|  |
| --- |
| Formation JSF par la pratique  Partie 3 : Utilisation des templates (Les Facelets) et utilisation de Primefaces |
| Développement d'une application web avec JSF, JPA et WildFly |

Sommaire

[1. Objet 2](#_Toc2804506)

[2. Pré requis 3](#_Toc2804507)

[3. Partie 1 : Préparation de l'environnement de développement 4](#_Toc2804508)

[4. Partie 2 : Développement de l'application web 4](#_Toc2804509)

[**Exercices :** 63](#_Toc2804510)

# Objet

Le présent TP vous montre comment utiliser les Facelets et comment utiliser Primefaces. Le site de Primefaces est : ***https://www.primefaces.org***. La démonstration en ligne de Primefaces est accessible moyennant le lien suivant : ***https://www.primefaces.org/showcase***.

Dans cet exemple,nous allons utiliser les composants suivants de Primefaces :

* **<p:layout> :** pour la disposition de la page.
* **<p:layoutUnit>:** pour la réalisation d'une partie de la page. Par exemple : la page commonHeader.xhtml constituera l'entête de notre template.
* **<p:growl> :** pour l'affichage des messages.
* **<ui:include> :**Tag standard de JSF, permet d'intégrer une page dans une autre page.
* **<p:calendar> :** Pour afficher le calendrier.
* **<p:panelMenu> :** Pour créer la disposition de menu.
* **<p:submenu> :** Pour définir un menu principal.
* **<p:menuitem> :** Pour définir un sous menu du menu principal.
* **<p:separator /> :** Pour tracer une ligne séparatrice.
* **<ui:composition> :** Tag standard de JSF, permet de définir le contenu de la page à intégrer dans un template.
* **<p:dock> :** est un composant de navigation composé de menus.
* **<p:menuButton> :** regroupe plusieurs éléments dans une fenêtre contextuelle.
* **<p:accordionPanel> :** est un composant conteneur avec des panneaux empilés verticalement.
* **<p:galleria> :** est un composant pour afficher une galerie d'images.
* **<p:graphicImage> :** Pour afficher une image.
* **<ui:define> :** Tag standard de JSF pour redéfinir une partie d'un template.
* **<p:themeSwitcher> :** ThemeSwitcher applique de nouveaux thèmes à la volée sans actualisation complète de la page.
* **<p:fileUpload> :** Le programme de téléchargement simple utilise le téléchargement de fichier du navigateur natif.
* **<p:PanelGrid> :** PanelGrid est une extension du standard panelGrid avec intégration de thèmes, fonctionnalités de regroupement et d'autres features.
* **<p:dataTable> :** DataTable affiche les données sous forme de tableau et offre plusieurs fonctionnalités par exemple : la pagination, l'édition avec AJAX, le tri, etc.
* **<p:dataExporter> :** l'export des données dans un fichier PDF/Excel/CSV/XML.

# Pré requis

Les outils que nous allons utiliser pour réaliser ce TP sont présentés au niveau du tableau ci-après :

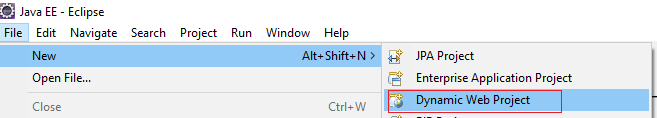
|  |  |
| --- | --- |
| Outil | Chemin du fichier  voir dossier : *formation\_Framework* |
| Eclipse Luna | formation\_Framework\tools\eclipse-luna.rar |
| WildFly 8.2.0 | formation\_Framework\tools\wildfly-8.2.0.Final.zip |
| Le Driver MySQL 5.0 | formation\_Framework\tools\lib\_divers\mysql-connector-java-5.1.17-bin.jar |
| Le serveur MySQL 5.0 | formation\_Framework\tools\MySQL 5.0\Setup.exe |
| Le client MySQL (l'interface graphique) | formation\_Framework\tools\MySQL 5.0\mysql-gui-tools-5.0-r17-win32.msi |
| JDK 1.7 | formation\_Framework\tools\jdk-7u75-windows-x64.exe |
| L'implémentation Hibernate de JPA. | formation\_Framework\tools\lib-hibernate |
| PrimeFaces version 7.0.RC3 | formation\_Framework\tools\Lib\_TP3\primefaces-7.0.RC3.jar |
| Les thèmes | formation\_Framework\tools\Lib\_TP3\\*.jar |
| Apache commons-io : Pour la copie d'une image d'un chemin vers un autre. | formation\_Framework\tools\Lib\_TP3\commons-io-2.4.jar |
| Apache POI pour l'export en format PDF. | formation\_Framework\tools\Lib\_TP3\poi-\*.jar |
| xmlbeans pour l'export en format XML. | formation\_Framework\tools\Lib\_TP3\xmlbeans-2.6.0.jar |
| Itext pour l'export en formet excel. | formation\_Framework\tools\Lib\_TP3\itext-4.2.1.jar |
| Les photos | formation\_Framework\TPs\images |

# Partie 1 : Préparation de l'environnement de développement

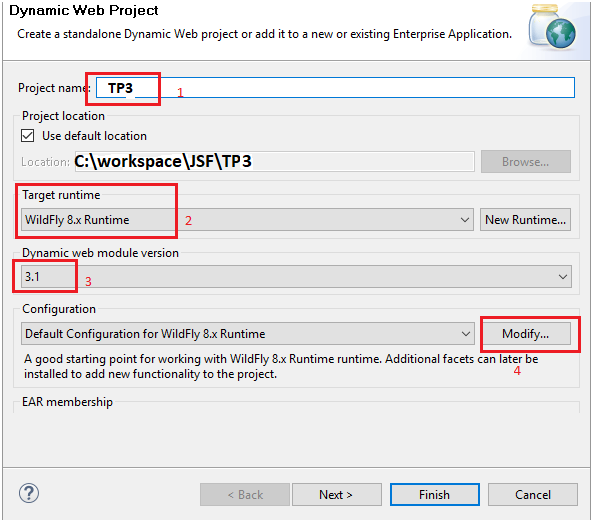
Idem que le TP n°2.

# Partie 2 : Développement de l'application web

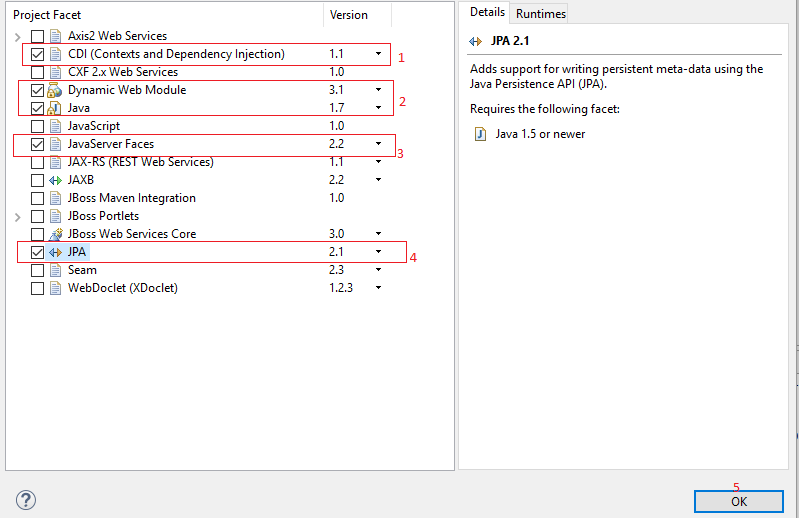
Au niveau d'Eclipse, créer un projet web comme illustre la fenêtre suivante :



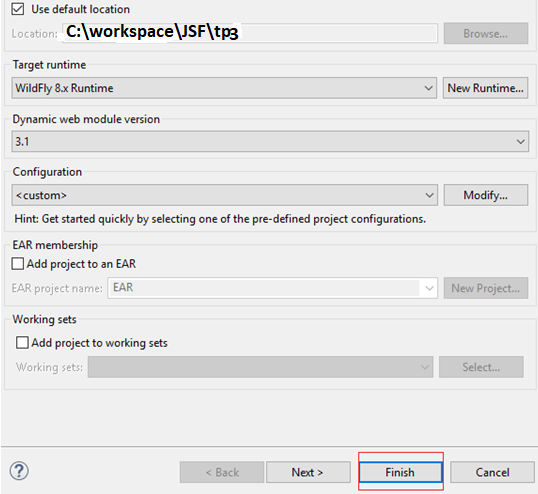
Cliquer sur "Dynamic Web Project" :



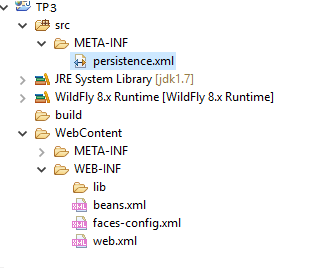
1. Saisir le nom de votre projet (ici : TP3);
2. Préciser la "Target runtime" Wildfly 8.x Runtime;
3. Préciser la version de la Servlet (ici : 3.1) ;
4. Cliquer ensuite sur Modifiy.... L'écran suivant sera affiché :



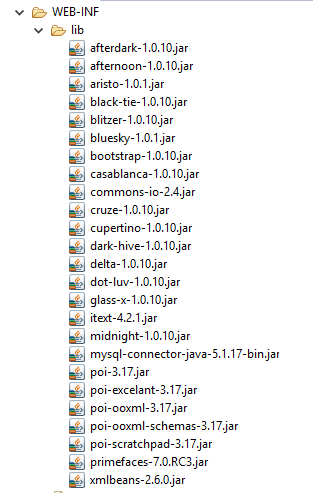
1. Cocher CDI (Context and Dependency Injection). L'api CDI implémente le Design Pattern IOC, nous en parlerons par la suite.
2. Vérifier que la version de "Dynamic Web Module" est bien 3.1 et la version de java est bien 1.7 ;
3. Cocher JavaServerFaces et préciser la version 2.2 ;
4. Cocher JPA et préciser la version 2.1 ;
5. Enfin, cliquer sur le bouton OK.



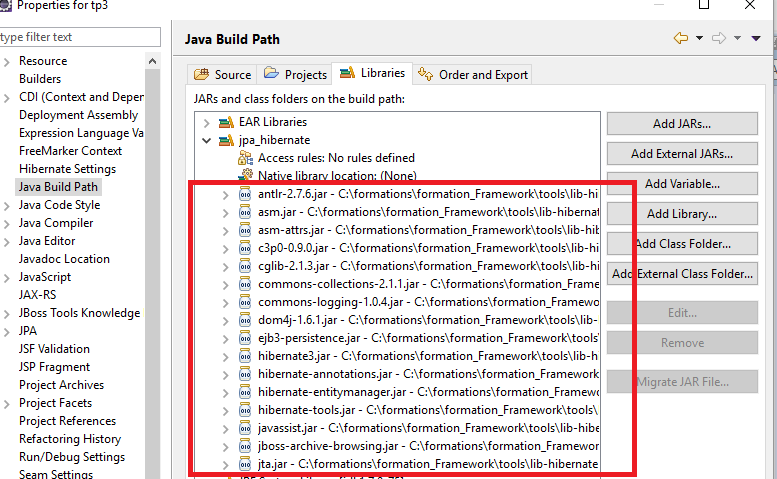
- Cliquer sur Finish. L'arborescence du projet ci-dessous est créée :



Copier les lib du dossier Lib\_TP3 dans WEB-INF/lib :



\* Ajouter les libs d'Hibernate au niveau du CLASSPATH comme le montre l'écran suivant :



1. **Le modèle :**

|  |
| --- |
| **La classe BaseEntity :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** javax.persistence.GeneratedValue;  **import** javax.persistence.Id;  **import** javax.persistence.MappedSuperclass;  @MappedSuperclass  **public** **abstract** **class** BaseEntity **implements** Serializable{  **private** **static** **final** **long** ***serialVersionUID*** = -663957177982798319L;  @GeneratedValue  @Id  **protected** Integer id;  **public** Integer getId() {  **return** id;  }  **public** **void** setId(Integer id) {  **this**.id = id;  }  } |

|  |
| --- |
| **La classe Article :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** javax.persistence.Basic;  **import** javax.persistence.CascadeType;  **import** javax.persistence.Column;  **import** javax.persistence.Entity;  **import** javax.persistence.FetchType;  **import** javax.persistence.JoinColumn;  **import** javax.persistence.ManyToOne;  **import** javax.persistence.Table;  @Entity  @Table(name = "Article")  **public** **class** Article **extends** BaseEntity **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 6161046340508532663L;  @Column(name = "libelle")  @Basic(fetch = FetchType.***LAZY***)  **private** String designation;  **private** Double prix;  **private** Integer quantiteEnStock;  @ManyToOne(cascade = CascadeType.***ALL***)  @JoinColumn(name = "CAT\_ID", nullable = **false**)  **private** Categorie categorie = **new** Categorie();  **private** String photo;  @Override  **public** String toString() {  **return** "Article [id=" + id + ", designation=" + designation + ", prix="  + prix + ", quantiteEnStock=" + quantiteEnStock + "]";  }  **public** String getPhoto() {  **return** photo;  }  **public** **void** setPhoto(String photo) {  **this**.photo = photo;  }  **public** Categorie getCategorie() {  **return** categorie;  }  **public** **void** setCategorie(Categorie categorie) {  **this**.categorie = categorie;  }  **public** Article() {  }  **public** Article(String designation, Double prix, Integer quantiteEnStock) {  **super**();  **this**.designation = designation;  **this**.prix = prix;  **this**.quantiteEnStock = quantiteEnStock;  }    **public** Article(String designation, Double prix, Integer quantiteEnStock,String photo) {  **this**(designation, prix, quantiteEnStock);  **this**.photo = photo;  }  **public** Article(String designation, Double prix, Integer quantiteEnStock,  Categorie categorie, String photo) {  **super**();  **this**.designation = designation;  **this**.prix = prix;  **this**.quantiteEnStock = quantiteEnStock;  **this**.categorie = categorie;  **this**.photo = photo;  }  **public** Article(String designation, Double prix, Integer quantiteEnStock,  Categorie categorie) {  **this**(designation, prix, quantiteEnStock);  **this**.categorie = categorie;  }  **public** Article(Integer id, String designation, Double prix,  Integer quantiteEnStock, Categorie categorie) {  **this**(designation, prix, quantiteEnStock, categorie);  **this**.id = id;  }  **public** Article(Integer id) {  **this**.id = id;  }    **public** String getDesignation() {  **return** designation;  }  **public** **void** setDesignation(String designation) {  **this**.designation = designation;  }  **public** Double getPrix() {  **return** prix;  }  **public** **void** setPrix(Double prix) {  **this**.prix = prix;  }  **public** Integer getQuantiteEnStock() {  **return** quantiteEnStock;  }  **public** **void** setQuantiteEnStock(Integer quantiteEnStock) {  **this**.quantiteEnStock = quantiteEnStock;  }  } |

|  |
| --- |
| **La classe Categorie :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** java.util.ArrayList;  **import** java.util.List;  **import** javax.persistence.CascadeType;  **import** javax.persistence.Entity;  **import** javax.persistence.FetchType;  **import** javax.persistence.OneToMany;  @Entity  **public** **class** Categorie **extends** BaseEntity **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = -5342496290945790545L;  **private** String code;  **private** String libelle;    @OneToMany(mappedBy="categorie",fetch=FetchType.***LAZY***,cascade=CascadeType.***ALL***)  **private** List<Article> articles=**new** ArrayList<>();  **public** Categorie() {  **super**();  }  **public** Categorie(**int** id) {  **this**.id = id;  }    **public** Categorie(String code, String libelle) {  **super**();  **this**.code = code;  **this**.libelle = libelle;  }  **public** String getCode() {  **return** code;  }  **public** **void** setCode(String code) {  **this**.code = code;  }  **public** String getLibelle() {  **return** libelle;  }  **public** **void** setLibelle(String libelle) {  **this**.libelle = libelle;  }  **public** **void** addArticle(Article article) {  articles.add(article);  article.setCategorie(**this**);  }  } |

|  |
| --- |
| **La classe Theme :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** javax.persistence.Entity;  @Entity  **public** **class** Theme **extends** BaseEntity **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = -8553346591231350594L;  **private** String code;  **private** String libelle;  **private** String themeByDefault;    **public** String getThemeByDefault() {  **return** themeByDefault;  }  **public** **void** setThemeByDefault(String themeByDefault) {  **this**.themeByDefault = themeByDefault;  }  **public** Theme() {  }    **public** Theme(Integer id) {  **this**.id=id;  }    **public** Theme(String code, String libelle) {  **super**();  **this**.code = code;  **this**.libelle = libelle;  }    **public** Theme(String code, String libelle,String themeByDefault) {  **super**();  **this**.code = code;  **this**.libelle = libelle;  **this**.themeByDefault=themeByDefault;  }    **public** Theme(Integer id,String code, String libelle) {  **this**.id=id;  **this**.code = code;  **this**.libelle = libelle;  }  **public** String getCode() {  **return** code;  }  **public** **void** setCode(String code) {  **this**.code = code;  }  **public** String getLibelle() {  **return** libelle;  }  **public** **void** setLibelle(String libelle) {  **this**.libelle = libelle;  }  } |

|  |
| --- |
| **La classe User :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** java.util.ArrayList;  **import** java.util.List;  **import** javax.persistence.CascadeType;  **import** javax.persistence.Entity;  **import** javax.persistence.OneToMany;  @Entity  **public** **class** User **extends** BaseEntity **implements** Serializable{  **private** **static** **final** **long** ***serialVersionUID*** = 1091781481106770718L;  **private** String userName;  **private** String password;  @OneToMany(mappedBy="user",cascade = CascadeType.***ALL***)  **private** List<UserRole> roles=**new** ArrayList<>();    **public** User() {  **super**();  // **TODO** Auto-generated constructor stub  }    **public** User(Integer id) {  **this**.id = id;  }  **public** User(Integer id, String userName, String password, List<UserRole> roles) {  **super**();  **this**.id = id;  **this**.userName = userName;  **this**.password = password;  **this**.roles = roles;  }    **public** User(Integer id, String userName, String password) {  **super**();  **this**.id = id;  **this**.userName = userName;  **this**.password = password;  }    **public** User(String userName, String password, List<UserRole> roles) {  **super**();  **this**.userName = userName;  **this**.password = password;  **this**.roles = roles;  }  **public** User(String userName, String password) {  **super**();  **this**.userName = userName;  **this**.password = password;  }    **public** String getUserName() {  **return** userName;  }  **public** **void** setUserName(String userName) {  **this**.userName = userName;  }  **public** String getPassword() {  **return** password;  }  **public** **void** setPassword(String password) {  **this**.password = password;  }  **public** List<UserRole> getRoles() {  **return** roles;  }  **public** **void** setRoles(List<UserRole> roles) {  **this**.roles = roles;  }  **public** **void** addUserRole(UserRole userRole) {  roles.add(userRole);  userRole.setUser(**this**);  }  } |

|  |
| --- |
| **La classe UserRole :** |
| **package** ma.gov.formation.cigma.service.modele;  **import** java.io.Serializable;  **import** javax.persistence.Entity;  **import** javax.persistence.JoinColumn;  **import** javax.persistence.ManyToOne;  @Entity  **public** **class** UserRole **extends** BaseEntity **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = -4769371604279552851L;  **private** String role;  **private** String libelle;  @ManyToOne  @JoinColumn(name="USER\_ID",nullable=**false**)  **private** User user;    **public** UserRole(Integer id) {  **this**.id = id;  }    **public** UserRole(String role, String libelle, User user) {  **super**();  **this**.role = role;  **this**.libelle = libelle;  **this**.user = user;  }  **public** UserRole(Integer id, String role, String libelle, User user) {  **super**();  **this**.id = id;  **this**.role = role;  **this**.libelle = libelle;  **this**.user = user;  }  **public** User getUser() {  **return** user;  }  **public** **void** setUser(User user) {  **this**.user = user;  }  **public** UserRole() {  **super**();  }    **public** UserRole(Integer id, String role, String libelle) {  **super**();  **this**.id = id;  **this**.role = role;  **this**.libelle = libelle;  }  **public** UserRole(String role, String libelle) {  **super**();  **this**.role = role;  **this**.libelle = libelle;  }  **public** String getRole() {  **return** role;  }  **public** **void** setRole(String role) {  **this**.role = role;  }  **public** String getLibelle() {  **return** libelle;  }  **public** **void** setLibelle(String libelle) {  **this**.libelle = libelle;  }  } |

1. **Les "Value Object"**

|  |
| --- |
| **La classe BaseVo :** |
| **package** ma.gov.formation.cigma.domaine;  **import** java.io.Serializable;  **public** **class** BaseVo **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 8500324480292121343L;  **protected** Integer id;  **protected** String code;  **protected** String libelle;  **public** BaseVo(Integer id) {  **super**();  **this**.id = id;  }  **public** BaseVo() {  **super**();  }  **public** BaseVo(String code, String libelle) {  **super**();  **this**.code = code;  **this**.libelle = libelle;  }  **public** BaseVo(Integer id, String code, String libelle) {  **super**();  **this**.id = id;  **this**.code = code;  **this**.libelle = libelle;  }  **public** Integer getId() {  **return** id;  }  **public** **void** setId(Integer id) {  **this**.id = id;  }  **public** String getCode() {  **return** code;  }  **public** **void** setCode(String code) {  **this**.code = code;  }  **public** String getLibelle() {  **return** libelle;  }  **public** **void** setLibelle(String libelle) {  **this**.libelle = libelle;  }  } |

|  |
| --- |
| **La classe ArticleVo :** |
| **package** ma.gov.formation.cigma.domaine.article;  **import** java.io.Serializable;  **import** ma.gov.formation.cigma.domaine.BaseVo;  **import** ma.gov.formation.cigma.domaine.categorie.CategorieVo;  **public** **class** ArticleVo **extends** BaseVo **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 6161046340508532663L;  **private** String designation;  **private** Double prix;  **private** Integer quantiteEnStock;  **private** CategorieVo categorie=**new** CategorieVo();  **private** String photo;  **public** ArticleVo(Integer id) {  **super**();  **this**.id = id;  }  **public** String getPhoto() {  **return** photo;  }  **public** **void** setPhoto(String photo) {  **this**.photo = photo;  }  **public** CategorieVo getCategorie() {  **return** categorie;  }  **public** **void** setCategorie(CategorieVo categorie) {  **this**.categorie = categorie;  }  **private** **boolean** editable;  **public** **boolean** isEditable() {  **return** editable;  }  **public** **void** setEditable(**boolean** editable) {  **this**.editable = editable;  }  **public** ArticleVo() {  }  **public** ArticleVo(Integer id, String designation, Double prix,  Integer quantiteEnStock) {  **this**.id = id;  **this**.designation = designation;  **this**.prix = prix;  **this**.quantiteEnStock = quantiteEnStock;  categorie=**new** CategorieVo();  photo=**null**;  }    **public** **void** clear() {  id = **null**;  designation = **null**;  prix = **null**;  quantiteEnStock = **null**;  }  **public** String getDesignation() {  **return** designation;  }  **public** **void** setDesignation(String designation) {  **this**.designation = designation;  }  **public** Double getPrix() {  **return** prix;  }  **public** **void** setPrix(Double prix) {  **this**.prix = prix;  }  **public** Integer getQuantiteEnStock() {  **return** quantiteEnStock;  }  **public** **void** setQuantiteEnStock(Integer quantiteEnStock) {  **this**.quantiteEnStock = quantiteEnStock;  }  } |

|  |
| --- |
| **La classe ArticleConverter :** |
| **package** ma.gov.formation.cigma.domaine.article;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.domaine.categorie.CategoreConverter;  **import** ma.gov.formation.cigma.service.modele.Article;  **public** **class** ArticleConverter {  **public** **static** Article createModele(ArticleVo vo) {  Article modele = **new** Article();  modele.setId(vo.getId());  modele.setDesignation(vo.getDesignation());  modele.setPrix(vo.getPrix());  modele.setQuantiteEnStock(vo.getQuantiteEnStock());  modele.getCategorie().setCode(vo.getCategorie().getCode());  modele.setPhoto(vo.getPhoto());  **return** modele;  }  **public** **static** ArticleVo createArticleVo(Article modele) {  ArticleVo vo = **new** ArticleVo();  vo.setId(modele.getId());  vo.setDesignation(modele.getDesignation());  vo.setPrix(modele.getPrix());  vo.setQuantiteEnStock(modele.getQuantiteEnStock());  vo.setCategorie(CategoreConverter.*createCategorieVo*(modele.getCategorie()));  vo.setPhoto(modele.getPhoto());  **return** vo;  }  **public** **static** List<ArticleVo> getArticleVoList(List<Article> modeles) {  List<ArticleVo> vos = **new** ArrayList<>();  **if** (modeles == **null** || modeles.isEmpty())  **return** **null**;  **for** (Article modele : modeles) {  vos.add(*createArticleVo*(modele));  }  **return** vos;  }  } |

|  |
| --- |
| **La classe CategorieVo :** |
| **package** ma.gov.formation.cigma.domaine.categorie;  **import** java.io.Serializable;  **import** ma.gov.formation.cigma.domaine.BaseVo;  **public** **class** CategorieVo **extends** BaseVo **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 6901655927457141058L;  **public** CategorieVo() {  **super**();  }  **public** CategorieVo(Integer id) {  **super**();  **this**.id = id;  }    **public** CategorieVo(String code,String libelle) {  **super**(code,libelle);  }  } |

|  |
| --- |
| **La classe CategoreConverter :** |
| **package** ma.gov.formation.cigma.domaine.categorie;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **class** CategoreConverter {  **public** **static** Categorie createModele(CategorieVo vo) {  Categorie modele = **new** Categorie();  modele.setId(vo.getId());  modele.setCode(vo.getCode());  modele.setLibelle(vo.getLibelle());  **return** modele;  }  **public** **static** CategorieVo createCategorieVo(Categorie modele) {  CategorieVo vo = **new** CategorieVo();  vo.setId(modele.getId());  vo.setCode(modele.getCode());  vo.setLibelle(modele.getLibelle());  **return** vo;  }    **public** **static** List<CategorieVo> getCategorieVoList(List<Categorie> modeles) {  List<CategorieVo> vos=**new** ArrayList<>();  **if** (modeles == **null** || modeles.isEmpty())  **return** **null**;  **for** (Categorie modele : modeles) {  vos.add(*createCategorieVo*(modele));  }  **return** vos;  }  } |

|  |
| --- |
| **La classe ThemeVo :** |
| **package** ma.gov.formation.cigma.domaine.theme;  **import** ma.gov.formation.cigma.domaine.BaseVo;  **public** **class** ThemeVo **extends** BaseVo{  **private** **static** **final** **long** ***serialVersionUID*** = -8884029347619819836L;  **private** String themeByDefault;  **public** String getThemeByDefault() {  **return** themeByDefault;  }  **public** **void** setThemeByDefault(String themeByDefault) {  **this**.themeByDefault = themeByDefault;  }  **public** ThemeVo() {  }  **public** ThemeVo(**int** id, String code, String libelle) {  **super**(id,code,libelle);  }    **public** ThemeVo(**int** id, String code, String libelle,String themeByDefault) {  **super**(id,code,libelle);  **this**.themeByDefault=themeByDefault;  }  } |

|  |
| --- |
| **La classe ThemeConverter :** |
| **package** ma.gov.formation.cigma.domaine.theme;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.service.modele.Theme;  **public** **class** ThemeConverter {  **public** **static** Theme createModele(ThemeVo vo) {  Theme modele = **new** Theme();  modele.setCode(vo.getCode());  modele.setLibelle(vo.getLibelle());  modele.setId(vo.getId());  modele.setThemeByDefault(vo.getThemeByDefault());  **return** modele;  }  **public** **static** ThemeVo createThemeVo(Theme modele) {  ThemeVo vo = **new** ThemeVo();  vo.setId(modele.getId());  vo.setCode(modele.getCode());  vo.setLibelle(modele.getLibelle());  vo.setThemeByDefault(modele.getThemeByDefault());  **return** vo;  }  **public** **static** List<ThemeVo> getThemeVoList(List<Theme> modeles) {  List<ThemeVo> vos = **new** ArrayList<>();  **if** (modeles == **null** || modeles.isEmpty())  **return** **null**;  **for** (Theme modele : modeles) {  vos.add(*createThemeVo*(modele));  }  **return** vos;  }  } |

|  |
| --- |
| **La classe UserVo :** |
| **package** ma.gov.formation.cigma.domaine.user;  **import** java.io.Serializable;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.domaine.BaseVo;  **import** ma.gov.formation.cigma.presentation.Constants;  **public** **class** UserVo **extends** BaseVo **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = -5500475381869205248L;  **private** String userName;  **private** String password;  **private** List<UserRoleVo> roles = **new** ArrayList<>();    **public** UserVo(Integer id) {  **super**();  **this**.id = id;  }  **public** **void** setAdmin(**boolean** admin) {  **this**.admin = admin;  }  **public** **void** setClient(**boolean** client) {  **this**.client = client;  }  **private** **boolean** admin;  **private** **boolean** client;  **public** **boolean** isAdmin() {  **return** haseRole(Constants.***ADMIN\_ROLE***);  }  **public** **boolean** isClient() {  **return** haseRole(Constants.***CLIENT\_ROLE***);  }  **public** List<UserRoleVo> getRoles() {  **return** roles;  }  **public** **void** setRoles(List<UserRoleVo> roles) {  **this**.roles = roles;  }  **public** UserVo() {  **super**();  // **TODO** Auto-generated constructor stub  }  /\*\*  \* **@param** userName  \* **@param** password  \*/  **public** UserVo(String userName, String password) {  **super**();  **this**.userName = userName;  **this**.password = password;  }  /\*\*  \* **@param** id  \* **@param** userName  \* **@param** password  \*/  **public** UserVo(Integer id, String userName, String password) {  **super**();  **this**.id = id;  **this**.userName = userName;  **this**.password = password;  }  **public** String getUserName() {  **return** userName;  }  **public** **void** setUserName(String userName) {  **this**.userName = userName;  }  **public** String getPassword() {  **return** password;  }  **public** **void** setPassword(String password) {  **this**.password = password;  }  **public** **boolean** haseRole(String role) {  **boolean** result = **false**;  **if** (roles.isEmpty())  **return** **false**;  **for** (UserRoleVo vo : roles) {  **if** (role.equals(vo.getCode()))  **return** **true**;  }  **return** result;  }  } |

|  |
| --- |
| **La classe UserRoleVo :** |
| **package** ma.gov.formation.cigma.domaine.user;  **import** java.io.Serializable;  **import** ma.gov.formation.cigma.domaine.BaseVo;  **public** **class** UserRoleVo **extends** BaseVo **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 4928414063636905875L;  **public** UserRoleVo() {  **super**();  }  **public** UserRoleVo(String code, String libelle) {  **super**();  **this**.code = code;  **this**.libelle = libelle;  }  **public** UserRoleVo(Integer id, String code, String libelle) {  **this**(code,libelle);  **this**.id = id;  }  } |

|  |
| --- |
| **La classe UserConverter :** |
| **package** ma.gov.formation.cigma.domaine.user.converter;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.domaine.user.UserVo;  **import** ma.gov.formation.cigma.service.modele.User;  **public** **class** UserConverter {  **public** **static** User createModele(UserVo vo) {  User modele = **new** User();  modele.setId(vo.getId());  modele.setUserName(vo.getUserName());  modele.setPassword(vo.getPassword());  **return** modele;  }  **public** **static** UserVo createUserVo(User modele) {  UserVo vo = **null**;  **if** (modele == **null**)  **return** vo;  vo = **new** UserVo();  vo.setId(modele.getId());  vo.setUserName(modele.getUserName());  vo.setPassword(modele.getPassword());  **if** (modele.getRoles() == **null** || modele.getRoles().isEmpty())  **return** vo;  vo.setRoles(UserRoleConverter.*getUserRoleVoList*(modele.getRoles()));  **return** vo;  }  **public** **static** List<UserVo> getUserVoList(List<User> modeles) {  List<UserVo> vos = **new** ArrayList<>();  **if** (modeles == **null** || modeles.isEmpty())  **return** **null**;  **for** (User modele : modeles) {  vos.add(*createUserVo*(modele));  }  **return** vos;  }  } |

|  |
| --- |
| **La classe UserRoleConverter :** |
| **package** ma.gov.formation.cigma.domaine.user.converter;  **import** java.util.ArrayList;  **import** java.util.List;  **import** ma.gov.formation.cigma.domaine.user.UserRoleVo;  **import** ma.gov.formation.cigma.service.modele.UserRole;  **public** **class** UserRoleConverter {  **public** **static** UserRole createModele(UserRoleVo vo) {  UserRole modele = **new** UserRole();  modele.setId(vo.getId());  modele.setLibelle(vo.getLibelle());  modele.setRole(vo.getCode());  **return** modele;  }  **public** **static** UserRoleVo createUserRoleVo(UserRole modele) {  UserRoleVo vo = **new** UserRoleVo();  vo.setId(modele.getId());  vo.setLibelle(modele.getLibelle());  vo.setCode(modele.getRole());  **return** vo;  }  **public** **static** List<UserRoleVo> getUserRoleVoList(List<UserRole> modeles) {  List<UserRoleVo> vos = **new** ArrayList<>();  **if** (modeles == **null** || modeles.isEmpty())  **return** vos;  **for** (UserRole modele : modeles) {  vos.add(*createUserRoleVo*(modele));  }  **return** vos;  }  } |

1. **La couche DAO**

|  |
| --- |
| **L'interface IBaseDao :** |
| **package** ma.gov.formation.cigma.dao;  **import** java.util.List;  **public** **interface** IBaseDao {  **void** beginTransaction() **throws** Exception;  **void** validTransaction() **throws** Exception;  <T> T save(T t) **throws** Exception;  <T> T getById (T t) **throws** Exception;  <T> List<T> getAll(T t) **throws** Exception;  <T> **void** remove(T t) **throws** Exception;  } |

|  |
| --- |
| **L'interface IArticleDao :** |
| **package** ma.gov.formation.cigma.dao;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **interface** IArticleDao **extends** IBaseDao{  Categorie getCategoreByCode(String code) **throws** Exception;  } |

|  |
| --- |
| **L'interface IFileDao :** |
| **package** ma.gov.formation.cigma.dao;  **import** java.io.File;  **import** java.io.IOException;  **import** java.io.InputStream;  **public** **interface** IFileDao {  **void** copy(InputStream input, File fileDest) **throws** IOException;  } |

|  |
| --- |
| **L'interface IThemeDao :** |
| **package** ma.gov.formation.cigma.dao;  **import** ma.gov.formation.cigma.service.modele.Theme;  **public** **interface** IThemeDao **extends** IBaseDao {  Theme getDefaultTheme() **throws** Exception;  Theme getByCode(String code) **throws** Exception;  } |

|  |
| --- |
| **L'interface IUserDao :** |
| **package** ma.gov.formation.cigma.dao;  **import** ma.gov.formation.cigma.service.modele.User;  **public** **interface** IUserDao **extends** IBaseDao {  User getByLogin(String login) **throws** Exception;  } |

|  |
| --- |
| **La classe SessionBuilder :** |
| **package** ma.gov.formation.cigma.dao.impl;  **import** javax.persistence.EntityManager;  **import** javax.persistence.EntityManagerFactory;  **import** javax.persistence.Persistence;  **public** **class** SessionBuilder {  **private** **static** EntityManagerFactory *sessionFactory* = **null**;  **private** **static** EntityManager *session*=**null**;  **private** **static** SessionBuilder *instance* = **null**;  **private** SessionBuilder() {  }  **public** **static** SessionBuilder getInstance() **throws** Exception {  **if** (*instance* == **null**) {  *instance* = **new** SessionBuilder();  *sessionFactory* = Persistence.*createEntityManagerFactory*("unite1");  *session*=*sessionFactory*.createEntityManager();  }  **return** *instance*;  }  **public** EntityManager getSession() {  **return** *session*;  }  } |

|  |
| --- |
| **La classe BaseDaoImpl :** |
| **package** ma.gov.formation.cigma.dao.impl;  **import** java.util.List;  **import** javax.persistence.EntityManager;  **import** javax.persistence.EntityTransaction;  **import** ma.gov.formation.cigma.dao.IBaseDao;  **import** ma.gov.formation.cigma.service.modele.BaseEntity;  **public** **class** BaseDaoImpl **implements** IBaseDao {  **public** **static** EntityManager *session* = **null**;  **protected** EntityTransaction tx = **null**;  @Override  **public** **void** beginTransaction() **throws** Exception {  *session* = SessionBuilder.*getInstance*().getSession();  tx = *session*.getTransaction();  tx.begin();  }  @Override  **public** **void** validTransaction() **throws** Exception {  tx.commit();  }  @Override  **public** <T> T save(T t) **throws** Exception {  **return** *session*.merge(t);  }  @SuppressWarnings("unchecked")  @Override  **public** <T> T getById(T t) **throws** Exception {  String entityName = t.getClass().getName();  **int** id = ((BaseEntity) t).getId();  String req = "from " + entityName + " where id=" + id;  **return** (T) *session*.createQuery(req).getSingleResult();  }  @SuppressWarnings("unchecked")  @Override  **public** <T> List<T> getAll(T t) **throws** Exception {  String entityName = t.getClass().getName();  String req = "from " + entityName;  **return** *session*.createQuery(req).getResultList();  }  @Override  **public** <T> **void** remove(T t) **throws** Exception {  T tp = getById(t);  **if** (tp != **null**)  *session*.remove(tp);  }  } |

|  |
| --- |
| **La classe ArticleDaoImpl :** |
| **package** ma.gov.formation.cigma.dao.impl;  **import** java.util.List;  **import** javax.persistence.Query;  **import** ma.gov.formation.cigma.dao.IArticleDao;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **class** ArticleDaoImpl **extends** BaseDaoImpl **implements** IArticleDao {  @SuppressWarnings("unchecked")  @Override  **public** Categorie getCategoreByCode(String code) **throws** Exception{  List<Categorie> liste = **null**;    Query q=*session*.createQuery("select c from Categorie c where c.code=:code");  q.setParameter("code", code);  liste = (List<Categorie>) q.getResultList();  **return** liste==**null** || liste.isEmpty() ? **null**:liste.get(0);  }  } |

|  |
| --- |
| **La classe FileDaoImpl** : |
| **package** ma.gov.formation.cigma.dao.impl;  **import** java.io.File;  **import** java.io.FileOutputStream;  **import** java.io.IOException;  **import** java.io.InputStream;  **import** java.io.OutputStream;  **import** ma.gov.formation.cigma.dao.IFileDao;  **import** org.apache.commons.io.IOUtils;  **public** **class** FileDaoImpl **implements** IFileDao {  @Override  **public** **void** copy(InputStream input, File fileDest) **throws** IOException {  OutputStream outputStream= **new** FileOutputStream(fileDest);  IOUtils.*copy*(input, outputStream);  }  } |

|  |
| --- |
| **La classe ThemeDaoImpl :** |
| **package** ma.gov.formation.cigma.dao.impl;  **import** java.util.List;  **import** javax.persistence.Query;  **import** ma.gov.formation.cigma.dao.IThemeDao;  **import** ma.gov.formation.cigma.service.modele.Theme;  **public** **class** ThemeDaoImpl **extends** BaseDaoImpl **implements** IThemeDao {  @Override  @SuppressWarnings("unchecked")  **public** Theme getDefaultTheme() **throws** Exception {  Theme a = **null**;  Query q = *session*.createQuery("from Theme t where t.themeByDefault=:themeByDefault");  q.setParameter("themeByDefault", "YES");  List<Theme> result = q.getResultList();  a = (result != **null** && !result.isEmpty()) ? result.get(0) : **null**;  **return** a;  }  @Override  **public** Theme getByCode(String code) **throws** Exception {  Theme a = **null**;  Query q = *session*.createQuery("from Theme t where t.code=:code");  q.setParameter("code", code);  List<Theme> result = q.getResultList();  a = (result != **null** && !result.isEmpty()) ? result.get(0) : **null**;  **return** a;  }  } |

|  |
| --- |
| **La classe UserDaoImpl :** |
| **package** ma.gov.formation.cigma.dao.impl;  **import** java.util.List;  **import** javax.persistence.Query;  **import** ma.gov.formation.cigma.dao.IUserDao;  **import** ma.gov.formation.cigma.service.modele.User;  **public** **class** UserDaoImpl **extends** BaseDaoImpl **implements** IUserDao {  @Override  @SuppressWarnings("unchecked")  **public** User getByLogin(String userName) **throws** Exception {  User a = **null**;  Query q = *session*.createQuery("from User u where u.userName=:userName");  q.setParameter("userName", userName);  List<User> result = q.getResultList();  a = (result != **null** && !result.isEmpty()) ? result.get(0) : **null**;  **return** a;  }  } |

1. **La couche Service**

|  |
| --- |
| **L'interface IBaseService :** |
| **package** ma.gov.formation.cigma.service;  **import** java.util.List;  **public** **interface** IBaseService {  **void** beginTransaction() **throws** Exception;  **void** validTransaction() **throws** Exception;  <T> T getById (T t) **throws** Exception;  <T> List<T> getAll(T t) **throws** Exception;  <T> **void** remove(T t) **throws** Exception;  <T> **void** save(T t) **throws** Exception;  } |

|  |
| --- |
| **L'interface IArticleService :** |
| **package** ma.gov.formation.cigma.service;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **interface** IArticleService **extends** IBaseService {  Categorie getCategoreByCode(String code) **throws** Exception;  } |

|  |
| --- |
| **L'interface IFileService :** |
| **package** ma.gov.formation.cigma.service;  **import** java.io.IOException;  **import** java.io.InputStream;  **public** **interface** IFileService {  **void** copyPhotoArticle(InputStream inputstream, String fileName) **throws** IOException;  } |

|  |
| --- |
| **L'interface IThemeService :** |
| **package** ma.gov.formation.cigma.service;  **import** ma.gov.formation.cigma.domaine.theme.ThemeVo;  **public** **interface** IThemeService **extends** IBaseService {  **public** ThemeVo getDefaultTheme() **throws** Exception;  **public** ThemeVo updateThemeByDefault(String code) **throws** Exception;  } |

|  |
| --- |
| **L'interface IUserService :** |
| **package** ma.gov.formation.cigma.service;  **import** ma.gov.formation.cigma.domaine.user.UserVo;  **public** **interface** IUserService **extends** IBaseService{  **public** UserVo getByLogin(String login) **throws** Exception;  } |

|  |
| --- |
| **La classe BaseServiceImpl :** |
| **package** ma.gov.formation.cigma.service.impl;  **import** java.util.List;  **import** ma.gov.formation.cigma.dao.IBaseDao;  **import** ma.gov.formation.cigma.dao.impl.BaseDaoImpl;  **import** ma.gov.formation.cigma.domaine.article.ArticleConverter;  **import** ma.gov.formation.cigma.domaine.article.ArticleVo;  **import** ma.gov.formation.cigma.domaine.categorie.CategoreConverter;  **import** ma.gov.formation.cigma.domaine.categorie.CategorieVo;  **import** ma.gov.formation.cigma.domaine.theme.ThemeConverter;  **import** ma.gov.formation.cigma.domaine.theme.ThemeVo;  **import** ma.gov.formation.cigma.domaine.user.UserVo;  **import** ma.gov.formation.cigma.domaine.user.converter.UserConverter;  **import** ma.gov.formation.cigma.service.IBaseService;  **import** ma.gov.formation.cigma.service.modele.Article;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **import** ma.gov.formation.cigma.service.modele.Theme;  **import** ma.gov.formation.cigma.service.modele.User;  **public** **class** BaseServiceImpl **implements** IBaseService {  **protected** IBaseDao baseDao = **new** BaseDaoImpl();  @Override  **public** **void** beginTransaction() **throws** Exception {  baseDao.beginTransaction();  }  @Override  **public** **void** validTransaction() **throws** Exception {  baseDao.validTransaction();  }  @SuppressWarnings("unchecked")  @Override  **public** <T> T getById(T t) **throws** Exception {  T result = **null**;  beginTransaction();  **if** (t **instanceof** ArticleVo) {  Integer id = ((ArticleVo) t).getId();  result = (T) ArticleConverter.*createArticleVo*(baseDao  .getById(**new** Article(id)));  }  **if** (t **instanceof** UserVo) {  Integer id = ((UserVo) t).getId();  result = (T) UserConverter.*createUserVo*(baseDao  .getById(**new** User(id)));  }  **if** (t **instanceof** CategorieVo) {  Integer id = ((CategorieVo) t).getId();  result = (T) CategoreConverter.*createCategorieVo*(baseDao  .getById(**new** Categorie(id)));  }    **if** (t **instanceof** ThemeVo) {  Integer id = ((ThemeVo) t).getId();  result = (T) ThemeConverter.*createThemeVo*(baseDao  .getById(**new** Theme(id)));  }  validTransaction();  **return** result;  }  @SuppressWarnings("unchecked")  @Override  **public** <T> List<T> getAll(T t) **throws** Exception {  List<T> list = **null**;  beginTransaction();  **if** (t **instanceof** ArticleVo)  list = (List<T>) ArticleConverter.*getArticleVoList*(baseDao  .getAll(**new** Article()));  **if** (t **instanceof** UserVo)  list = (List<T>) UserConverter.*getUserVoList*(baseDao  .getAll(**new** User()));  **if** (t **instanceof** CategorieVo)  list = (List<T>) CategoreConverter.*getCategorieVoList*(baseDao  .getAll(**new** Categorie()));    **if** (t **instanceof** ThemeVo)  list = (List<T>) ThemeConverter.*getThemeVoList*(baseDao  .getAll(**new** Theme()));  validTransaction();  **return** list;  }  @Override  **public** <T> **void** remove(T t) **throws** Exception {  beginTransaction();  **if** (t **instanceof** ArticleVo)  baseDao.remove(ArticleConverter.*createModele*((ArticleVo) t));  **if** (t **instanceof** UserVo)  baseDao.remove(UserConverter.*createModele*((UserVo) t));  **if** (t **instanceof** CategorieVo)  baseDao.remove(CategoreConverter.*createModele*((CategorieVo) t));    **if** (t **instanceof** ThemeVo)  baseDao.remove(ThemeConverter.*createModele*((ThemeVo) t));    validTransaction();  }  @Override  **public** <T> **void** save(T t) **throws** Exception {  beginTransaction();  **if** (t **instanceof** ArticleVo)  baseDao.save(ArticleConverter.*createModele*((ArticleVo) t));  **if** (t **instanceof** UserVo)  baseDao.save(UserConverter.*createModele*((UserVo) t));  **if** (t **instanceof** CategorieVo)  baseDao.save(CategoreConverter.*createModele*((CategorieVo) t));    **if** (t **instanceof** ThemeVo)  baseDao.save(ThemeConverter.*createModele*((ThemeVo) t));  validTransaction();  }  } |

|  |
| --- |
| **La classe ArticleServiceImpl :** |
| **package** ma.gov.formation.cigma.service.impl;  **import** ma.gov.formation.cigma.dao.IArticleDao;  **import** ma.gov.formation.cigma.dao.impl.ArticleDaoImpl;  **import** ma.gov.formation.cigma.domaine.article.ArticleConverter;  **import** ma.gov.formation.cigma.domaine.article.ArticleVo;  **import** ma.gov.formation.cigma.service.IArticleService;  **import** ma.gov.formation.cigma.service.modele.Article;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **class** ArticleServiceImpl **extends** BaseServiceImpl **implements**  IArticleService {  **private** IArticleDao dao = **new** ArticleDaoImpl();  @Override  **public** <T> **void** save(T t) **throws** Exception {  ArticleVo vo = (ArticleVo) t;  beginTransaction();  String code = vo.getCategorie().getCode();  Categorie catP = getCategoreByCode(code);  Article article = ArticleConverter.*createModele*(vo);  article.setCategorie(catP);  article = dao.save(article);  validTransaction();  }  **public** Categorie getCategoreByCode(String code) **throws** Exception {  // **TODO** Auto-generated method stub  **return** dao.getCategoreByCode(code);  }  } |

|  |
| --- |
| **La classe FileServiceImpl :** |
| **package** ma.gov.formation.cigma.service.impl;  **import** java.io.File;  **import** java.io.IOException;  **import** java.io.InputStream;  **import** ma.gov.formation.cigma.dao.IFileDao;  **import** ma.gov.formation.cigma.dao.impl.FileDaoImpl;  **import** ma.gov.formation.cigma.presentation.Constants;  **import** ma.gov.formation.cigma.service.IFileService;  **public** **class** FileServiceImpl **implements** IFileService {  IFileDao fileDo=**new** FileDaoImpl();  @Override  **public** **void** copyPhotoArticle(InputStream inputstream, String fileName) **throws** IOException {  File dir=**new** File(Constants.***IMAGES\_ARTICLES\_PATH***,fileName);  fileDo.copy(inputstream, dir);  }  } |

|  |
| --- |
| **La classe ThemeServiceImpl :** |
| **package** ma.gov.formation.cigma.service.impl;  **import** ma.gov.formation.cigma.dao.IThemeDao;  **import** ma.gov.formation.cigma.dao.impl.ThemeDaoImpl;  **import** ma.gov.formation.cigma.domaine.theme.ThemeConverter;  **import** ma.gov.formation.cigma.domaine.theme.ThemeVo;  **import** ma.gov.formation.cigma.service.IThemeService;  **import** ma.gov.formation.cigma.service.modele.Theme;  **public** **class** ThemeServiceImpl **extends** BaseServiceImpl **implements** IThemeService {  **private** IThemeDao dao = **new** ThemeDaoImpl();  @Override  **public** ThemeVo getDefaultTheme() **throws** Exception {  ThemeVo vo = **null**;  beginTransaction();  vo = ThemeConverter.*createThemeVo*(dao.getDefaultTheme());  validTransaction();  **return** vo;  }  @Override  **public** ThemeVo updateThemeByDefault(String code) **throws** Exception {  ThemeVo vo=**null**;  beginTransaction();  Theme themeByDefaultInit=dao.getDefaultTheme();  themeByDefaultInit.setThemeByDefault(**null**);  Theme newThemeByDefault=dao.getByCode(code);  newThemeByDefault.setThemeByDefault("YES");  vo = ThemeConverter.*createThemeVo*(newThemeByDefault);  validTransaction();  **return** vo;  }  } |

|  |
| --- |
| **La classe UserServiceImpl :** |
| **package** ma.gov.formation.cigma.service.impl;  **import** ma.gov.formation.cigma.dao.IUserDao;  **import** ma.gov.formation.cigma.dao.impl.UserDaoImpl;  **import** ma.gov.formation.cigma.domaine.user.UserVo;  **import** ma.gov.formation.cigma.domaine.user.converter.UserConverter;  **import** ma.gov.formation.cigma.service.IUserService;  **public** **class** UserServiceImpl **extends** BaseServiceImpl **implements** IUserService {  **private** IUserDao dao = **new** UserDaoImpl();  @Override  **public** UserVo getByLogin(String login) **throws** Exception {  UserVo vo = **null**;  beginTransaction();  vo = UserConverter.*createUserVo*(dao.getByLogin(login));  validTransaction();  **return** vo;  }  } |

1. **La couche présentation**

|  |
| --- |
| **La classe Utils :** |
| **package** ma.gov.formation.cigma.presentation;  **import** javax.faces.context.FacesContext;  **import** javax.servlet.http.HttpServletRequest;  **import** javax.servlet.http.HttpSession;  **public** **class** Utils {  **public** **static** HttpSession getSession() {  **return** (HttpSession) FacesContext.*getCurrentInstance*()  .getExternalContext().getSession(**false**);  }  **public** **static** HttpServletRequest getRequest() {  **return** (HttpServletRequest) FacesContext.*getCurrentInstance*()  .getExternalContext().getRequest();  }  } |

|  |
| --- |
| **La classe Constants :** |
| **package** ma.gov.formation.cigma.presentation;  **public** **interface** Constants {  **public** **static** **final** String ***ADMIN\_ROLE***="ADMIN";  **public** **static** **final** String ***CLIENT\_ROLE***="CLIENT";  **public** **static** **final** String ***ADMIN\_ROLE\_DESCRIPTION***="L'administrateur";  **public** **static** **final** String ***CLIENT\_ROLE\_DESCRIPTION***="Le client";  **public** **static** **final** String ***ERROR\_MSG***="ERROR\_MSG";  **public** **static** **final** String ***INFO\_MSG***="INFO\_MSG";  **public** **static** **final** String ***IMAGES\_ARTICLES\_PATH*** = "c:/photos\_articles/";  } |

|  |
| --- |
| **La classe BaseAction:** |
| **package** ma.gov.formation.cigma.presentation;  **import** java.util.ResourceBundle;  **import** javax.faces.application.FacesMessage;  **import** javax.faces.bean.ManagedProperty;  **import** javax.faces.context.FacesContext;  **public** **class** BaseAction {  @ManagedProperty("#{msg}")  **protected** ResourceBundle bundle;  **public** ResourceBundle getBundle() {  **return** bundle;  }  **public** **void** setBundle(ResourceBundle bundle) {  **this**.bundle = bundle;  }  **public** **void** addMessage(String message, String severity) {  FacesMessage msg = **null**;  msg = **new** FacesMessage(bundle.getString(message));  **if** (severity.equals(Constants.***ERROR\_MSG***))  msg.setSeverity(FacesMessage.***SEVERITY\_ERROR***);  **if** (severity.equals(Constants.***INFO\_MSG***))  msg.setSeverity(FacesMessage.***SEVERITY\_INFO***);    FacesContext.*getCurrentInstance*().addMessage(**null**, msg);  }    **public** String toDo() {  **return** "/toDo.xhtml";  }  } |

|  |
| --- |
| **La classe ArticleBean :** |
| **package** ma.gov.formation.cigma.presentation.article;  **import** java.io.IOException;  **import** java.util.ArrayList;  **import** java.util.List;  **import** javax.annotation.PostConstruct;  **import** javax.faces.bean.ManagedBean;  **import** javax.faces.bean.RequestScoped;  **import** ma.gov.formation.cigma.domaine.article.ArticleVo;  **import** ma.gov.formation.cigma.domaine.categorie.CategorieVo;  **import** ma.gov.formation.cigma.presentation.BaseAction;  **import** ma.gov.formation.cigma.presentation.Constants;  **import** ma.gov.formation.cigma.service.IArticleService;  **import** ma.gov.formation.cigma.service.IFileService;  **import** ma.gov.formation.cigma.service.impl.ArticleServiceImpl;  **import** ma.gov.formation.cigma.service.impl.FileServiceImpl;  **import** org.primefaces.event.RowEditEvent;  **import** org.primefaces.model.UploadedFile;  @ManagedBean(name = "articleBean")  @RequestScoped  **public** **class** ArticleBean **extends** BaseAction {  **private** List<CategorieVo> categories;  **private** IArticleService service = **new** ArticleServiceImpl();  **private** List<ArticleVo> articlesVo = **new** ArrayList<>();  **private** ArticleVo articleVo = **new** ArticleVo();  **private** IFileService fileService = **new** FileServiceImpl();  **private** UploadedFile file;  **public** UploadedFile getFile() {  **return** file;  }  **public** **void** setFile(UploadedFile file) {  **this**.file = file;  }  @PostConstruct  **public** **void** init() {  **try** {  articlesVo = service.getAll(**new** ArticleVo());  categories = service.getAll(**new** CategorieVo());  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  }  }  **public** List<CategorieVo> getCategories() {  **return** categories;  }  **public** ArticleVo getArticleVo() {  **return** articleVo;  }  **public** **void** setArticleVo(ArticleVo articleVo) {  **this**.articleVo = articleVo;  }  **public** **void** setArticlesVo(List<ArticleVo> articlesVo) {  **this**.articlesVo = articlesVo;  }  **public** List<ArticleVo> getArticlesVo() {  **return** articlesVo;  }  **public** String addArticle() {  **try** {  fileService.copyPhotoArticle(file.getInputstream(),  file.getFileName());  articleVo.setPhoto(Constants.***IMAGES\_ARTICLES\_PATH***  + file.getFileName());  service.save(articleVo);  articlesVo = service.getAll(**new** ArticleVo());  articleVo.clear();  addMessage("article.ajoute.avec.succes", Constants.***INFO\_MSG***);  **return** **null**;  } **catch** (IOException e) {  addMessage("probleme.copie.fichier", Constants.***ERROR\_MSG***);  e.printStackTrace();  **return** **null**;  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  **return** **null**;  }  }  **public** String removeArticle(Integer id) {  **try** {  service.remove(id);  articlesVo = service.getAll(**new** ArticleVo());  articleVo.clear();  addMessage("article.supprime.avec.succes", Constants.***INFO\_MSG***);  **return** **null**;  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  **return** **null**;  }  }  **public** String list() {  **return** "/admin/article/articles.xhtml";  }  **public** **void** onRowEdit(RowEditEvent event) {  **try** {  articleVo = (ArticleVo) event.getObject();  addMessage("article.modifier.avec.succes", Constants.***INFO\_MSG***);  service.save(articleVo);  articlesVo = service.getAll(**new** ArticleVo());  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  }  }  **public** **void** onRowCancel(RowEditEvent event) {  addMessage("modifications.annulees", Constants.***INFO\_MSG***);  }  } |

|  |
| --- |
| **La classe LoginBean :** |
| **package** ma.gov.formation.cigma.presentation.authentication;  **import** javax.faces.application.ConfigurableNavigationHandler;  **import** javax.faces.bean.ManagedBean;  **import** javax.faces.bean.SessionScoped;  **import** javax.faces.context.FacesContext;  **import** javax.faces.event.ComponentSystemEvent;  **import** javax.servlet.http.HttpSession;  **import** ma.gov.formation.cigma.domaine.user.UserVo;  **import** ma.gov.formation.cigma.presentation.BaseAction;  **import** ma.gov.formation.cigma.presentation.Constants;  **import** ma.gov.formation.cigma.presentation.Utils;  **import** ma.gov.formation.cigma.service.IUserService;  **import** ma.gov.formation.cigma.service.impl.UserServiceImpl;  @ManagedBean(name = "loginBean")  @SessionScoped  **public** **class** LoginBean **extends** BaseAction {  **private** IUserService service = **new** UserServiceImpl();  **private** UserVo userVo = **new** UserVo();  **public** UserVo getUserVo() {  **return** userVo;  }  // logout event, invalidate session  **public** String logout() {  HttpSession session = Utils.*getSession*();  session.invalidate();  **return** "/authentication/login";  }  **public** **void** setUserVo(UserVo userVo) {  **this**.userVo = userVo;  }  **public** String login() {  UserVo userVoP;  **try** {  userVoP = service.getByLogin(userVo.getUserName());  **if** (userVoP == **null**) {  addMessage("login.doesnt.exist", Constants.***ERROR\_MSG***);  **return** "/authentication/login";  }  **if** (!userVo.getPassword().equals(userVoP.getPassword())) {  addMessage("password.incorrect", Constants.***ERROR\_MSG***);  **return** "/authentication/login";  }  userVo = userVoP;  HttpSession session = Utils.*getSession*();  session.setAttribute("user", userVo);  **return** "/mainTemplate/mainTemplate";  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  **return** "/authentication/login";  }  }  **public** **void** isAdmin(ComponentSystemEvent event) {  FacesContext fc = FacesContext.*getCurrentInstance*();  **if** (Utils.*getSession*().getAttribute("user") == **null**) {  ConfigurableNavigationHandler nav = (ConfigurableNavigationHandler) fc  .getApplication().getNavigationHandler();  addMessage("must.authentify", Constants.***ERROR\_MSG***);  nav.performNavigation("/authentication/login");  **return**;  }  **if** (!userVo.haseRole(Constants.***ADMIN\_ROLE***)) {  ConfigurableNavigationHandler nav = (ConfigurableNavigationHandler) fc  .getApplication().getNavigationHandler();  nav.performNavigation("accessDenied");  }  }  **public** **void** isClient(ComponentSystemEvent event) {  FacesContext fc = FacesContext.*getCurrentInstance*();  **if** (Utils.*getSession*().getAttribute("user") == **null**) {  ConfigurableNavigationHandler nav = (ConfigurableNavigationHandler) fc  .getApplication().getNavigationHandler();  addMessage("must.authentify", Constants.***ERROR\_MSG***);  nav.performNavigation("/authentication/login");  **return**;  }  **if** (!userVo.haseRole(Constants.***CLIENT\_ROLE***)) {  ConfigurableNavigationHandler nav = (ConfigurableNavigationHandler) fc  .getApplication().getNavigationHandler();  nav.performNavigation("accessDenied");  }  }  } |

|  |
| --- |
| **La classe LangueBean :** |
| **package** ma.gov.formation.cigma.presentation.internationalization;  **import** java.io.Serializable;  **import** java.util.Locale;  **import** javax.faces.bean.ManagedBean;  **import** javax.faces.bean.SessionScoped;  **import** javax.faces.context.FacesContext;  **import** ma.gov.formation.cigma.presentation.BaseAction;  @ManagedBean(name = "langue")  @SessionScoped  **public** **class** LangueBean **extends** BaseAction **implements** Serializable {  **private** **static** **final** **long** ***serialVersionUID*** = 1L;  **private** Locale locale = **new** Locale("en");  **public** Locale getLocale() {  **return** locale;  }  **public** String getLanguage() {  **return** locale.getLanguage();  }  **public** **void** changeLanguage(String language) {  locale = **new** Locale(language);  FacesContext.*getCurrentInstance*().getViewRoot().setLocale(locale);  }  } |

|  |
| --- |
| **La classe PanierBean :** |
| **package** ma.gov.formation.cigma.presentation.panier;  **import** java.util.ArrayList;  **import** java.util.List;  **import** javax.faces.bean.ManagedBean;  **import** javax.faces.bean.SessionScoped;  **import** ma.gov.formation.cigma.presentation.BaseAction;  @ManagedBean(name = "panierBean")  @SessionScoped  **public** **class** PanierBean **extends** BaseAction {  **private** **static** List<String> *images*;  **static** {  *images* = **new** ArrayList<String>();  **for** (**int** i = 1; i <= 6; i++) {  *images*.add("/images/articles/photo" + i + ".jpg");  }  }  **public** List<String> getImages() {  **return** *images*;  }  **public** String list() {  **return** "/panier/panier.xhtml";  }  } |

|  |
| --- |
| **La classe ThemeBean :** |
| **package** ma.gov.formation.cigma.presentation.theme;  **import** java.io.Serializable;  **import** java.util.List;  **import** javax.annotation.PostConstruct;  **import** javax.faces.bean.ApplicationScoped;  **import** javax.faces.bean.ManagedBean;  **import** ma.gov.formation.cigma.domaine.theme.ThemeVo;  **import** ma.gov.formation.cigma.presentation.BaseAction;  **import** ma.gov.formation.cigma.presentation.Constants;  **import** ma.gov.formation.cigma.service.IThemeService;  **import** ma.gov.formation.cigma.service.impl.ThemeServiceImpl;  @ManagedBean(name = "themeBean", eager = **true**)  @ApplicationScoped  **public** **class** ThemeBean **extends** BaseAction **implements** Serializable{  **private** **static** **final** **long** ***serialVersionUID*** = 2510779045682332364L;  **private** List<ThemeVo> themes;  **private** String selectedTheme;  IThemeService themeService=**new** ThemeServiceImpl();  **public** String getSelectedTheme() {  **return** selectedTheme;  }  **public** **void** setSelectedTheme(String selectedTheme) {  **this**.selectedTheme = selectedTheme;  }  @PostConstruct  **public** **void** init() {  **try** {  setSelectedTheme(themeService.getDefaultTheme().getCode());  themes = themeService.getAll(**new** ThemeVo());  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  }    }  **public** **void** setThemes(List<ThemeVo> themes) {  **this**.themes = themes;  }  **public** List<ThemeVo> getThemes() {  **return** themes;  }  **public** String edit() {  **return** "/admin/theme.xhtml";  }  **public** **void** saveTheme(ThemeVo theme) {  **try** {  themeService.updateThemeByDefault(selectedTheme);  setSelectedTheme(selectedTheme);  } **catch** (Exception e) {  addMessage("probleme.technique", Constants.***ERROR\_MSG***);  e.printStackTrace();  }  }  } |

1. **Le fichier persistence.xml**

|  |
| --- |
| **Créer le fichier persistence.xml dans le dossier src/META-INF** |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <persistence version=*"2.1"*  xmlns=*"http://xmlns.jcp.org/xml/ns/persistence"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence\_2\_1.xsd"*>  <persistence-unit name=*"unite1"*>  <provider>org.hibernate.ejb.HibernatePersistence</provider>  <class>ma.gov.formation.cigma.service.modele.Article</class>  <class>ma.gov.formation.cigma.service.modele.User</class>  <class>ma.gov.formation.cigma.service.modele.UserRole</class>  <class>ma.gov.formation.cigma.service.modele.Categorie</class>  <class>ma.gov.formation.cigma.service.modele.Theme</class>  <properties>  <!-- Classes persistantes -->  <property name=*"hibernate.show\_sql"* value=*"true"* />  <property name=*"hibernate.format\_sql"* value=*"true"* />  <property name=*"use\_sql\_comments"* value=*"true"* />  <!-- connexion JDBC -->  <property name=*"hibernate.connection.driver\_class"* value=*"com.mysql.jdbc.Driver"* />  <property name=*"hibernate.connection.url"* value=*"jdbc:mysql://localhost:3306/formation\_db"* />  <property name=*"hibernate.connection.username"* value=*"root"* />  <property name=*"hibernate.connection.password"* value=*"root"* />  <!-- création automatique du schéma -->  <property name=*"hibernate.hbm2ddl.auto"* value=*"update"* />  <!-- Dialecte -->  <property name=*"hibernate.dialect"* value=*"org.hibernate.dialect.MySQL5InnoDBDialect"* />  </properties>  </persistence-unit>  </persistence> |

1. **Les fichiers \*.xhtml**

|  |
| --- |
| **Créer la page WebContent/mainTemplate/mainTemplate.xml** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head />  <h:body>  <p:layout fullPage=*"true"*>  <f:view locale=*"#{langue.locale}"*>  <p:layoutUnit position=*"north"* size=*"60"*>  <ui:include src=*"/mainTemplate/commonHeader.xhtml"* />  </p:layoutUnit>  <p:layoutUnit position=*"west"* size=*"250"* resizable=*"true"*  closable=*"true"* collapsible=*"true"*>  <ui:include src=*"/mainTemplate/commonMenu.xhtml"* />  </p:layoutUnit>  <p:layoutUnit position=*"center"*>  <h:form id=*"templatePrincipal"*>  <p:growl id=*"msgs"* showDetail=*"true"*/>  </h:form>  <ui:insert name=*"body"*>  <ui:include src=*"/mainTemplate/commonBody.xhtml"* />  </ui:insert>  </p:layoutUnit>  <p:layoutUnit position=*"east"* size=*"320"* resizable=*"true"*>  <p:calendar id=*"inline"* value=*""* mode=*"inline"* />  </p:layoutUnit>  <p:layoutUnit position=*"south"* size=*"80"*>  <ui:include src=*"/mainTemplate/commonFooter.xhtml"* />  </p:layoutUnit>  </f:view>  </p:layout>  </h:body>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/mainTemplate/commonHeader.xml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head />  <ui:composition>  <h:form>  <p:menuButton value=*"#{msg.preference}"*>  <p:menuitem value=*"#{msg.profile}"* action=*"#{loginBean.toDo}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg.updatepwd}"* action=*"#{loginBean.toDo}"* ajax=*"false"*/>  <p:separator/>  <p:menuitem value=*"#{msg.logout}"* action=*"#{loginBean.logout}"* ajax=*"false"*/>  </p:menuButton>  Connected User : <h:outputLabel value=*"#{user.userName}"*/>  <h:commandLink action=*"#{langue.changeLanguage('ar')}"* style="float:*right*">  <h:graphicImage value=*"/resources/images/internationalization/ar.png"* />  </h:commandLink>  <h:commandLink action=*"#{langue.changeLanguage('fr')}"* style="float:*right*">  <h:graphicImage value=*"/resources/images/internationalization/fr.png"* />  </h:commandLink>  <h:commandLink action=*"#{langue.changeLanguage('en')}"* style="float:*right*">  <h:graphicImage value=*"/resources/images/internationalization/en.png"* />  </h:commandLink>  <br></br>      </h:form>  </ui:composition>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/mainTemplate/commonMenu.xml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head/>  <h:body>  <ui:composition>  <h:form>  <p:panelMenu style="width:*300px*">  <p:submenu label=*"#{msg['menu.admin']}"* rendered=*"#{user.admin}"*>  <p:menuitem value=*"#{msg['menu.admin.sm.gestArt']}"* action=*"#{articleBean.list}"* ajax=*"false"*/>  <p:submenu label=*"#{msg['menu.admin.sm.gestUtil']}"* icon=*"pi pi-home"*>  <p:menuitem value=*"#{msg['menu.admin.sm.gestUtil.sm.addUser']}"* action=*"#{articleBean.toDo}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg['menu.admin.sm.gestUtil.sm.addRole']}"* action=*"#{articleBean.toDo}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg['menu.admin.sm.gestUtil.sm.listUsers']}"* action=*"#{articleBean.toDo}"* ajax=*"false"*/>  </p:submenu>  </p:submenu>  <p:separator />  <p:submenu label=*"#{msg['menu.preference']}"* rendered=*"#{user.admin}"*>  <p:menuitem value=*"#{msg['menu.preference.sm.Theme']}"* action=*"#{themeBean.edit}"* ajax=*"false"*/>  </p:submenu>  <p:separator />  <p:submenu label=*"#{msg['menu.panier']}"* rendered=*"#{user.client}"*>  <p:menuitem value=*"#{msg['menu.panier.sb.consulter']}"* action=*"#{panierBean.list}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg['menu.panier.sb.edit']}"* action=*"#{panierBean.toDo}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg['menu.panier.sb.acheter']}"* action=*"#{panierBean.toDo}"* ajax=*"false"*/>  <p:menuitem value=*"#{msg['menu.panier.sb.editerfact']}"* action=*"#{panierBean.toDo}"* ajax=*"false"*/>  </p:submenu>  </p:panelMenu>  </h:form>  </ui:composition>  </h:body>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/mainTemplate/commonFooter.xml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head/>  <h:body>  <ui:composition>  <h:form>  <p:dock position=*"bottom"*>  <p:menuitem value=*"Home"* icon=*"/resources/images/footer/home.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"Music"* icon=*"/resources/images/footer/music.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"Video"* icon=*"/resources/images/footer/video.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"Email"* icon=*"/resources/images/footer/email.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"Portfolio"* icon=*"/resources/images/footer/portfolio.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"Link"* icon=*"/resources/images/footer/link.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"RSS"* icon=*"/resources/images/footer/rss.png"* action=*"#{loginBean.toDo}"*/>  <p:menuitem value=*"History"* icon=*"/resources/images/footer/history.png"* action=*"#{loginBean.toDo}"*/>  </p:dock>  </h:form>  </ui:composition>  </h:body>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/mainTemplate/commonBody.xml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head/>  <h:body>  <ui:composition>  Contenu à mettre ici  </ui:composition>  </h:body>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/admin/article/articles.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head/>  <ui:composition template=*"/mainTemplate/mainTemplate.xhtml"*>  <ui:define name=*"body"*>  <ui:include src=*"/admin/article/ajouter.xhtml"* />  <ui:include src=*"/admin/article/liste.xhtml"* />  </ui:define>  </ui:composition>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/admin/article/ajouter.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <ui:composition>  <h:form enctype=*"multipart/form-data"*>  <p:panelGrid columns=*"2"*>  <f:facet name=*"header"*>  <h:outputText value=*"#{msg['ajouter.nouvel.article']}"* />  </f:facet>  <h:outputLabel value=*"#{msg['formulaire.article.designation']}"* />  <h:inputText id=*"designation"*  value=*"#{articleBean.articleVo.designation}"* required=*"true"*  immediate=*"true"*>  </h:inputText>  <h:outputLabel value=*"#{msg['formulaire.article.prix']}"* />  <h:inputText id=*"price"* value=*"#{articleBean.articleVo.prix}"* />  <h:outputLabel value=*"#{msg['formulaire.article.quantiteEnStock']}"* />  <h:inputText id=*"quantiteEnStock"* value=*"#{articleBean.articleVo.quantiteEnStock}"* />    <h:outputLabel value=*"Catégorie"* />  <h:selectOneMenu value=*"#{articleBean.articleVo.categorie.code}"* style="width:*100%*">  <f:selectItems value=*"#{articleBean.categories}"* var=*"categorie"* itemLabel=*"#{categorie.libelle}"* itemValue=*"#{categorie.code}"*/>  </h:selectOneMenu>    <h:outputLabel value=*"Photo"* />  <p:fileUpload value=*"#{articleBean.file}"* mode=*"simple"* skinSimple=*"true"* allowTypes=*"/(\.|\/)(gif|jpe?g|png)$/"*/>  <f:facet name=*"footer"*>  <h:panelGroup style="display:*block*; text-align:*center*">  <p:commandButton value=*"#{msg['formulaire.bouton.ajouter']}"* ajax=*"false"* action=*"#{articleBean.addArticle}"*/>  </h:panelGroup>  </f:facet>  </p:panelGrid>  </h:form>  </ui:composition>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/admin/article/liste.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head/>  <ui:composition>  <h:form>  <h2>#{msg['message.listeDesArticles']}</h2>  <p:dataTable id=*"tbl"* var=*"article"* value=*"#{articleBean.articlesVo}"* editable=*"true"* paginatorTemplate=*"{CurrentPageReport} {FirstPageLink} {PreviousPageLink} {PageLinks} {NextPageLink} {LastPageLink} {Exporters}"*  paginator=*"true"* rows=*"5"* style="margin-bottom:*20px*">  <f:facet name=*"{Exporters}"*>  <div style="float: *right*; padding-top: *5px*;">  <h:commandLink>  <p:graphicImage name=*"images/export/excel.png"* width=*"24"* />  <p:dataExporter type=*"xls"* target=*"tbl"* fileName=*"#{msg.file\_name}"* />  </h:commandLink>  <h:commandLink>  <p:graphicImage name=*"images/export/pdf.png"* width=*"24"* />  <p:dataExporter type=*"pdf"* target=*"tbl"* fileName=*"#{msg.file\_name}"*/>  </h:commandLink>  <h:commandLink>  <p:graphicImage name=*"images/export/csv.png"* width=*"24"* />  <p:dataExporter type=*"csv"* target=*"tbl"* fileName=*"#{msg.file\_name}"* />  </h:commandLink>  <h:commandLink>  <p:graphicImage name=*"images/export/xml.png"* width=*"24"* />  <p:dataExporter type=*"xml"* target=*"tbl"* fileName=*"#{msg.file\_name}"* />  </h:commandLink>  </div>  </f:facet>  <p:ajax event=*"rowEdit"* listener=*"#{articleBean.onRowEdit}"* update=*":templatePrincipal:msgs"* />  <p:ajax event=*"rowEditCancel"* listener=*"#{articleBean.onRowCancel}"* update=*":templatePrincipal:msgs"* />      <p:column headerText=*"#{msg['intitule.colonne.identifiant.listeDesArticles']}"*>  <p:cellEditor>  <f:facet name=*"output"*><h:outputText value=*"#{article.id}"* /></f:facet>  <f:facet name=*"input"*><p:inputText id=*"modelInputid"* value=*"#{article.id}"* style="width:*100%*"/></f:facet>  </p:cellEditor>  </p:column>    <p:column headerText=*"#{msg['intitule.colonne.designation.listeDesArticles']}"*>  <p:cellEditor>  <f:facet name=*"output"*><h:outputText value=*"#{article.designation}"* /></f:facet>  <f:facet name=*"input"*><p:inputText id=*"modelInputdesc"* value=*"#{article.designation}"* style="width:*100%*"/></f:facet>  </p:cellEditor>  </p:column>    <p:column headerText=*"#{msg['intitule.colonne.prix.listeDesArticles']}"*>  <p:cellEditor>  <f:facet name=*"output"*><h:outputText value=*"#{article.prix}"* /></f:facet>  <f:facet name=*"input"*><p:inputText id=*"modelInputprix"* value=*"#{article.prix}"* style="width:*100%*"/></f:facet>  </p:cellEditor>  </p:column>      <p:column headerText=*"#{msg['intitule.colonne.quantiteEnStock.listeDesArticles']}"*>  <p:cellEditor>  <f:facet name=*"output"*><h:outputText value=*"#{article.quantiteEnStock}"* /></f:facet>  <f:facet name=*"input"*><p:inputText id=*"modelInputquan"* value=*"#{article.quantiteEnStock}"* style="width:*100%*"/></f:facet>  </p:cellEditor>  </p:column>    <p:column headerText=*"Catgorie"*>  <p:cellEditor>  <f:facet name=*"output"*><h:outputText value=*"#{article.categorie.libelle}"* /></f:facet>  <f:facet name=*"input"*>  <h:selectOneMenu value=*"#{article.categorie.code}"* style="width:*100%*">  <f:selectItems value=*"#{articleBean.categories}"* var=*"categorie"* itemLabel=*"#{categorie.code}"* itemValue=*"#{categorie.libelle}"* />  </h:selectOneMenu>  </f:facet>  </p:cellEditor>  </p:column>  <p:column style="width:*35px*" exportable=*"false"*>  <p:rowEditor />  </p:column>  </p:dataTable>  </h:form>  </ui:composition>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/admin/theme.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <h:head />  <ui:composition template=*"/mainTemplate/mainTemplate.xhtml"*>  <ui:define name=*"body"*>  <h:form>  <h:panelGrid columns=*"2"* cellpadding=*"10"*>  <p:outputLabel for=*"basic"* value=*"Basic:"* />  <p:themeSwitcher id=*"basic"* style="width:*165px*" value=*"#{themeBean.selectedTheme}"*>  <f:selectItem itemLabel=*"#{msg['choose.theme']}"* itemValue=*""* />  <f:selectItems value=*"#{themeBean.themes}"* var=*"theme"* itemLabel=*"#{theme.libelle}"* itemValue=*"#{theme.code}"*/>  <f:ajax listener=*"#{themeBean.saveTheme(theme)}"*></f:ajax>  </p:themeSwitcher>  </h:panelGrid>  </h:form>  </ui:define>  </ui:composition>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/** **authentication/login.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <f:view locale=*"#{langue.locale}"*>  <h:head />  <h:body>  <h:form id=*"LoginForm"*>  <p:growl id=*"msgs"* showDetail=*"true"* />  <p:panelGrid columns=*"2"*>  <f:facet name=*"header"*>  <h:outputText value=*"#{msg['se.connecter']}"* />  </f:facet>  <p:outputLabel value=*"#{msg['formulaire.login.username']}"* />  <p:inputText id=*"userName"* value=*"#{loginBean.userVo.userName}"* size=*"40"* required=*"true"* />  <p:outputLabel value=*"#{msg['formulaire.login.password']}"* />  <p:password id=*"password"* value=*"#{loginBean.userVo.password}"* size=*"40"* required=*"true"* />  <f:facet name=*"footer"*>  <h:panelGroup>  <p:commandButton value=*"#{msg['login.bouton.sautentifier']}"* action=*"#{loginBean.login}"* update=*":LoginForm:msgs"* />  </h:panelGroup>  </f:facet>  </p:panelGrid>  </h:form>  </h:body>  </f:view>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/panier/panier.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <f:view locale=*"#{langue.locale}"*>  <h:head />  <ui:composition template=*"/mainTemplate/mainTemplate.xhtml"*>  <ui:define name=*"body"*>  <h:form>  <h3>Liste des aticles par catégories :</h3>  <p:accordionPanel multiple=*"true"*>  <p:tab title=*"Matetriels informatiques"*>  <p:galleria value=*"#{panierBean.images}"* var=*"image"*  panelWidth=*"600"* panelHeight=*"200"* showCaption=*"true"*>  <p:graphicImage name=*"#{image}"*  alt=*"Image Description for #{image}"* title=*"#{image}"* />  </p:galleria>  </p:tab>  <p:tab title=*"Sport"*>  <p:galleria value=*"#{panierBean.images}"* var=*"image"*  panelWidth=*"600"* panelHeight=*"200"* showCaption=*"true"*>  <p:graphicImage name=*"#{image}"*  alt=*"Image Description for #{image}"* title=*"#{image}"* />  </p:galleria>  </p:tab>  <p:tab title=*"Autre"*>  <p:galleria value=*"#{panierBean.images}"* var=*"image"*  panelWidth=*"600"* panelHeight=*"200"* showCaption=*"true"*>  <p:graphicImage name=*"#{image}"*  alt=*"Image Description for #{image}"* title=*"#{image}"* />  </p:galleria>  </p:tab>  </p:accordionPanel>  </h:form>  </ui:define>  </ui:composition>  </f:view>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/accessDenied.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <f:view locale=*"#{langue.locale}"*>  <h:head />  <ui:composition template=*"/mainTemplate/mainTemplate.xhtml"*>  <ui:define name=*"body"*>  <h:form>  <h3>  <h:outputFormat value=*"#{msg['message.access.denied']}"*>  <f:param value=*"#{user.userName}"*></f:param></h:outputFormat>  </h3>  </h:form>  </ui:define>  </ui:composition>  </f:view>  </h:html> |

|  |
| --- |
| **Créer la page WebContent/toDo.xhtml :** |
| <h:html lang=*"#{langue.language}"* xmlns=*"http://www.w3.org/1999/xhtml"*  xmlns:f=*"http://xmlns.jcp.org/jsf/core"*  xmlns:ui=*"http://xmlns.jcp.org/jsf/facelets"*  xmlns:a=*"http://xmlns.jcp.org/jsf/passthrough"*  xmlns:c=*"http://java.sun.com/jsp/jstl/core"*  xmlns:p=*"http://primefaces.org/ui"*  xmlns:h=*"http://xmlns.jcp.org/jsf/html"*>  <f:view locale=*"#{langue.locale}"*>  <h:head />  <ui:composition template=*"/mainTemplate/mainTemplate.xhtml"*>  <ui:define name=*"body"*>  <h:form>  <h3>To be completed</h3>  </h:form>  </ui:define>  </ui:composition>  </f:view>  </h:html> |

1. **Les fichier web.xhtml**

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://xmlns.jcp.org/xml/ns/javaee"* xsi:schemaLocation=*"http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"* version=*"3.1"*>  <display-name>tp2</display-name>  <welcome-file-list>  <welcome-file>/authentication/login.xhtml</welcome-file>  </welcome-file-list>  <servlet>  <servlet-name>Faces Servlet</servlet-name>  <servlet-class>javax.faces.webapp.FacesServlet</servlet-class>  <load-on-startup>1</load-on-startup>  </servlet>  <context-param>  <param-name>primefaces.THEME</param-name>  <param-value>#{themeBean.selectedTheme}</param-value>  </context-param>  <servlet-mapping>  <servlet-name>Faces Servlet</servlet-name>  <url-pattern>\*.xhtml</url-pattern>  </servlet-mapping>  </web-app> |

1. **Le fihier faces-config.xml**

|  |
| --- |
| **WebContent/WEB-INF/faces-config.xml** |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <faces-config xmlns=*"http://xmlns.jcp.org/xml/ns/javaee"*  xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-facesconfig\_2\_2.xsd"*  version=*"2.2"*>  <application>  <locale-config>  <default-locale>en</default-locale>  <supported-locale>fr</supported-locale>  <supported-locale>ar</supported-locale>  </locale-config>  <resource-bundle>  <base-name>messages</base-name>  <var>msg</var>  </resource-bundle>  </application>  </faces-config> |

1. **Le fichier messages.properties**

|  |
| --- |
| **src/messages.properties :** |
| #page articles.xhtml et edit.xhtml  titre=Web Application with : JSF, JPA and Wildfly  formulaire.article.designation=Article Designation  formulaire.article.prix=Price  formulaire.article.quantiteEnStock=Quantity in stock  #page articles.xhtml  ajouter.nouvel.article=Add a new article :  formulaire.bouton.ajouter=Add  message.listeDesArticles=Articles list :  intitule.colonne.identifiant.listeDesArticles=Identifier  intitule.colonne.designation.listeDesArticles=Designation  intitule.colonne.prix.listeDesArticles=Price  intitule.colonne.quantiteEnStock.listeDesArticles=Quantity in stock  intitule.lien.supprimer=Delete  intitule.lien.modifier=Update  #page edit.xhtml  modifier.article=Update an article :  formulaire.bouton.modifier=Update  #page login.xhtml  se.connecter=Autentication :  formulaire.login.username=Login  formulaire.login.password=Password  login.bouton.sautentifier=Login  logout=Logout  #page panier.xhtml  msg.panier=Panier  #page accessDenied.xhtml  message.access.denied=Hello **{0}** : You are not allowed to access this page. Please contact the administrator.  lien.revenir.page.login=Go Back to authentication page  #Page menu :  menu.admin=Administration  menu.admin.sm.gestArt=Article Management  menu.admin.sm.gestUtil=User Management  menu.admin.sm.gestUtil.sm.addUser=Create User  menu.admin.sm.gestUtil.sm.addRole=Create Role  menu.admin.sm.gestUtil.sm.listUsers=Users List  menu.admin.sm.parametres=Parameters  menu.panier=Basket Management  menu.panier.sb.consulter=Consult Articles  menu.panier.sb.edit=Consulter panier  menu.panier.sb.acheter=Buy Article  menu.panier.sb.editerfact=Edit Invoice  menu.preference=Preference  menu.preference.sm.Theme=Update Theme  footer=JSF training: use of the facelets  logout=Logout  # Page theme :  choose.theme=Choose Theme  #Validations messages  login.doesnt.exist=Login doesnt exist  password.incorrect=Password incorrect  problem.user.role=You dont have any role. Please contact your administrator  save.update=Enregistrer les modifications  cancel.update=Annuler les modifications  must.authentify=Merci de vous vous authentifiez pour accéder à l'application.  #Infos messages  article.ajoute.avec.succes=Article added successfully.  article.supprime.avec.succes=article successfully deleted.  article.modifier.avec.succes=Articles successfully edited.  modifications.annulees=Modifications canceled.  probleme.copie.fichier=Problem of uploading the photo of the article.  #Page header  preference=Preference  profile=Profile  updatepwd=Update password  probleme.technique=Technical problem.  file\_name=Liste\_Articles |

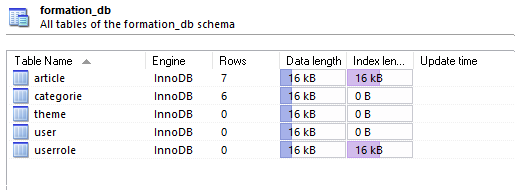
1. **Les images**

Copier le dossier images dans le répertoire WebContent/resources.

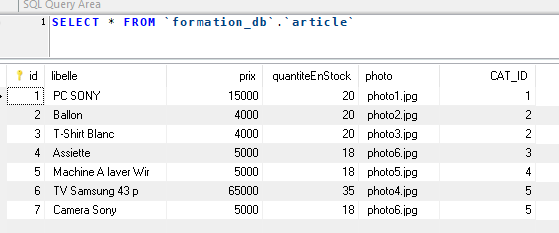
1. **Initialisation de la base de données :**

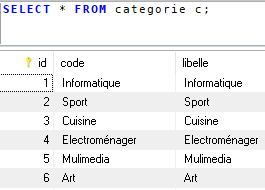
|  |
| --- |
| **Créer la classe test.article.Test et exécuter la méthode main ci-dessous afin d'insérer des données au niveau des deux tables CATEGORIE et ARTICLE :** |
| **package** test.article;  **import** ma.gov.formation.cigma.dao.IArticleDao;  **import** ma.gov.formation.cigma.dao.impl.ArticleDaoImpl;  **import** ma.gov.formation.cigma.service.modele.Article;  **import** ma.gov.formation.cigma.service.modele.Categorie;  **public** **class** Test {  **private** **static** IArticleDao *dao* = **new** ArticleDaoImpl();  **public** **static** **void** main(String[] args) {  **try** {  *dao*.beginTransaction();    Categorie c1=**new** Categorie("Informatique","Informatique");  Categorie c2=**new** Categorie("Sport","Sport");  Categorie c3=**new** Categorie("Cuisine","Cuisine");  Categorie c4=**new** Categorie("Electroménager","Electroménager");  Categorie c5=**new** Categorie("Mulimedia","Mulimedia");  Categorie c6=**new** Categorie("Art","Art");    Article a1 = **new** Article("PC SONY", 15000.0, 20,"photo1.jpg");  Article a2 = **new** Article("Ballon", 4000.0, 20,"photo2.jpg");  Article a3 = **new** Article("T-Shirt Blanc", 4000.0, 20,"photo3.jpg");  Article a4 = **new** Article("TV Samsung 43 p", 65000.0, 35,"photo4.jpg");  Article a5 = **new** Article("Machine A laver Wir", 5000.0, 18,"photo5.jpg");  Article a6 = **new** Article("Camera Sony", 5000.0, 18,"photo6.jpg");  Article a7 = **new** Article("Assiette", 5000.0, 18,"photo6.jpg");    c1.addArticle(a1);  c2.addArticle(a2); c2.addArticle(a3);  c5.addArticle(a4);c5.addArticle(a6);  c4.addArticle(a5);  c3.addArticle(a7);  *dao*.save(c1);  *dao*.save(c2);  *dao*.save(c3);  *dao*.save(c4);  *dao*.save(c5);  *dao*.save(c6);  *dao*.validTransaction();    } **catch** (Exception e) {  e.printStackTrace();  }  }  } |

Vérifier que les 05 tables suivantes ont été crées automatiquement :

****

Vérifier également que les données de test ont été insérées au niveau des deux tables CATEGORIE et ARTICLE.

****

****

|  |
| --- |
| **Créer la classe test.theme.Test et exécuter la méthode main ci-dessous :** |
| **package** test.theme;  **import** ma.gov.formation.cigma.dao.IBaseDao;  **import** ma.gov.formation.cigma.dao.impl.BaseDaoImpl;  **import** ma.gov.formation.cigma.service.modele.Theme;  **public** **class** Test {  **private** **static** IBaseDao *dao* = **new** BaseDaoImpl();  **public** **static** **void** main(String[] args) {  **try** {  *dao*.beginTransaction();  Theme theme1 = **new** Theme("glass-x", "glass-x");  Theme theme2 = **new** Theme("cruze", "cruze");  Theme theme3 = **new** Theme("blitzer", "blitzer");  Theme theme4 = **new** Theme("afternoon", "afternoon","YES");  Theme theme5 = **new** Theme("afterdark", "afterdark");  Theme theme6 = **new** Theme("bootstrap", "bootstrap");  Theme theme7 = **new** Theme("bluesky", "bluesky");  Theme theme8 = **new** Theme("midnight", "midnight");  Theme theme9 = **new** Theme("cupertino", "cupertino");  Theme theme10 = **new** Theme("aristo", "aristo");  Theme theme11 = **new** Theme("blitzer", "blitzer");  Theme theme12 = **new** Theme("casablanca", "casablanca");  Theme theme13 = **new** Theme("dark-hive", "dark-hive");  Theme theme14 = **new** Theme("delta", "delta");  Theme theme15 = **new** Theme("dot-luv", "dot-luv");  *dao*.save(theme1);  *dao*.save(theme2);  *dao*.save(theme3);  *dao*.save(theme4);  *dao*.save(theme5);  *dao*.save(theme6);  *dao*.save(theme7);  *dao*.save(theme8);  *dao*.save(theme9);  *dao*.save(theme10);  *dao*.save(theme11);  *dao*.save(theme12);  *dao*.save(theme13);  *dao*.save(theme14);  *dao*.save(theme15);  *dao*.validTransaction();  } **catch** (Exception e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }  }  } |

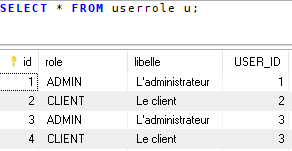
Vérfier que les données ont été insérées au niveau de la table THEME :

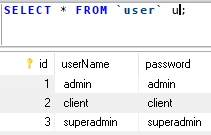


Remarquez que le thème par défaut qui est utilisé par l'application est **"afternoon".**

|  |
| --- |
| Créer la classe test.user.Test et exécuter la méthode main ci-dessous : |
| **package** test.user;  **import** ma.gov.formation.cigma.dao.IBaseDao;  **import** ma.gov.formation.cigma.dao.impl.BaseDaoImpl;  **import** ma.gov.formation.cigma.presentation.Constants;  **import** ma.gov.formation.cigma.service.modele.User;  **import** ma.gov.formation.cigma.service.modele.UserRole;  **public** **class** Test {  **private** **static** IBaseDao *dao* = **new** BaseDaoImpl();  **public** **static** **void** main(String[] args) {  **try** {  *dao*.beginTransaction();  UserRole adminRole1=**new** UserRole(Constants.***ADMIN\_ROLE***,Constants.***ADMIN\_ROLE\_DESCRIPTION***);  UserRole adminRole2=**new** UserRole(Constants.***ADMIN\_ROLE***,Constants.***ADMIN\_ROLE\_DESCRIPTION***);  UserRole clientRole1=**new** UserRole(Constants.***CLIENT\_ROLE***,Constants.***CLIENT\_ROLE\_DESCRIPTION***);  UserRole clientRole2=**new** UserRole(Constants.***CLIENT\_ROLE***,Constants.***CLIENT\_ROLE\_DESCRIPTION***);    User user1=**new** User("admin","admin");user1.addUserRole(adminRole1);  User user2=**new** User("client","client");user2.addUserRole(clientRole1);  User user3=**new** User("superadmin","superadmin");user3.addUserRole(adminRole2);user3.addUserRole(clientRole2);  *dao*.save(user1);  *dao*.save(user2);  *dao*.save(user3);  *dao*.validTransaction();  } **catch** (Exception e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }    }  } |

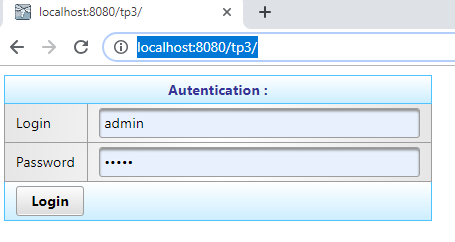
Vérfier que les données suivantes ont été bien insérées :



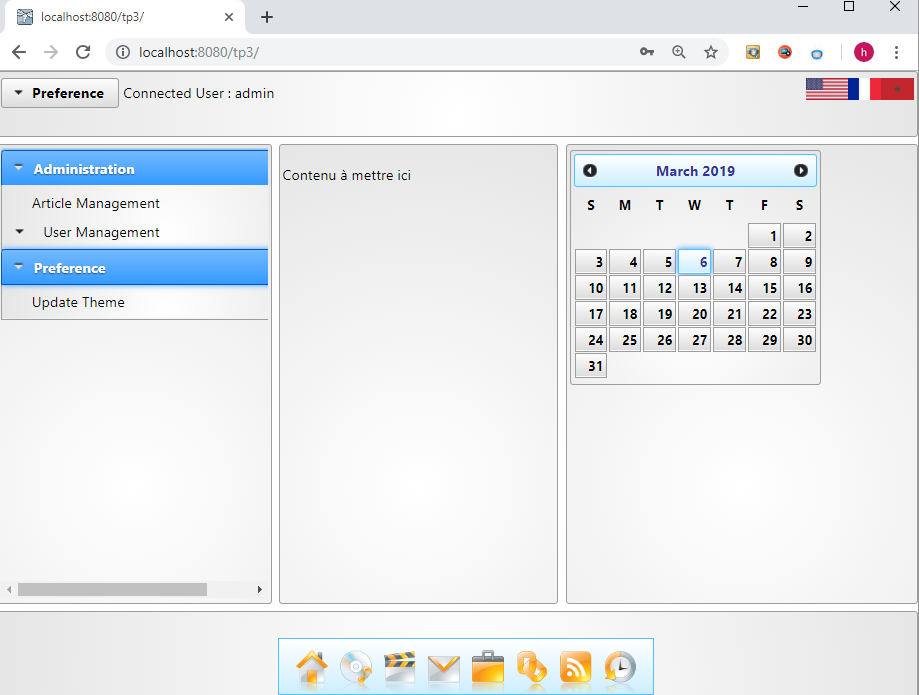


1. **Test de l'application**

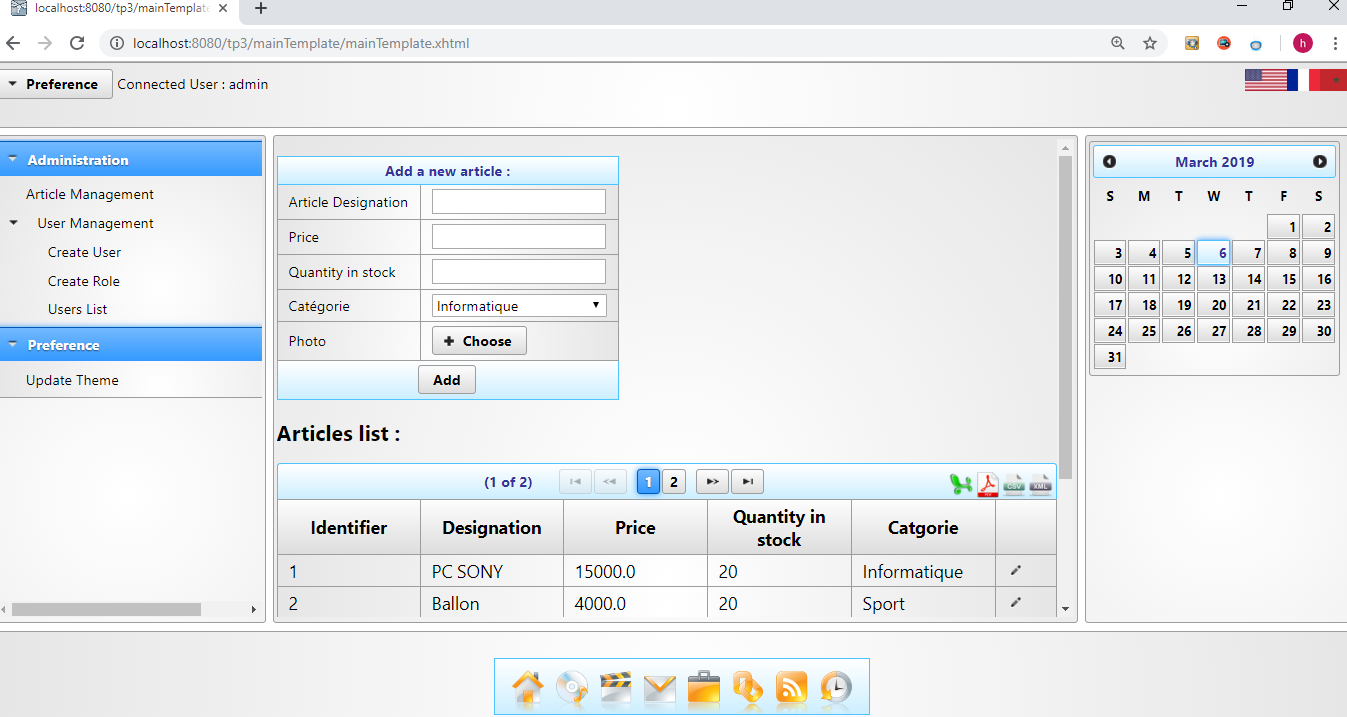
Déployer votre application au niveau de Wildfly et ensuite lancer le lien **http://localhost:8080/tp3.** La page suivante sera affichée :

****

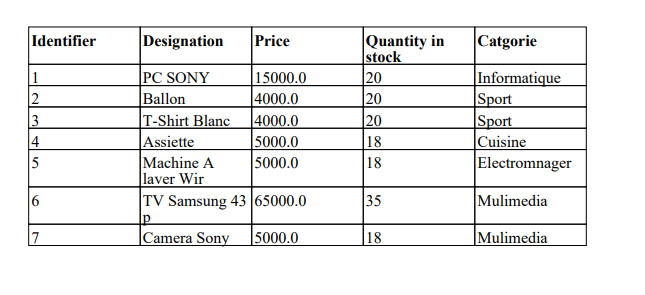
Entrer un compte utilisateur (par exemple : admin/admin). La page suivante sera affichée :



\* Pour la gestion des articles, cliquer sur le menu "Article Management" ci-dessus :

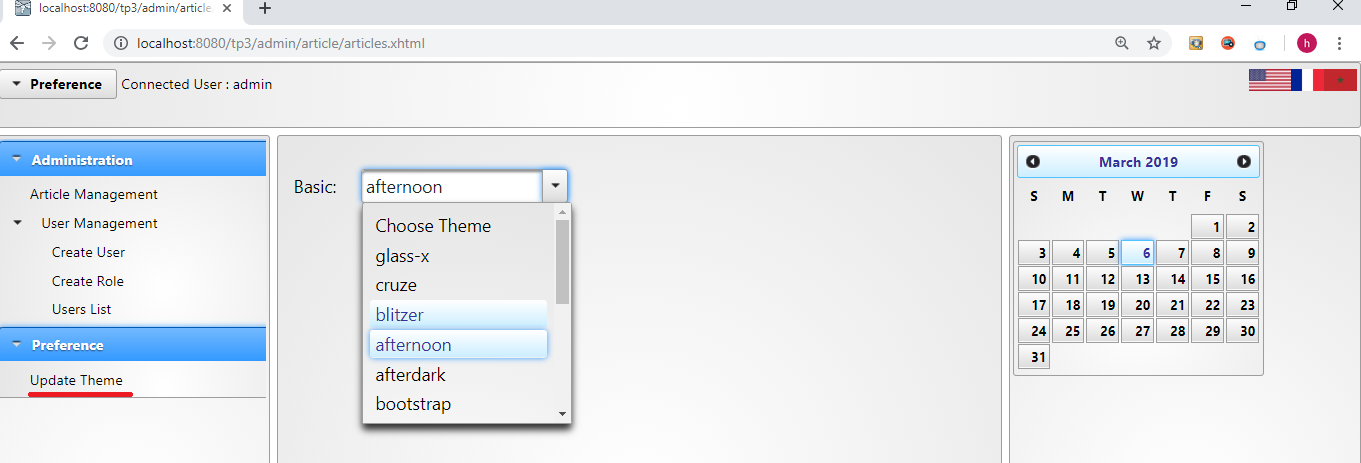


\* Pour exporter les articles en fichier PDF, cliquer sur l'icon "pdf" et vérifier que le fichier List\_Articles.pdf a été bien généré :

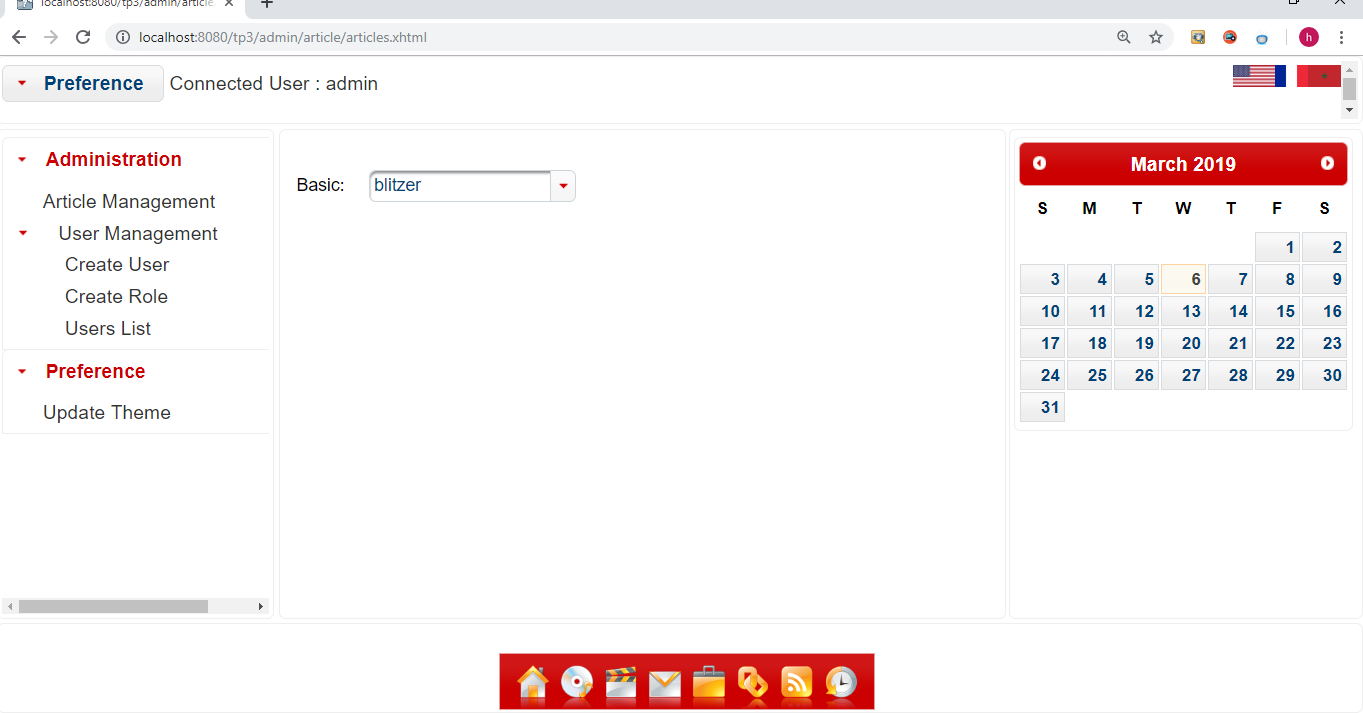


\* Idem pour les autres formats : Excel, XML et CSV.

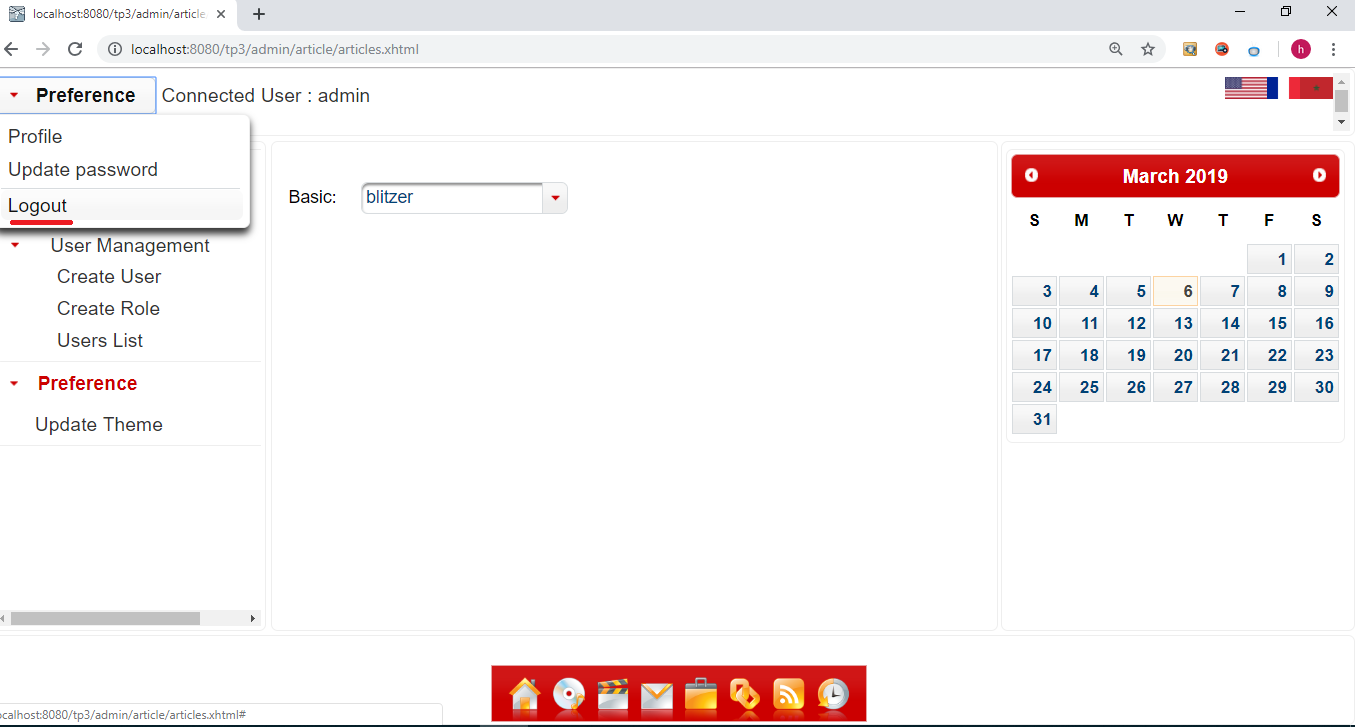
\* Pour changer le thème, cliquer sur "Update Theme" comme le montre l'écran suivant :



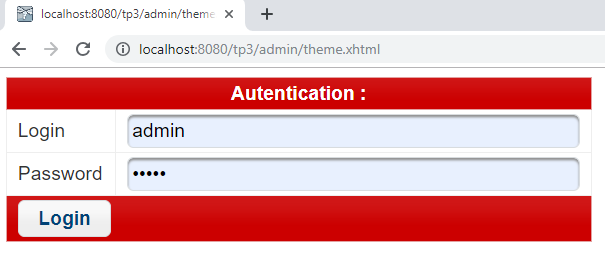
\* Choisir un autre thème, par exemple "blitzer". Vérifier que le thème a été appliqué à l'ensemble de vos pages :



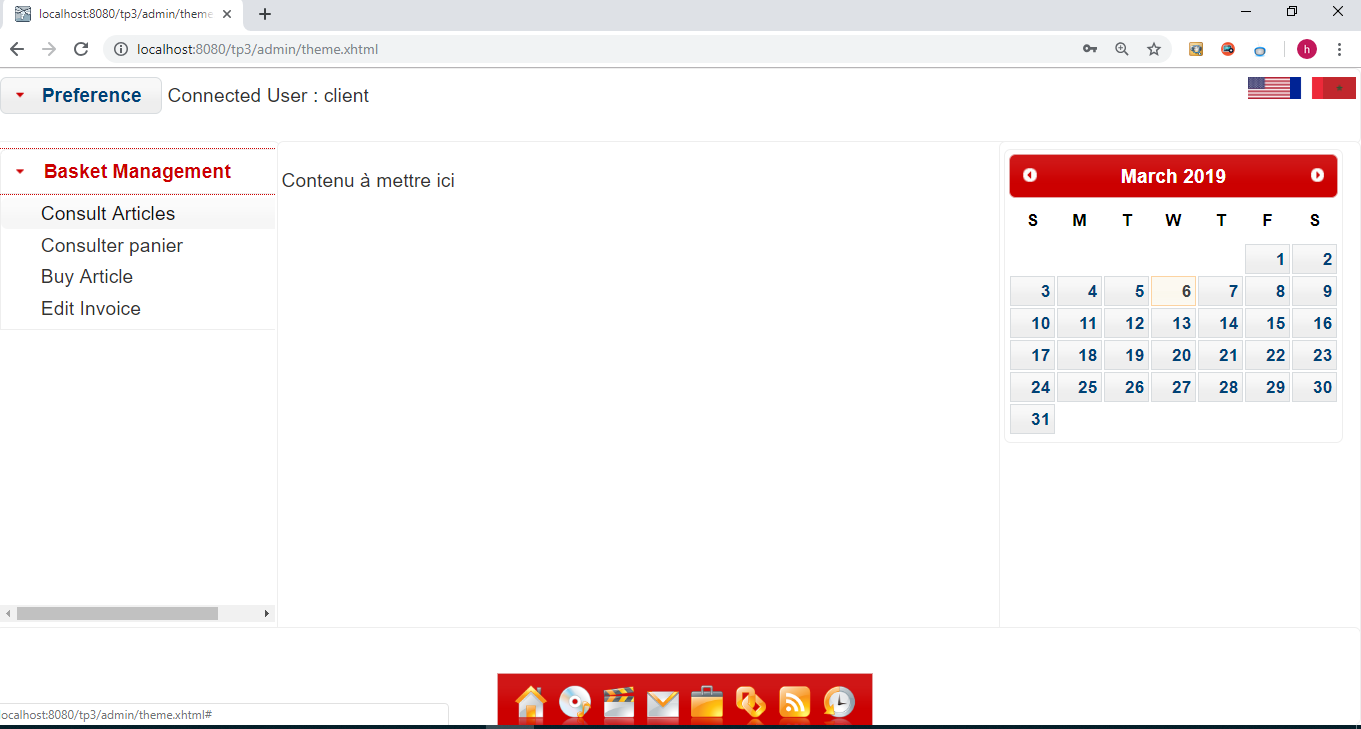
**\*** Pour se déconnecter, cliquer sur le lien suivant :



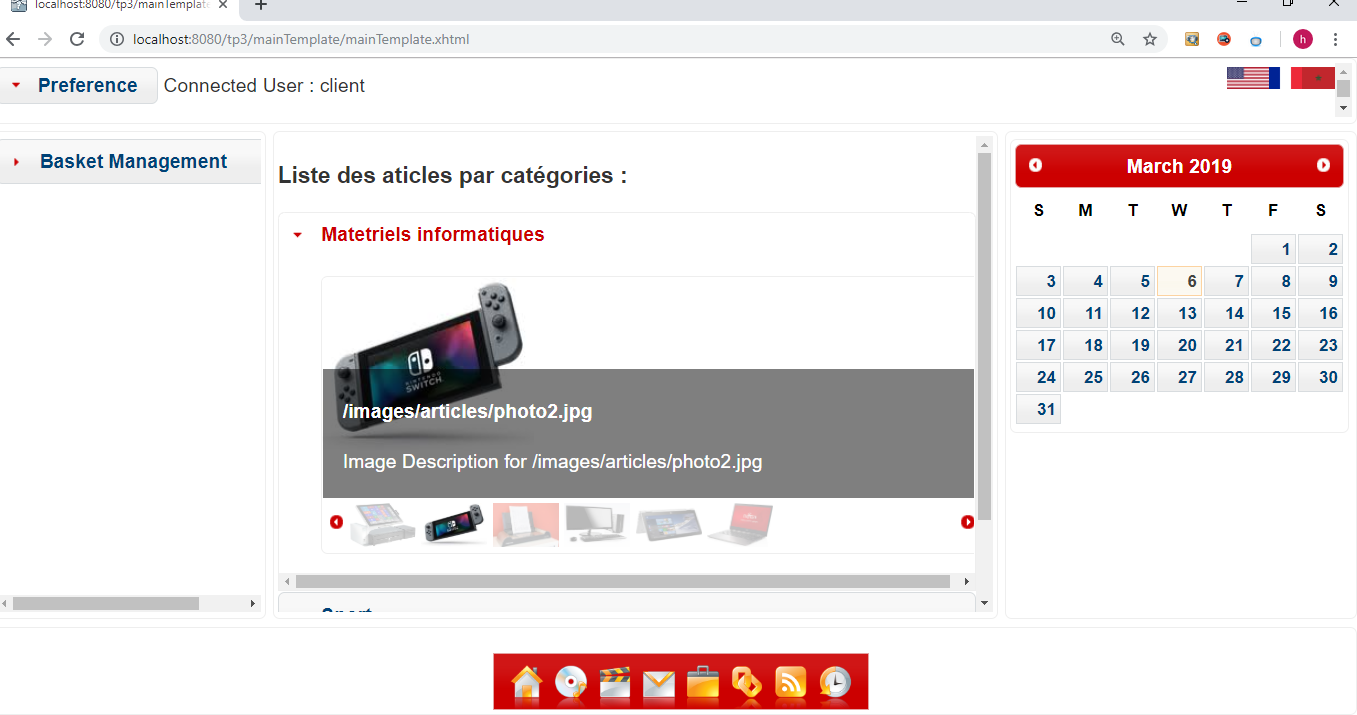
La page suivante sera affichée :



\* Connectez vous avec un compte client (clien/client par exemple). La page suivante sera affichée :



\* Pour consulter les articles, cliquer sur "Consult Articles" :



**Exercices :**

Reprendre le TP3 et implémenter les fonctionnalités suivantes :

* En tant que client (profile=CLIENT), chercher les articles par catégorie, par description et par prix.
* Ajouter les articles au panier en spécifiant la quantité à acheter de chaque article. Il faut s'assurer que la quantité demandée existe dans le stock.
* Consulter et éditer le panier.
* Editer la facture en format PDF.