

Application Web Java MVC : Servlets, JSP, standard tags

Cas: E-Banque

Diagramme des cas d'utilisation

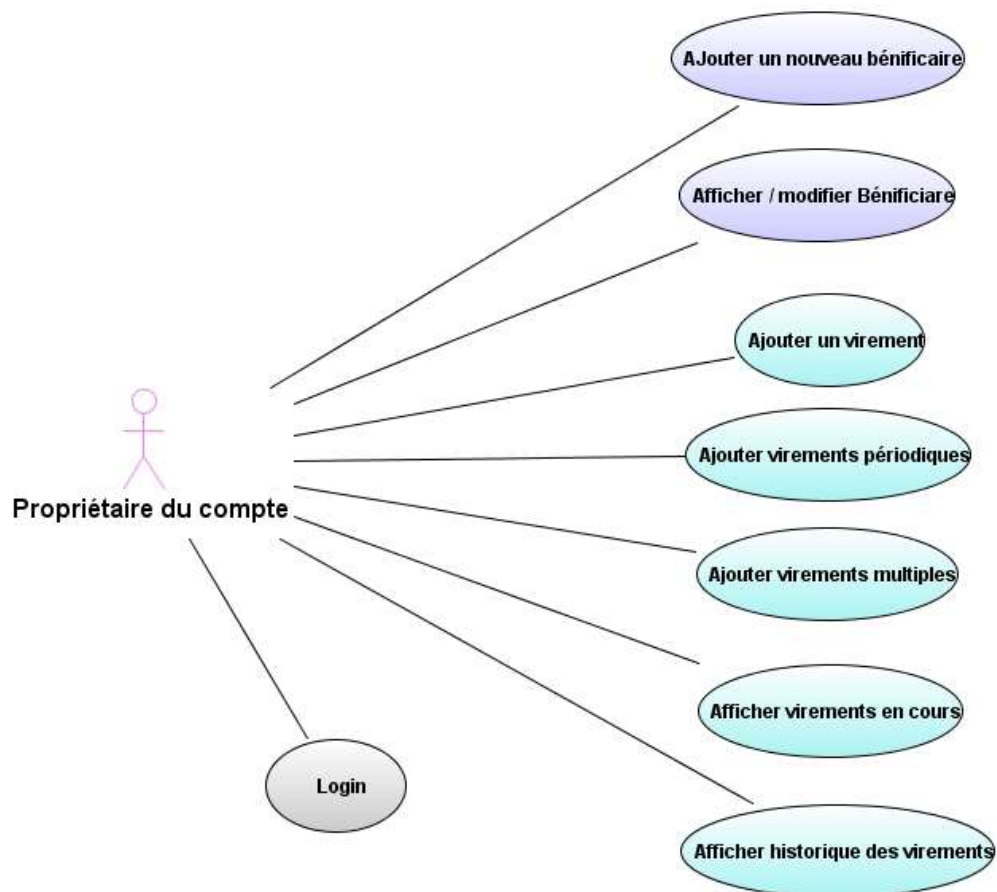
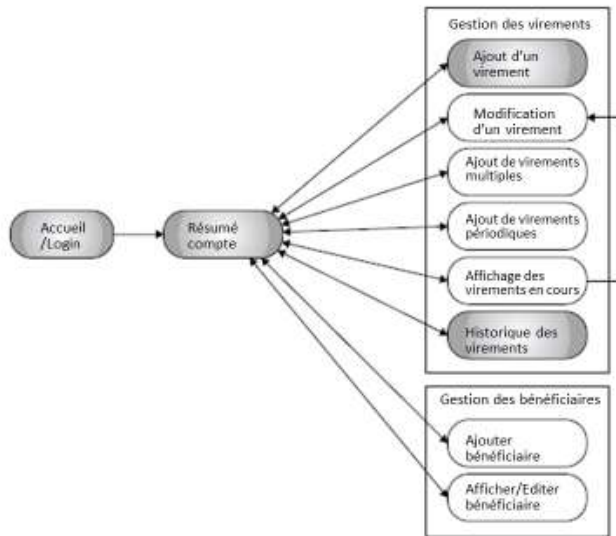


Diagramme de navigation



Ecrans de l'application

/Login.jsp

Nom:

Mot de passe:

/Compte.jsp	
Ajouter un virement Ajouter virements multiples Ajouter virements périodiques Virements en cours Historique des virements Ajouter bénéficiaire Afficher/modifier bénéficiaire	Solde: XX XXX

/ajouter/virement.jsp

Bénéf: montant:














Date d'envoi:

Memo:







/historique.jsp

Bénéf	Date	Montant

Organisation des packages de l'application

- ▼  app_mvc
 - >  app_mvc
 - >  JAX-WS Web Services
 - ▼  Java Resources
 - ▼  src
 - ▼  banque
 - >  Connexion.java
 - ▼  banque.controllers
 - >  BanqueController.java
 - ▼  banque.dal
 - >  BanqueDAO.java
 - ▼  banque.model
 - >  Banque.java

Les vues

- ▼  WebContent
 - >  META-INF
 - >  WEB-INF
 -  banque.jsp
 -  banques.jsp
 -  modifierBanque.jsp

La classe Connexion

```
package banque;

import java.sql.Connection;

public class Connexion {

    private static Connection cnx;

    static {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            cnx = DriverManager.getConnection("jdbc:mysql://localhost/banque", "root", "");
            if (cnx == null)
                System.out.println("connexion!!!");
        }

        catch (SQLException e) {
            System.out.println("Erreur SQL:" + e.getMessage());
            e.printStackTrace();
        } catch (ClassNotFoundException e) {

            System.out.println("Erreur Driver:" + e.getMessage());
        }
    }

    static public Connection getConnexion() {
        return cnx;
    }
}
```

La classe Banque

```
package banque.model;

public class Banque {
    private int numero;

    private String nom;

    public int getNumero() {
        return numero;
    }

    public void setNumero(int numero) {
        this.numero = numero;
    }

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom = nom;
    }
}
```

La classe BanqueDAO

d

```
package banque.dal;

import java.sql.Connection;

public class BanqueDAO {
    static private Connection cnx = Connexion.getConnexion();

    public static void ajouter(Banque b) {
        if (cnx == null)
            System.out.println("Connexion nulle");
        String sql = "insert into banques values(?,?)";
        try {

            PreparedStatement pst = cnx.prepareStatement(sql);
            pst.setInt(1, b.getNumero());
            pst.setString(2, b.getNom());

            pst.executeUpdate();

            pst.close();
        } catch (SQLException e) {

            System.out.println("Erreur SQL:" + e.getMessage());
        }
    }

    public static void modifier(Banque b) {
        String sql = "update banques set nom=? where numero=?";
        try {
            PreparedStatement pst = cnx.prepareStatement(sql);
            pst.setString(1, b.getNom());
            pst.setInt(2, b.getNumero());
            pst.executeUpdate();
        } catch (SQLException e) {
            System.out.println("Erreur SQL:" + e.getMessage());
        }
    }

    public static void supprimer(int numero) {
        String sql = "delete from banques where numero=?";
        try {
            PreparedStatement pst = cnx.prepareStatement(sql);
            pst.setInt(1, numero);
            pst.executeUpdate();
        } catch (SQLException e) {
            System.out.println("Erreur SQL:" + e.getMessage());
        }
    }
}
```

```
public static Banque get(int numero) {
    String sql = "select * from banques Where numero=?";

    try {
        PreparedStatement pst = cnx.prepareStatement(sql);
        pst.setInt(1, numero);
        ResultSet rs = pst.executeQuery();

        Banque b = null;
        if (rs.next()) {
            b = new Banque();
            b.setNom(rs.getString("nom"));
            b.setNumero(rs.getInt("numero"));
            return b;
        }
        pst.close();
    } catch (SQLException e) {
        System.out.println("Erreur SQL:" + e.getMessage());
    }

    return null;
}

public static List<Banque> get() {
    String sql = "select * from banques";
    List<Banque> banques = new ArrayList<>();
    try {
        PreparedStatement pst = cnx.prepareStatement(sql);
        ResultSet rs = pst.executeQuery();

        Banque b = null;
        while (rs.next()) {
            b = new Banque();
            b.setNom(rs.getString("nom"));
            b.setNumero(rs.getInt("numero"));
            banques.add(b);
        }
        pst.close();
    } catch (SQLException e) {
        System.out.println("Erreur SQL:" + e.getMessage());
    }

    return banques;
}
```

La classe BanqueController

```
package banque.controllers;

import java.io.IOException;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import banque.dal.BanqueDAO;
import banque.model.Banque;

@WebServlet("/banques.do")
public class BanqueController extends HttpServlet {

    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String action = "list";
        if (request.getParameter("action") != null)
            action = request.getParameter("action");
        String url = "";
        switch (action) {
            case "list":
                List<Banque> banques = BanqueDAO.get();
                request.setAttribute("banques", banques);
                url = "banques.jsp";
                break;
            case "ajout":
                if (request.getParameter("annuler") != null)
                    url = "banques.do?action=list";

                else if (request.getParameter("enregistrer") != null) {
                    Banque banque = new Banque();
                    banque.setNumero(Integer.parseInt(request.getParameter("numero")));
                    banque.setNom(request.getParameter("nom"));
                    BanqueDAO.ajouter(banque);
                    url = "banques.do?action=list";
                } else {
                    url = "banque.jsp";
                }
                break;
        }

        case "modif":
            if (request.getParameter("nom") == null) {
                int num = Integer.parseInt(request.getParameter("numero"));
                Banque banque = BanqueDAO.get(num);
                request.setAttribute("banque", banque);
                url = "modifierBanque.jsp";
            } else { // Modification
                Banque banque = new Banque();
                banque.setNumero(Integer.parseInt(request.getParameter("numero")));
                banque.setNom(request.getParameter("nom"));
                BanqueDAO.modifier(banque);
                url = "banques.do?action=list";
            }
            break;

        case "suppr":
            if (request.getParameter("numero") != null) {
                int num = Integer.parseInt(request.getParameter("numero"));
                BanqueDAO.supprimer(num);
            }
            url = "banques.do?action=list";
            break;
        }

        request.getRequestDispatcher(url).forward(request, response);
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        doGet(request, response);
    }
}
```

Les vues

Banques.jsp

```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1" import="banque.model.Banque, java.util.List"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7 <title>Liste des banques</title>
8 </head>
9 <body>
10 <h2>Liste des banques</h2>
11 <a href="banques.do?action=ajout">Ajouter une banque</a>
12 <table border="1">
13 <tr><th>Numéro</th><th>Nom</th><th></th><th></th></tr>
14 <%
15 List<Banque> banques=(List<Banque> ) request.getAttribute("banques");
16 for(Banque b: banques){
17   %>
18   %>
19   <tr>
20 <td><%=b.getNumero() %></td>
21 <td><%=b.getNom()%></td>
22 <td><a href="banques.do?action=modif&numero=<%=b.getNumero() %>">Modifier</a></td>
23 <td><a href="banques.do?action=suppr&numero=<%=b.getNumero() %>">Supprimer</a></td>
24 </tr>
25 <%
26   }
27   %>
28 </table>
29 </body>
30 </html>
```

Banque.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
  pageEncoding="ISO-8859-1"%>
<!DOCTYPE html >
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Ajouter une banque</title>
</head>
<body>
  <form action="banques.do">
    <input type="hidden" name="action" value="ajout" /> Numéro:<input
      type="text" name="numero" /><br> Nom:<input type="text"
      name="nom" /><br> <input type="submit" name="enregistrer"
      value="Enregistrer" /> <input type="submit" name="annuler"
      value="Annuler" />
  </form>
</body>
</html>
```

modifierBanque.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html >
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Modifier une banque</title>
</head>
<body>
<form action="banques.do" >
<input type="hidden" name="action" value="modif"/>
Numéro:<input type="text" value="${banque.numero }" name="numero" readonly/><br>
Nom:<input type="text" value="${banque.nom }" name="nom"/><br>
<input type="submit" value="Modifier"/>

</form>
</body>
</html>
```