ISM : Manuel de développement

Table des matières

[**1.** **Propriétés entreprises – sociétés - staff :** 2](#_Toc451370131)

[**3.1.** **Définitions** 2](#_Toc451370132)

[**3.2.** **Diagramme ESS (entreprise – société – staff)** 2](#_Toc451370133)

[**2.** **Propriétés des non conformités :** 3](#_Toc451370134)

[**3.** **PI : Extension Primefaces :** 3](#_Toc451370135)

[**3.3.** **Extension du Ruban** 3](#_Toc451370136)

[**3.3.1.** **Ajout de la barre « shortcut »** 3](#_Toc451370137)

[**3.3.2.** **Ajout des fonctionnalités de « TabView »** 4](#_Toc451370138)

[**3.3.3.** **Ajout d’un mode de groupage « ribbonGroupSet »** 4](#_Toc451370139)

# **Structure ESS (Entreprises – Sociétés – Staff)**

## **Définitions**

***Une entreprise :*** est considérer comme un groupe de société. C’est-à-dire un ensemble de sociétés pour lesquelles un ou plusieurs actionnaires veulent en avoir la maîtrise de gestion, de processus et de finance.

***Une société :*** est considérer comme telle. C’est-dire une entité de l’entreprise, une société appartenant un groupe de société.

***Un staff :*** est considérer comme un utilisateur. Ce staff en fonction de sa situation agit tant bien au niveau entrepris qu’au niveau, d’une société ou de plusieurs sociétés.

## **Diagramme ESS (entreprise – société – staff)**

Le diagramme ci-dessous représente l’architecture ESS. On peut distinguer ainsi les différentes variétés de staff. On constate également que les sociétés indépendamment des entreprises permettent de regrouper les différents staffs. Sur le diagramme,

* Les staffs en rouge ont un accès complet sur plusieurs entreprises cas de 1, 3, 6, 8.
* Ceux en orange cas de 1, 4, 5 ont un accès sur plusieurs sociétés d’une seule et même entreprise.
* Enfin, ceux en vert ont un accès à une seule société d’une seule entreprise.



Figure 1 Diagramme ESS (Entreprise Société Staff)

# **Architecture CRUD**

Afin de simplifier la prise en main de l’application la structure présente dans ce chapitre doit être utilisée. Effectivement, l’outil étant orienté base de données, il faut disposer des 4 fonctionnalités CRUD les plus connus : Créer (Create), Lecture (Read), Editer/Modifier (Update), Supprimer (Delete).

## **Diagramme CRUD**



Le diagramme ci-contre présente la structure de l’architecture CRUD à respecter.

Effectivement, chaque fonctionnalité de l’application doit être munie des possibilités suivantes :

1. Consultation de liste ou création de la fonctionnalité,
2. Une fois dans la visualisation en liste ou tableau, créer – aperçu – supprimer
3. Seule une visualisation permet d’éditer ou de supprimer
4. A chaque option, CRUD, la possibilité de valider ou d’annuler permet de retourner à la vue de la liste.

# **Structure des non conformités (NC)**

Les non conformités sont tous les constats inacceptables d’une industrie d’ordre technique, de management de propreté, …

Les NC peuvent avoir les états suivant :

|  |  |  |  |
| --- | --- | --- | --- |
| N° | Code | Désignation | Description |
| 1 | A | En cours de validation | Une fois qu’une NC a été créé, elle porte le statut e NC |
| 2 | B | En attente de solution |  |
| 3 | C | En cours | En cours  Postposé  En retard |
| 4 | D | Terminée |  |
| 5 | E | Annulée |  |
| 6 |  |  |  |

# **PI : Extension Primefaces :**

Toutes les extensions faites sur Primefaces pour l’application ISM porte le préfixe « pi ».

## **Extension du Ruban**

Afin d’améliorer les performances du ruban, le ruban de la bibliothèque Primefaces a été personnalisé.

Le but est de fournir les fonctionnalités décrites ci-dessous.

### **Ajout de la barre « shortcut »**

La barre « shortcut » est une barre de raccourci des menus, fonctionnalités les plus courantes. Le but étant de permettre une rapidité d’exécution sur certain tâche de l’application.

* + - 1. *Une zone de boutons « raccourci » :*

Elle est placée sur la gauche. Elle est composée d’un bouton d’options et une série de raccourci.

Le bouton d’options (ci-dessous en jaune) contrôle les raccourcis et les options (voir le paragraphe 1). Cette zone se présente comme le montre la figure ci-dessous (Figure 1 Ruban (barre de raccourci) - description du menu option).



Figure Ruban (barre de raccourci) - description du menu option

Le bouton des options se présente sous forme de menu déroulant comme le montre la ci-contre (Figure 2 Ruban (barre de raccourci) - description du menu options).

Les catégories d’options y sont séparées par une ligne représentée ici en rouge.

Le nombre maximal de **raccourci** est de **10**.

Le nombre maximal **d’options** est de **5**.

* + - 1. *Une zone de titre qui permet d’indiquer :*
* La version actuelle du logiciel (de préférence défini dans la configuration de l’application),
* Le nom de la société

Figure 3 Ruban (barre de raccourci) - description du menu options

* Les informations de l’utilisateur actuellement connecté,

Ceci est représenté ci-dessous (Figure 3 Ruban (barre de raccourci) - description de la zone de titre).



Figure Ruban (barre de raccourci) - description de la zone de titre

* + - 1. *Une zone d’options qui permet :*
* Choisir le thème de l’application,
* Fournir un bouton de déconnexion à l’application,
* Afficher le temps restant de connexion sur le temps totale avec un rafraichissement toutes les 30 secondes

En définitif, la barre « shortcut » est constituée de ces trois zone pour être intégrer en haut de la barre de ruban.



Figure Ruban - Composition de la barre de raccourci

Le projet étant généré par Maven, le code suivant est placé dans le pom pour considérer les ressources comme objet de compilation.

<build>

…

<resources>

<resource>

<directory>src/main/resources</directory>

<filtering>true</filtering>

</resource>

</resources>

….

</build>

Enfin placer dans le bundle ism.properties les fonctions suivante :



### **Ajout des fonctionnalités de « TabView »**

Plusieurs fonctionnalité de « primefaces…tabView » ont porté au ruban. En particulier les fonctionnalités recherchées sont :

* L’évènement « onTabChange » afin de conserver la valeur de l’index du dernier onglet activé par l’utilisateur,
* Le champ « tabIndex » afin de redémarrer le ruban avec le dernier onglet sélectionné,
* …

### **Ajout d’un mode de groupage « ribbonGroupSet »**

La fonctionnalité d’ensemble du groupe a pour objectif de facilité la disposition des boutons, menus de sélection et choix, de filtre,…

Pour cet effet, la hauteur de la **zone des groupes** du ruban (ui-ribbon-groups) est augmentée à **95 pix**els. Ceci permet d’augmenter la zone de contenu des groupes (ui-ribbon-group-content). Pour obtenir cette augmentation, la zone de label (ui-ribbon-group-label) conserve sa propriété de hauteur.

Les fonctionnalités proposées par le « ribbonGroupSet » sont divisé en 2 catégories. La catégorie normale (sans préfixe) et la catégorie miniature (avec préfixe « tiny »). La version « normale » se différencie de la version « tiny » par la visibilité ou non de la désignation ou de l’icône uniquement. Le tableau suivant reprend leur description.

|  |  |  |
| --- | --- | --- |
| Nom | Set | Style |
| RIBBON\_SET\_BIG\_BUTTON | big | ui-ribbonset-big |
| RIBBON\_SET\_MID\_BUTTON | middle | ui-ribbonset-mid |
| RIBBON\_SET\_SMALL\_BUTTON | small | ui-ribbonset-small |
|  |  |  |
| RIBBON\_SET\_TINY\_BIG\_BUTTON | tiny-big | ui-ribbonset-tinybig |
| RIBBON\_SET\_TINY\_MID\_BUTTON | tiny-middle | ui-ribbonset-tinymid |
| RIBBON\_SET\_TINY\_SMALL\_BUTTON | tiny-small | ui-ribbonset-tinysmall |

* + - 1. ***Le set « big » (RIBBON\_SET\_BIG\_BUTTON)***

Le set « big » permet d’afficher des boutons en grand format. Dans ce format, on peut mettre au **maximum un bouton sur la hauteur** du ruban. Les boutons sont constitués d’une zone d’icône et une zone de t**exte sur deux lignes en dessous**. La figure ci-dessous (Figure 5 Ruban set "big").



Figure Ruban set "big"

Il faut noter le sens des boutons qui vont de gauche à droite selon l’ordre de déclaration

* + - 1. ***Le set « middle » (RIBBON\_SET\_MID\_BUTTON)***

Le set « middle » permet d’afficher des boutons en format intermédiaire. Dans ce format, on peut mettre au **maximum deux boutons sur la hauteur** du ruban. Les boutons sont constitués d’une zone d’icone sur la gauche et une zone de **texte sur deux lignes** à leur droite.



Figure Ruban set "middle"

Il faut noter le sens des boutons qui vont de haut en bas sur une hauteur puis vers la droite. Si le nombre de bouton est impair comme le montre la figure, cela ne pose aucun problème car le set s’arrête en cet endroit.

* + - 1. ***Le set « small » (RIBBON\_SET\_SMALL\_BUTTON)***

Le set « small » permet d’afficher des boutons en format diminué ou petit. Dans ce format, on peut mettre au **maximum 3 boutons sur la hauteur** du ruban. Les boutons sont constitués d’une zone d’icône sur la gauche et d’une zone de **texte sur** **1 ligne** à leur droite.



Figure Ruban set "small"

Il faut noter le sens des boutons qui vont de haut en bas sur une hauteur puis vers la droite. Si le nombre de bouton est impair comme le montre la figure, cela ne pose aucun problème car le set s’arrête en cet endroit.

* + - 1. ***Le set « tiny-big » (RIBBON\_SET\_TINY\_BIG\_BUTTON)***

Le set « tiny-big » est l’équivalent du big dont la largeur a été divisée par 2 ! La zone de texte a été conservé mais ne peux plus être importante.



Figure Rub set "tiny-big"

* + - 1. ***Le set « tiny-middle » (RIBBON\_SET\_TINY\_MID\_BUTTON)***

Le set « tiny-middle » est l’équivalent du « middle » dont la largeur a été réduite pour ne conserver que l’icône. Par conséquent, la zone de texte a disparu.



Figure Ruban "tiny-middle"

* + - 1. ***Le set « tiny-small » (RIBBON\_SET\_TINY\_SMALL\_BUTTON)***

Le set « tiny-small » est l’équivalent du « small » dont la largeur a été réduite pour n’en conserver que l’icône. Par conséquent, la zone de texte a disparu.



Figure Ruban set "tiny-small"