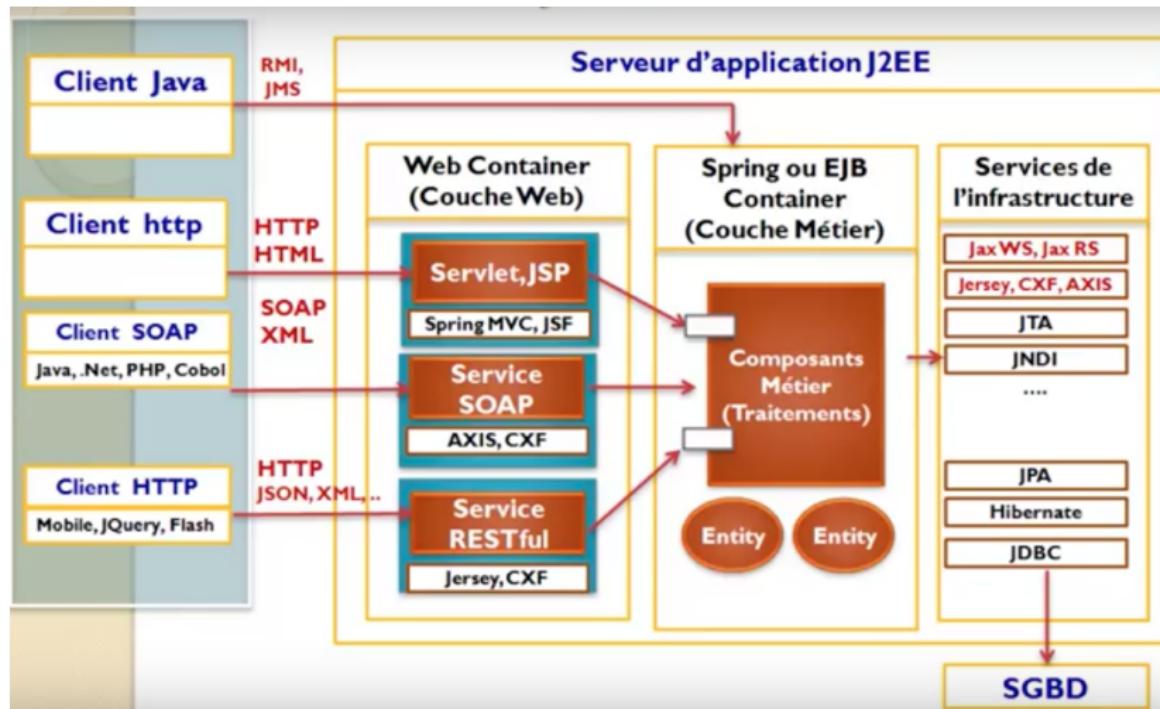
Architecture web

Yassamine Seladji

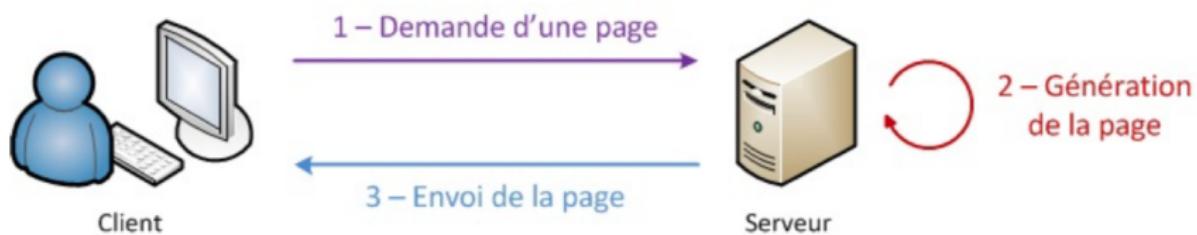
yassamine.seladji@gmail.com

1^{er} mars 2022

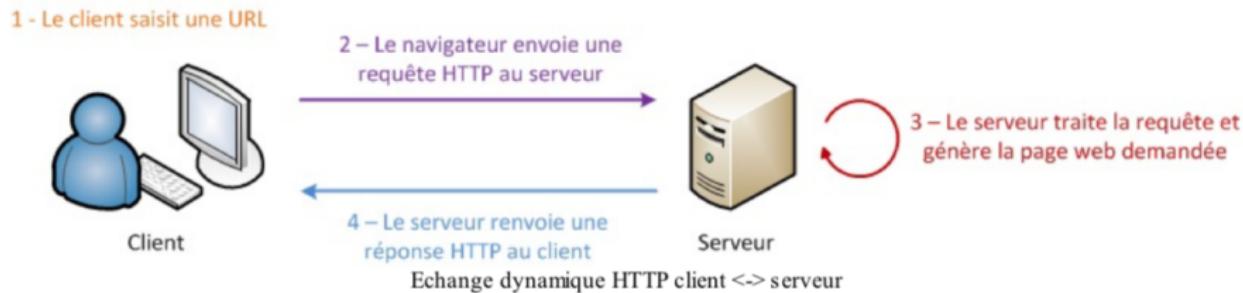
Introduction



Introduction



Introduction



Introduction

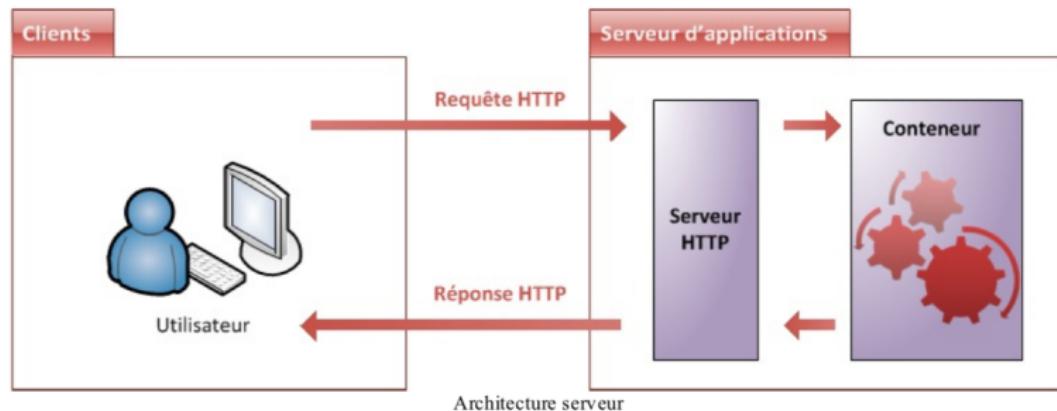
Une application web :

- ▶ échange entre client et serveur via internet.
- ▶ Internet désigne le réseau physique.
- ▶ Web désigne le contenu accessible via internet.
- ▶ La communication utilise le **protocole HTTP**.

Le protocole HTTP

- ▶ Un protocole de communication utiliser par les navigateurs.
- ▶ Elle utilise les méthodes suivantes :
 - ▶ Get : récupérer le contenu d'un document via une URL.
 - ▶ POST : envoyer des données via un formulaire.
 - ▶ PUT : envoyer un fichier du client vers le serveur.
 - ▶ DELETE : demander au serveur de supprimer un fichier.
 - ▶ HEAD : récupérer des informations sur un document.

Serveur d'application

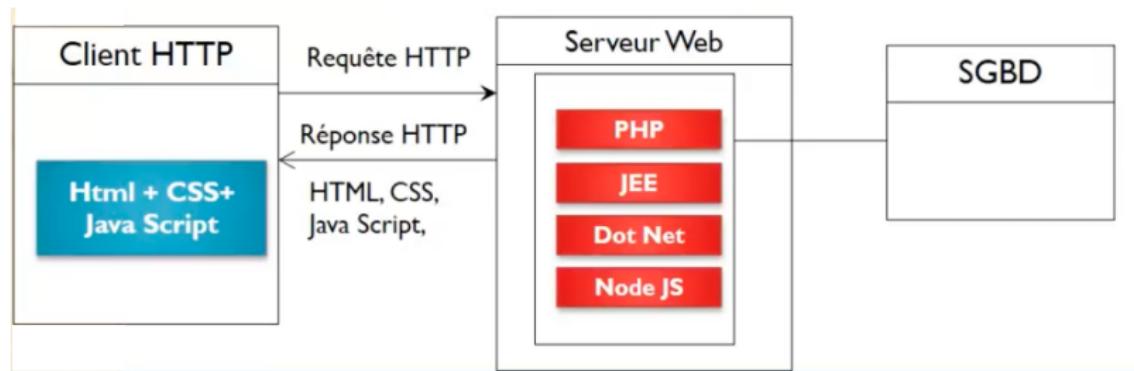


Serveur d'application

Le serveur d'application :

- ▶ Traite les requêtes HTTP.
- ▶ Transforme les données.
- ▶ Enregistre les données.

Architecture Web



Application web JEE

Une application web JEE (Java Entreprise Edition) :

- ▶ est une plateforme de développement d'applications web.
- ▶ est une technologie Java.
- ▶ JEE utilise un serveur d'application.

Les serveurs d'applications

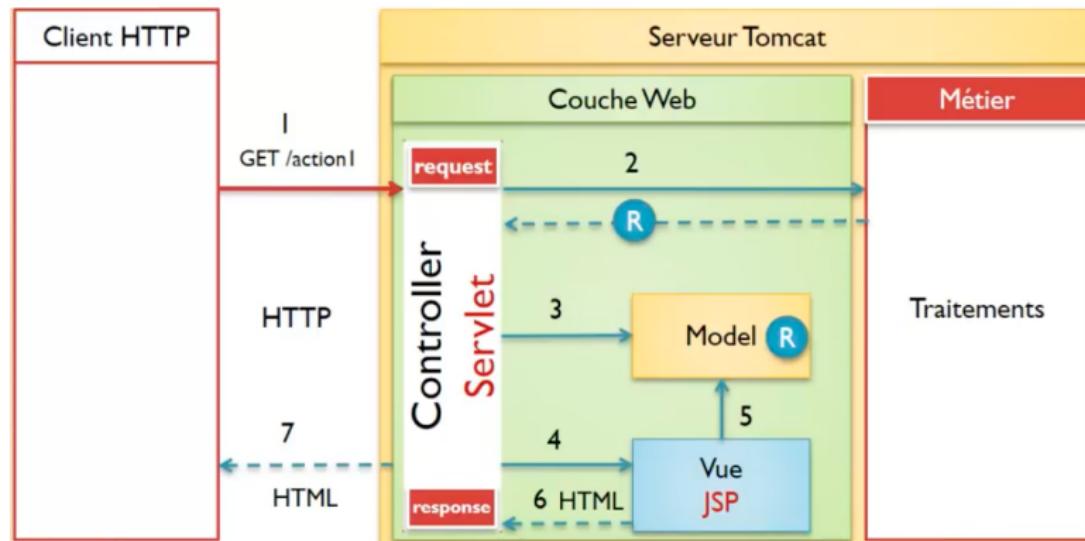
Les serveurs d'applications :

- ▶ payantes : WebLogic (Oracle), WebSphere (IBM).
- ▶ gratuites : Apache Tomcat, JBoss, GlassFish ...

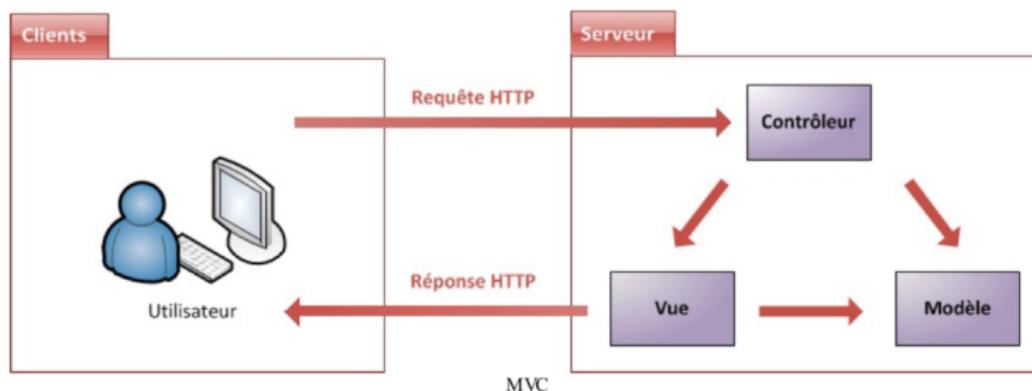
Principes de fonctionnement

- ▶ JEE se base sur le modèle MVC.
- ▶ Découper l'application en couches.
- ▶ Utiliser des frameworks prédéfinis : Spring, Struts ...

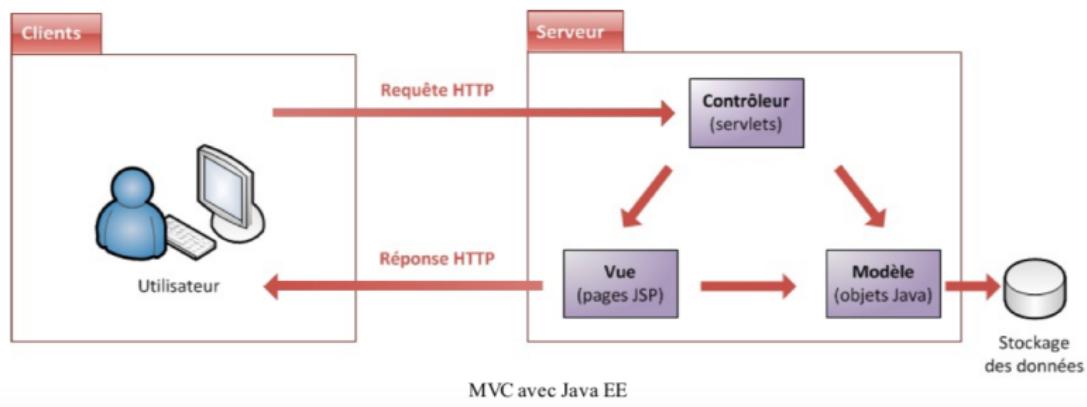
Principes de fonctionnement



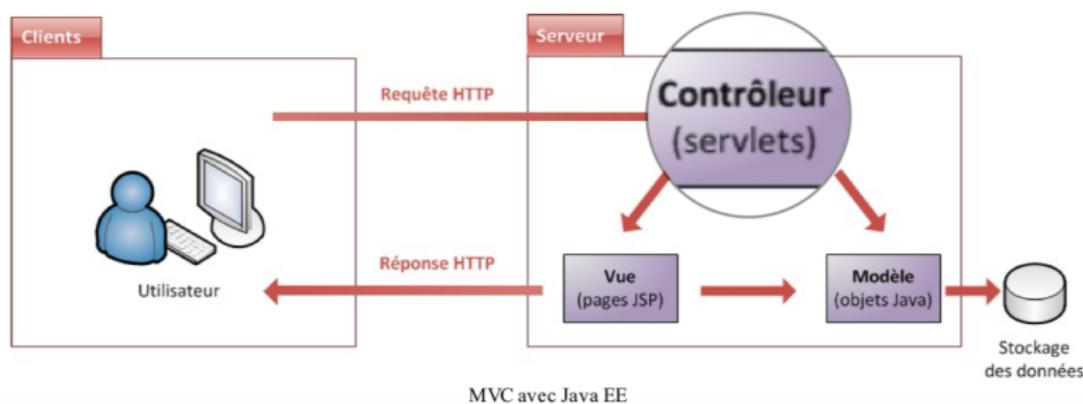
MVC



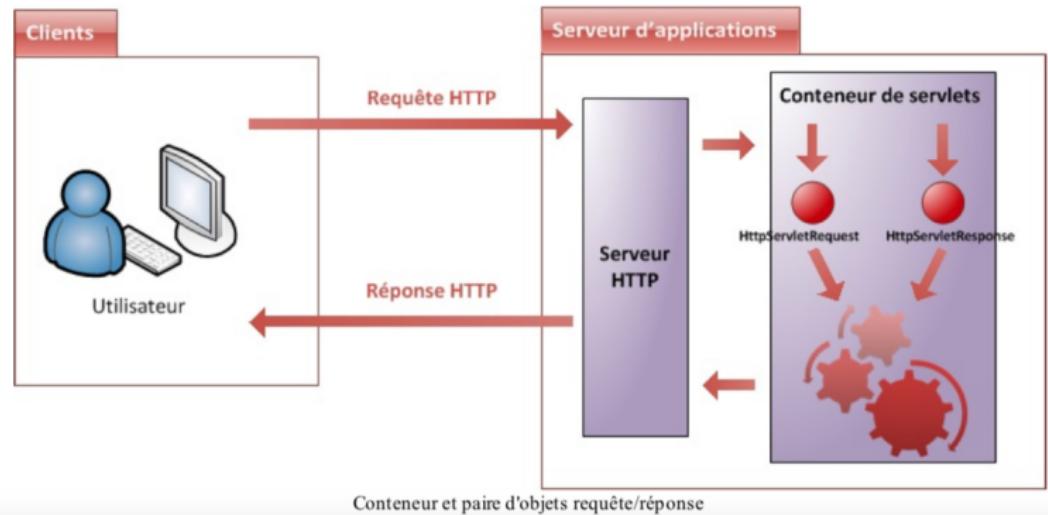
MVC et JEE



MVC et JEE



La servlet



La servlet

- ▶ La servlet permet de traiter les requêtes, dont les requête HTTP.
- ▶ Associer chaque type de requête HTTP à la méthode Java correspondante.
- ▶ La servlet est une classe Java qui hérite de la classe **HttpServlet**.

La servlet

- ▶ La servlet permet de traiter les requêtes, dont les requête HTTP.
- ▶ Associer chaque type de requête HTTP à la méthode Java correspondante.
- ▶ La servlet est une classe Java qui hérite de la classe **HttpServlet**.

```
import javax.servlet.http.HttpServlet;  
  
public class Test extends HttpServlet {  
}
```

La servlet

- ▶ La servlet est le contrôleur dans le modèle MVC.
- ▶ Chaque servlet correspond à une URL.
- ▶ Le mapping servlet/URL est défini dans le fichier **web.xml**.

La servlet

- ▶ La servlet est le contrôleur dans le modèle MVC.
- ▶ Chaque servlet correspond à une URL.
- ▶ Le mapping servlet/URL est défini dans le fichier **web.xml**.

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
    version="3.0">
    <servlet>
        <servlet-name>Test</servlet-name>
        <servlet-class>servlets.Test</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>Test</servlet-name>
        <url-pattern>/toto</url-pattern>
    </servlet-mapping>
</web-app>
```

La servlet

- ▶ Le mapping servlet/URL est défini en utilisant l'annotation @WebServlet avant la définition de la classe

```
@WebServlet(name="Test",urlPatterns="/toto")
```

La servlet

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
    version="3.0">
    <servlet>
        <servlet-name>Test</servlet-name>
        <servlet-class>servlets.Test</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>Test</servlet-name>
        <url-pattern>/toto</url-pattern>
    </servlet-mapping>
</web-app>
```

La servlet

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
    version="3.0">
    <servlet>
        <servlet-name>Test</servlet-name>
        <servlet-class>servlets.Test</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>Test</servlet-name>
        <url-pattern>/toto</url-pattern>
    </servlet-mapping>
</web-app>
```

http://localhost:8080/test/toto

La servlet

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
    xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
    version="3.0">
    <servlet>
        <servlet-name>Test</servlet-name>
        <servlet-class>servlets.Test</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>Test</servlet-name>
        <url-pattern>/toto</url-pattern>
    </servlet-mapping>
</web-app>
```

http://localhost:8080/test/toto

Etat HTTP 405 - La méthode HTTP GET n'est pas supportée par cette URL

Type Rapport d'état

message La méthode HTTP GET n'est pas supportée par cette URL.

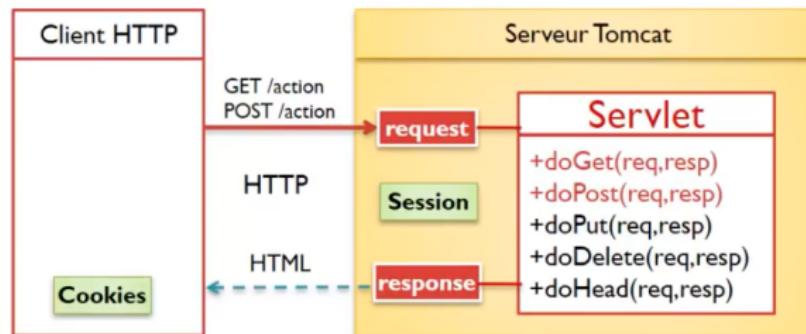
description La méthode HTTP spécifiée n'est pas autorisée pour la ressource demandée (La méthode HTTP GET n'est pas supportée par cette URL).

La servlet

- ▶ La servlet est une classe Java qui hérite de la classe **HttpServlet**.
- ▶ La servlet doit surcharger les méthodes DoXXX() : DoGet(), DoPost(), DoHead()...

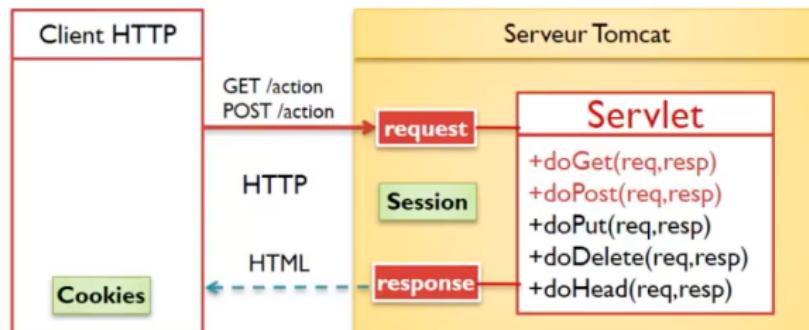
La servlet

- ▶ La servlet est une classe Java qui hérite de la classe **HttpServlet**.
- ▶ La servlet doit surcharger les méthodes DoXXX() : DoGet(), DoPost(), DoHead()...



La servlet

- ▶ La servlet est une classe Java qui hérite de la classe **HttpServlet**.
- ▶ La servlet doit surcharger les méthodes DoXXX() : DoGet(), DoPost(), DoHead()...



```
public class Test extends HttpServlet {  
    public void doGet( HttpServletRequest request, HttpServletResponse  
    response ) throws ServletException, IOException{  
    }  
}
```

La servlet : exemple

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException{
    response.setContentType("text/html");
    response.setCharacterEncoding( "UTF-8" );
    PrintWriter out = response.getWriter();
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<meta charset=\"utf-8\" />");
    out.println("<title>Test</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<p>Ceci est une page générée depuis une
servlet.</p>");
    out.println("</body>");
    out.println("</html>");
}
```

La servlet : exemple

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException{
    response.setContentType("text/html");
    response.setCharacterEncoding( "UTF-8" );
    PrintWriter out = response.getWriter();
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<meta charset=\"utf-8\" />");
    out.println("<title>Test</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<p>Ceci est une page générée depuis une
servlet.</p>");
    out.println("</body>");
    out.println("</html>");
}
```

http://localhost:8080/test/toto

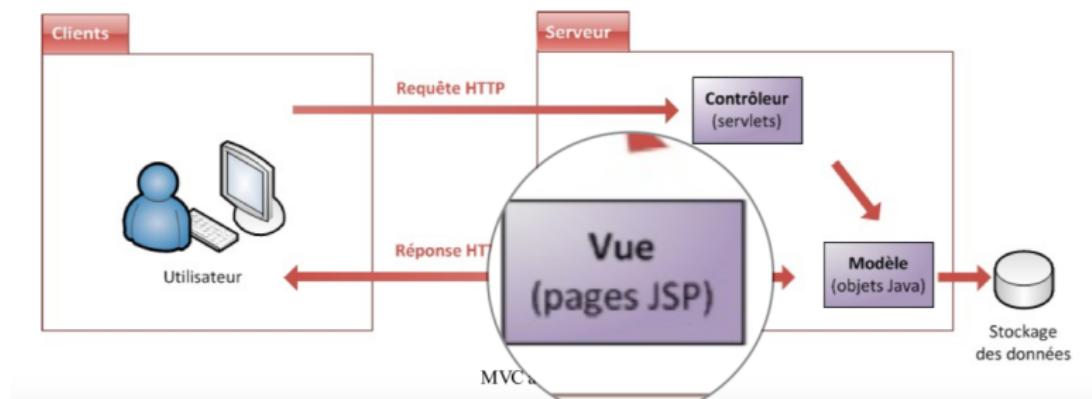
La servlet : exemple

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException{
    response.setContentType("text/html");
    response.setCharacterEncoding( "UTF-8" );
    PrintWriter out = response.getWriter();
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<meta charset=\"utf-8\" />");
    out.println("<title>Test</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<p>Ceci est une page générée depuis une
servlet.</p>");
    out.println("</body>");
    out.println("</html>");
}
```

http://localhost:8080/test/toto

Ceci est une page générée depuis une servlet.

Les JSP



Les JSP

Une JSP

- ▶ est un fichier avec une extension .jsp.
- ▶ contient des balises HTML et JSP.
- ▶ est exécuté côté serveur.
- ▶ définir des page web dynamique.

Les JSP

test.jsp

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Test</title>
    </head>

    <body>
        <p>Ceci est une page générée depuis une JSP.</p>
    </body>
</html>
```

Les JSP

test.jsp

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Test</title>
    </head>

    <body>
        <p>Ceci est une page générée depuis une JSP.</p>
    </body>
</html>
```

<http://localhost:8080/test/test.jsp>

Les JSP

test.jsp

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Test</title>
    </head>

    <body>
        <p>Ceci est une page générée depuis une JSP.</p>
    </body>
</html>
```

<http://localhost:8080/test/test.jsp>

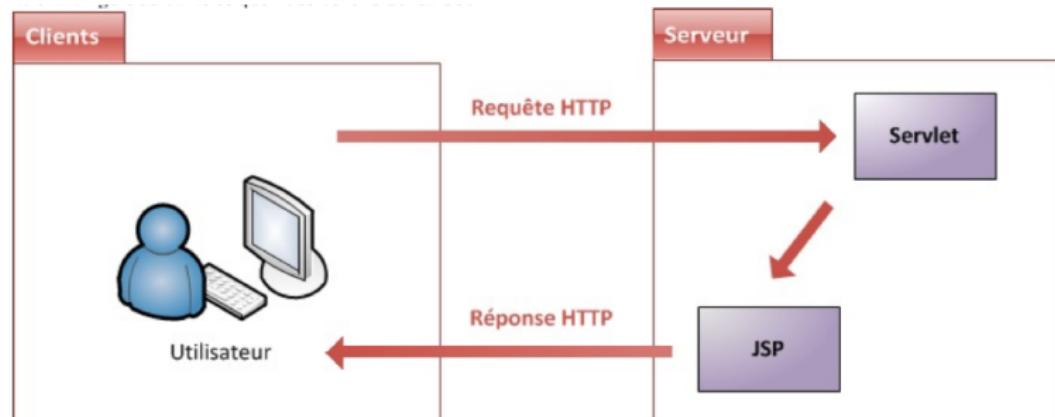
Ceci est une page générée depuis une JSP.

La servlet + JSP

Mapping servlet/JSP

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException {
    this.getServletContext().getRequestDispatcher( "/WEB-INF/test.jsp"
).forward( request, response );
}
```

La servlet + JSP



La servlet + JSP

La transmission de données côté **servlet**.

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException{
    String message = "Transmission de variables : OK !";
    request.setAttribute( "test", message );
    this.getServletContext().getRequestDispatcher( "/WEB-INF/test.jsp"
).forward( request, response );
}
```

La servlet + JSP

La transmission de données côté JSP.

```
<%@ page pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Test</title>
    </head>
    <body>
        <p>Ceci est une page générée depuis une JSP.</p>
        <p>
            <%
                String attribut = (String) request.getAttribute("test");
                out.println( attribut );
            %}
        </p>
    </body>
</html>
```

La servlet + JSP

La transmission de données côté **client**.

```
<!-- URL sans paramètres -->
/page.jsp

<!-- URL avec un paramètre nommé 'cat' et ayant pour valeur 'java'
-->
/page.jsp?cat=java

<!-- URL avec deux paramètres nommés 'lang' et 'admin', et ayant
pour valeur respectivement 'fr' et 'true' -->
/page.jsp?lang=fr&admin=true
```

La servlet + JSP

La transmission de données exemple :

`http://localhost:8080/test/toto?auteur=client1`

La servlet + JSP

La transmission de données exemple :

http://localhost:8080/test/toto?auteur=client1

Côté servlet :

```
public void doGet( HttpServletRequest request, HttpServletResponse
response ) throws ServletException, IOException{
String paramAuteur = request.getParameter( "auteur" );
String message = "Transmission de variables : OK ! " + paramAuteur;
request.setAttribute( "test", message );

this.getServletContext().getRequestDispatcher( "/WEB-INF/test.jsp"
).forward( request, response );
}
```

La servlet + JSP

La transmission de données exemple :

`http://localhost:8080/test/toto?auteur=client1`

La servlet + JSP

La transmission de données exemple :

http://localhost:8080/test/toto?auteur=client1

Côté JSP :

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8" />
        <title>Test</title>
    </head>
    <body>
        <p>Ceci est une page générée depuis une JSP.</p>
        <p>
            Transmission de variables : OK ! Client1
        </p>
    </body>
</html>
```

Application

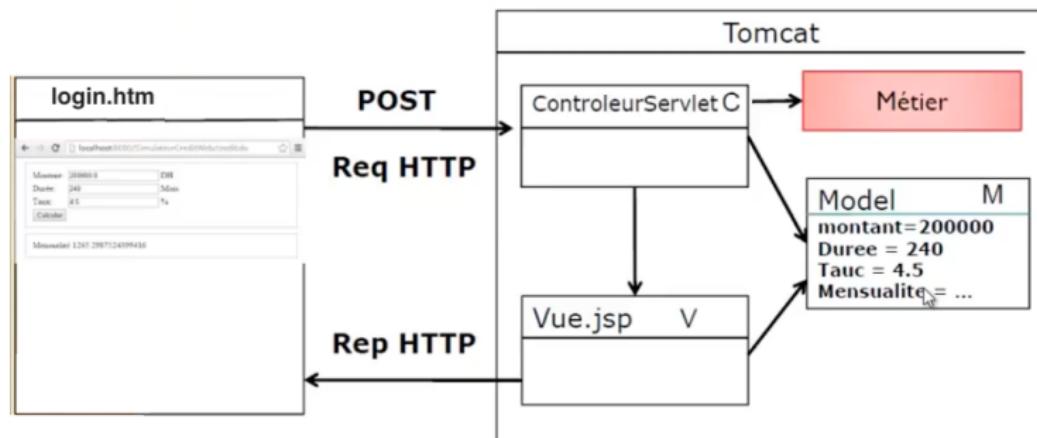
Créer une application web JEE qui calcule la mensualité m d'un crédit bancaire en fonction :

- ▶ du montant emprunté C .
- ▶ de la durée de l'emprunt en mois.
- ▶ du taux effectif t (en pourcentage).

La formule :

$$m = \frac{C \frac{t}{12}}{1 - (1 + \frac{t}{12})^{-n}}$$

Application



Application

```
public class CreditMetierImpl implements ICreditMetier {  
    @Override  
    public double calculerMesualiteCredit(double capital, double taux, int duree) {  
        double t=taux/100;  
        double t1=capital*t/12;  
        double t2=1-Math.pow(1+t/12, -duree);  
        return t1/t2;]  
    }  
}
```

Application

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xml
  <display-name>banque</display-name>
  <servlet>
    <servlet-name>cs</servlet-name>
    <servlet-class>web.ControleurServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>cs</servlet-name> I
    <url-pattern>*.do</url-pattern>
  </servlet-mapping>
</web-app>
```

Application

```
<div>
    <form action="calculerMensualite.do" method="post">
        <table>
            <tr>
                <td>Montant:</td>
                <td><input type="text" name="montant" /></td>
                <td>DH</td>
            </tr>
            <tr>
                <td>Taux:</td>
                <td><input type="text" name="taux" /></td>
                <td>%</td>
            </tr>
            <tr>
                <td>Durée:</td>
                <td><input type="text" name="duree" /></td>
                <td>mois</td>
            </tr>
        </table>
        <button type="submit">Calculer</button>
    </form>

```

Application

The screenshot shows a web browser window with the following details:

- Title Bar:** Crédit bancaire
- Address Bar:** localhost:8080/banque/test.do

The main content area displays a form for calculating a bank loan:

Montant: DH

Taux:

Durée: mois

Calculator button

Application

```
public class CreditModel {  
    private double montant;  
    private double taux;  
    private int duree;  
    private double mensualite;  
    public double getMontant() {  
        return montant;  
    }  
    public void setMontant(double montant) {  
        this.montant = montant;  
    }  
    public double getTaux() {  
        return taux;  
    }  
    public void setTaux(double taux) {  
        this.taux = taux;  
    }  
}
```

Application

```
@WebServlet(name="cs",urlPatterns= {"/contrroleur","*.do"})
public class ContrroleurServlet extends HttpServlet {
    @Override
    public void init() throws ServletException {
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        request.getRequestDispatcher("VueCredit.jsp").forward(request, response);
    }
}
```

Application

```
@Override  
protected void doPost(HttpServletRequest request, HttpServletResponse response)  
    double montant=Double.parseDouble(request.getParameter("montant"));  
    double taux=Double.parseDouble(request.getParameter("taux"));  
    int duree =Integer.parseInt(request.getParameter("duree"));  
  
    CreditModel model=new CreditModel();  
    model.setMontant(montant);  
    model.setTaux(taux);  
    model.setDuree(duree);  
  
    double res=metier.calculerMesualiteCredit(montant, taux, duree);  
    model.setMensualite(res);|
```

Application

```
/*
 * Faire appel à la couche métier pour effectuer les traitements
 */
double res=metier.calculerMesualiteCredit(montant, taux, duree);
/*
 * Stocker les résultats dans le modèle
 */
model.setMensualite(res);
/*
 * Stocker le modèle dans l'objet request
 */
request.setAttribute("creditModel", model);

/*
 * Faire un forward vers la vue JSP
 */
request.getRequestDispatcher("VueCredit.jsp").forward(request, response);
```

Application

```
<%@page import="web.CreditModel"%>
<%
    CreditModel model=request.getAttribute("creditModel");
%>
<!DOCTYPE html>
<html>
<head>
    <title>Crédit bancaire</title>
    <meta charset="utf-8">
</head>
<body>
    <div>
        <form action="calculerMensualite.do" method="post">
            <table>
                <tr>
                    <td>Montant:</td>
                    <td><input type="text" name="montant" /></td>
                    <td>DH</td>
                </tr>
                <tr>
                    <td>Taux:</td>
                    <td><input type="text" name="taux" /></td>
                    <td>%</td>
                </tr>
            </table>
        </form>
    </div>
</body>
</html>
```

Application

```
<td>Montant:</td>
<td><input type="text" name="montant" /></td>
<td>DH</td>
</tr>
<tr>
    <td>Taux:</td>
    <td><input type="text" name="taux" /></td>
    <td>%</td>
</tr>
<tr>
    <td>Durée:</td>
    <td><input type="text" name="duree" /></td>
    <td>mois</td>
</tr>
</table>
<button type="submit">Calculer</button>
</form>
</div>
<p></p>
<div>
    Mensualité = <%=model.getMensualite()%>
</div>
</body>
```