

# Technologie JEE

## Rapport TP N°1 : JSF- Java Server Faces

Réalisé par :

Abdelghani MOUSSAID

Encadré par :

Pr. Noredidine GHERABI

# 1. Création d'un projet JSF :

- Ajout d'un projet Web dynamique << JSF >>

The screenshot shows the 'New Dynamic Web Project' dialog box in the Eclipse IDE. The dialog is titled 'New Dynamic Web Project' and has a subtitle 'Dynamic Web Project'. Below the subtitle, it says 'Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.' The dialog contains several sections with input fields and buttons:

- Project name:** A text field containing 'JSF'.
- Project location:** A section with a checked checkbox 'Use default location'. Below it, a text field shows the location 'C:\development\eclipse-workspace\JSF' and a 'Browse...' button.
- Target runtime:** A dropdown menu showing '<None>' and a 'New Runtime...' button.
- Dynamic web module version:** A dropdown menu showing '3.0'.
- Configuration:** A dropdown menu showing 'JavaServer Faces v2.2 Project' and a 'Modify...' button. Below it, a note says 'Configures a Dynamic Web application to use JSF v2.2'.
- EAR membership:** A section with an unchecked checkbox 'Add project to an EAR'. Below it, a text field shows 'JSFEAR' and a 'New Project...' button.
- Working sets:** A section with an unchecked checkbox 'Add project to working sets' and a 'New...' button. Below it, a text field is empty and a 'Select...' button is present.

At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Finish' (highlighted with a blue border), and 'Cancel'.

- Ajout des librairies nécessaires

```

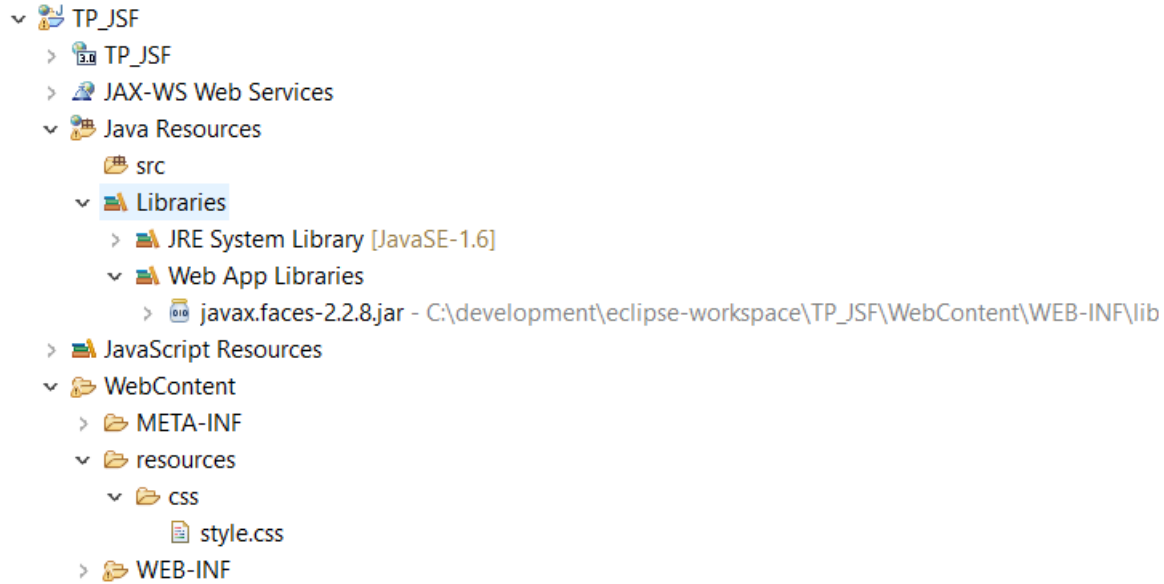
v TP_JSF
  > TP_JSF
  > JAX-WS Web Services
  v Java Resources
    src
    v Libraries
      > JRE System Library [JavaSE-1.6]
      v Web App Libraries
        > javax.faces-2.2.8.jar - C:\development\eclipse-workspace\TP_JSF\WebContent\WEB-INF\lib
  
```

- Création d'un fichier index.xhtml pour la saisie des informations d'une commande

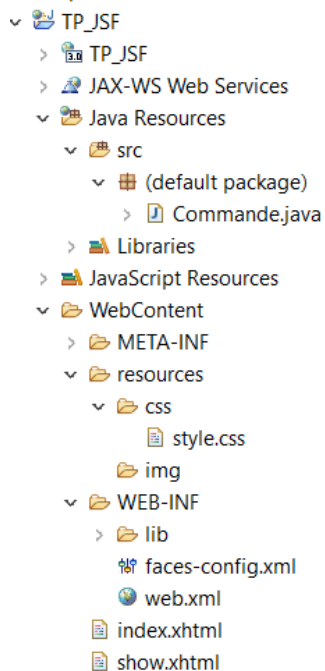
```

<div id="content">
<h:form>
  <h:outputText value="Date " /><h:outputLabel style="color:orange" value="*" />
  <h:inputText styleClass="inputs" p:placeholder="(MM-DD-YYYY)" value="#{date}" />
  <h:outputText value="Montant " /><h:outputLabel style="color:orange" value="*" />
  <h:inputText styleClass="inputs" value="#{montant}" />
  <h:outputText value="Mode de paiement " /><h:outputLabel style="color:orange" value="*" />
  <h:inputText styleClass="inputs" value="#{modePaiement}" />
  <h:outputText value="Statut de paiement " />
  <h:inputText styleClass="inputs" value="#{statutPaiement}" />
  <h:outputText value="Mode de Livraison " /><h:outputLabel style="color:orange" value="*" />
  <h:inputText styleClass="inputs" value="#{modeLivraison}" />
  <h:outputText value="Statut de Livraison " />
  <h:inputText styleClass="inputs" value="#{statutLivraison}" />
  <h:commandButton styleClass="submit" value="Submit" action="show"
    style="height:3rem; width:100%; font-size:1.5rem; color:white; background-color: #4CAF50;" />
</h:form>
</div>
  
```

- Insertion des feuilles de styles



- Structure du projet :



## 2. Affichage, Validation et conversion des données :

a) Création de la classe `Commande.java` :

Commande.java

```
1 import java.util.Date;
2 import javax.faces.bean.ManagedBean;
3 import javax.faces.bean.SessionScoped;
4
5 @ManagedBean
6 @SessionScoped
7 public class Commande {
8
9     private Date date;
10    private Long montant = null;
11    private String modePaiement;
12    private String statutPaiement;
13    private String modelivraison;
14    private String statutLivraison;
15
16    public Commande() {
17        super();
18        // TODO Auto-generated constructor stub
19    }
20
21    public Commande(Date date, Long montant, String modePaiement, String statutPaiement, String modelivraison,
22        String statutLivraison) {
23        super();
24        this.date = date;
25        this.montant = montant;
26        this.modePaiement = modePaiement;
27        this.statutPaiement = statutPaiement;
28        this.modelivraison = modelivraison;
29        this.statutLivraison = statutLivraison;
30    }
31
32    public Date getDate() {
33        return date;
34    }
35
36    public void setDate(Date date) {
37        this.date = date;
38    }
39
40    public Long getMontant() {
41        return montant;
42    }
43}
```

Commande.java

```
44 public void setMontant(Long montant) {
45     this.montant = montant;
46 }
47
48 public String getModePaielement() {
49     return modePaielement;
50 }
51
52 public void setModePaielement(String modePaielement) {
53     this.modePaielement = modePaielement;
54 }
55
56 public String getStatutPaielement() {
57     return statutPaielement;
58 }
59
60 public void setStatutPaielement(String statutPaielement) {
61     this.statutPaielement = statutPaielement;
62 }
63
64 public String getModelivraison() {
65     return modelivraison;
66 }
67
68 public void setModelivraison(String modelivraison) {
69     this.modelivraison = modelivraison;
70 }
71
72 public String getStatutLivraison() {
73     return statutLivraison;
74 }
75
76 public void setStatutLivraison(String statutLivraison) {
77     this.statutLivraison = statutLivraison;
78 }
79
80 @Override
81 public String toString() {
82     return "Commande [date=" + date + ", montant=" + montant + ", modePaielement=" + modePaielement
83         + ", statutPaielement=" + statutPaielement + ", modelivraison=" + modelivraison +
84         ", statutLivraison=" + statutLivraison + "]";
85 }
86
```

b) Validation et Conversion des données :

Date :

```
<h:outputText styleClass="outputs" value="Date " />
<h:outputLabel style="color:orange" value="*" />
<h:inputText id = "date"
    styleClass = "inputs"
    label = "Date"
    p:placeholder = "(DD-MM-YYYY)"
    value="#{commande.date}"
    required="true">
<c:convertDateTime pattern="dd-MM-yyyy" />
</h:inputText>
<h:message styleClass="error" for="date" /><br />
```

Montant :

```
<h:outputText styleClass="outputs" value="Montant " />
<h:outputLabel style="color:orange" value="*" />
<h:inputText id="montant"
    label = "Montant"
    styleClass="inputs"
    value="#{commande.montant}"
    p:type="number"
    p:min="0"
    p:max="20000"
    required="true"
    validateMessage="Champs Requis!">
</h:inputText>
<h:message styleClass="error" for="montant" /><br />
```

Mode de paiement :

```
<h:outputText styleClass="outputs" value="Mode de paiement " />
<h:outputLabel style="color:orange" value="*" />
<h:inputText styleClass="inputs"
    id="mode-paiement"
    label="Mode Paiement"
    value="#{commande.modePaiement}"
    required="true"
    validateMessage="Veuillez entrer une chaine entre 6 et 20 carctères!">
<c:validateLength minimum="6" maximum="20" />
</h:inputText>
<h:message styleClass="error" for="mode-paiement" /><br />
```



Mode de livraison :

```
<h:outputText styleClass="outputs" value="Mode de Livraison" />
<h:outputLabel style="color:orange" value="*" />
<h:inputText styleClass="inputs" value="#{commande.modeLivraison}"
    id="mode-livraison"
    label="Mode Livraison"
    required="true">
<c:validateLength minimum="0" maximum="30" />
</h:inputText>
<h:message styleClass="error" for="mode-livraison" /><br />
```

c) Définir les règles de styles dans le fichier css :

```
» .inputs {
    width: 100%;
    padding: 12px 20px;
    margin: 8px 0;
    display: flex;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
» .outputs {
    font-weight : bold;
    font-size : large;
}
» #content {
    border-radius: 5px;
    background-color: #f2f2f2;
    padding: 20px;
}
» .final {
    background-color : #408080;
    color : white;
}
» .error {
    display : table-row;
    color: green;
    font-style : italic;
}
```

d) Afficher les données avec un nouveau fichier << show.xhtml >>

```
<h:outputLabel styleClass="outputs" value="Date " />
<h:outputText styleClass="inputs final" value="#{commande.date}">
<c:convertDateTime dateStyle="full" locale="fr_FR" />
</h:outputText>

<h:outputLabel styleClass="outputs" value="Montant " />
<h:outputText styleClass="inputs final" value="#{commande.montant}" >
  <c:convertNumber type="currency" currencySymbol="DH" />
</h:outputText>

<h:outputLabel styleClass="outputs" value="Mode de paiement " />
<h:outputText styleClass="inputs final" value="#{commande.modePaiement}" />

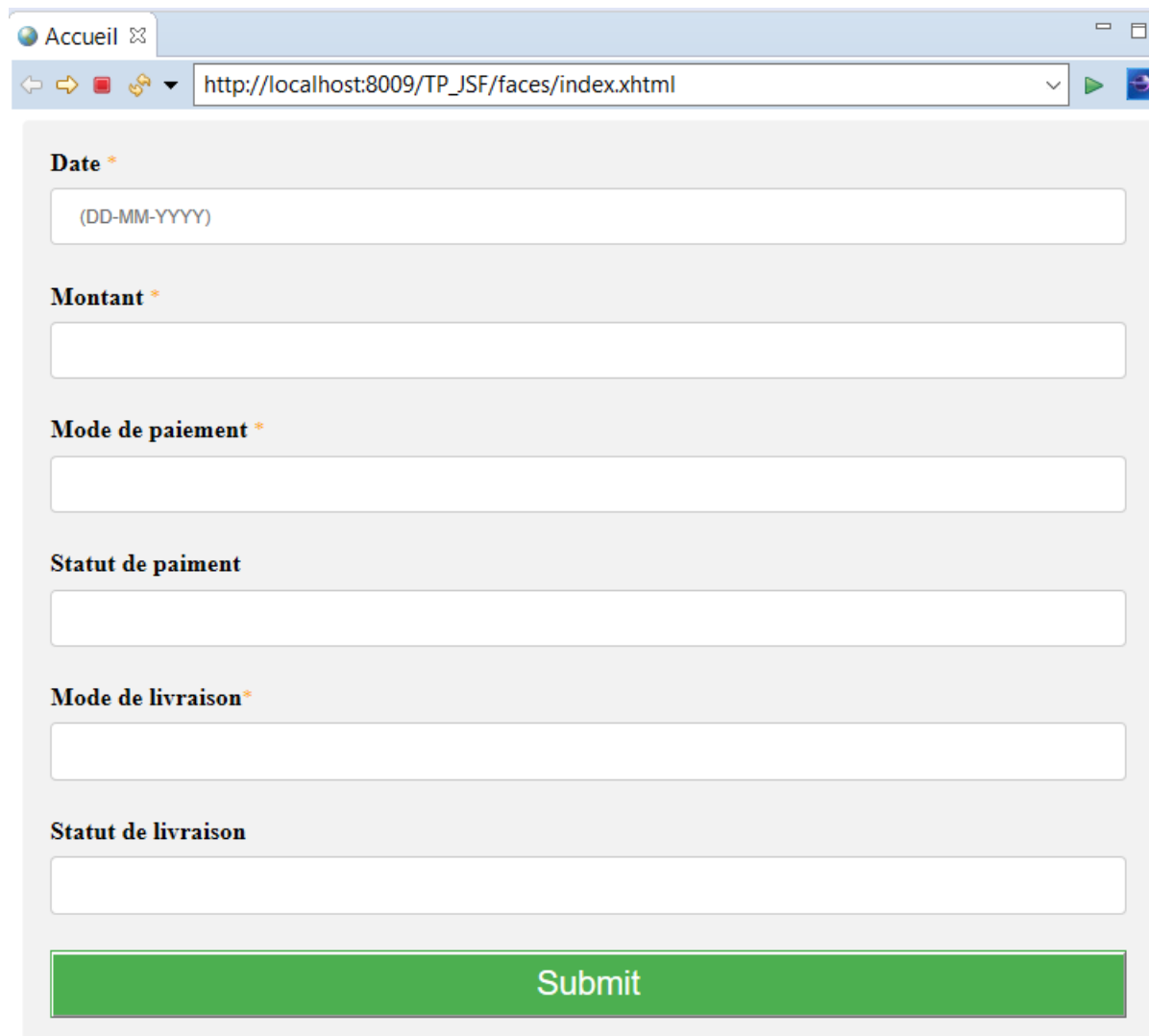
<h:outputLabel styleClass="outputs" value="Statut de paiement " />
<h:outputText styleClass="inputs final" value="#{commande.statutPaiement}" />

<h:outputLabel styleClass="outputs" value="Mode de livraison " />
<h:outputText styleClass="inputs final" value="#{commande.modeLivraison}" />

<h:outputLabel styleClass="outputs" value="Statut de livraison " />
<h:outputText styleClass="inputs final" value="#{commande.statutLivraison}" />
```

Résultats :

Page d'accueil:



The screenshot shows a web browser window with the title 'Accueil'. The address bar displays the URL 'http://localhost:8009/TP\_JSF/faces/index.xhtml'. The main content area contains a form with the following fields:

- Date \***: A text input field with a placeholder '(DD-MM-YYYY)'.
- Montant \***: A text input field.
- Mode de paiement \***: A text input field.
- Statut de paiement**: A text input field.
- Mode de livraison \***: A text input field.
- Statut de livraison**: A text input field.

At the bottom of the form is a large green button labeled 'Submit'.

Validation des champs obligatoires :

Accueil

http://localhost:8009/TP\_JSF/faces/index.xhtml

**Date \***

(DD-MM-YYYY)

*Date : erreur de validation. Vous devez indiquer une valeur.*

**Montant \***

*Montant : erreur de validation. Vous devez indiquer une valeur.*

**Mode de paiement \***

*Mode Paiement : erreur de validation. Vous devez indiquer une valeur.*

**Statut de paiement**

**Mode de livraison \***

*Mode Livraison : erreur de validation. Vous devez indiquer une valeur.*

**Statut de livraison**

Submit

Validation de la date :

**Date \***

12-13-2020

*Date : «12-13-2020» n'a pas pu être interprété en tant que date. Exemple : 05-02-2021*

Validation du montant :

**Montant \***

20002

You must enter a value between 0 and 20000

Validation du mode de paiement :

**Mode de paiement \***

Espèce de type feuille de 200 dirhames marocain

*Mode Paiement : erreur de validation. La longueur est supérieure à la valeur maximale autorisée, "20".*

Validation du mode de livraison :

**Mode de livraison \***

A domicile emporté par la société monTrans depuis l'entrepot de Ain Sebaa

*Mode Livraison : erreur de validation. La longueur est supérieure à la valeur maximale autorisée, "30".*

Affichage après validation et conversion :

A screenshot of a web browser window. The address bar shows the URL `http://localhost:8009/TP_JSF/faces/index.xhtml`. The page content is a form with several fields, each with a label and a value:

|                            |                         |
|----------------------------|-------------------------|
| <b>Date</b>                | vendredi 5 février 2021 |
| <b>Montant</b>             | 20 000,00 DH            |
| <b>Mode de paiement</b>    | Espèce                  |
| <b>Statut de paiement</b>  | Fait                    |
| <b>Mode de livraison</b>   | A domicile              |
| <b>Statut de livraison</b> | En cours                |