

2023/2024	Projet JAVA – Etape 08
BTS SIO	Auteurs : Loric Worms / Stefen Rialland
SIOB	Date de rédaction : 17/04/2024

Compte rendu étape n°8 fonctionnalité d'ajout d'un nouveau salarié

Création du bouton ajouter. Afin de créer un nouveau salarié.

Code lors de l'action sur le bouton depuis JFrameListeSalaries.java :

Il récupère le service sélectionné de la liste déroulante, afin de l'utiliser en non-éditable sur la page d'ajout.

```
private void jButtonAjouterActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    // Récupérer le service sélectionné
    Service serviceSelectionne = getServiceSelectionne();

    try {
        // Récupérer toutes les catégories de la base de données
        ArrayList<Categorie> desCategories = DaoCategorie.getAll();

        // Créer et afficher la boîte de dialogue avec la liste des catégories
        this.toBack();
        JDialogAjouter ajouterDialog = new JDialogAjouter(aThis: this, par:true, serviceSelectionne, desCategories);
        ajouterDialog.setVisible(b: true);
        ajouterDialog.toFront();
    } catch (SQLException | IOException ex) {
        // Gérer les erreurs de récupération des catégories depuis la base de données
        ex.printStackTrace(); // À remplacer par une gestion d'erreur appropriée
    }
}

public Service getServiceSelectionne() {
    return (Service) jComboBoxLesServices.getSelectedItem();
}
```

Création de la liste déroulante des catégories et du service sélectionné. Et donc affichage dans un jTextField son contenu (désignation).

Et enfin appel de la classe de remplissage de la table déroulante.

```
public class JDialogAjouter extends javax.swing.JDialog {

    private DefaultComboBoxModel modeleJComboLesCategories;
    private Service serviceSelectionne;

    /**
     * Creates new form JDialogConsulter
     */
    public JDialogAjouter(java.awt.Frame parent, boolean modal) {
        super(owner: parent, modal);
        initComponents();
    }

    public JDialogAjouter(JFrameListeSalaries aThis, boolean par, Service serviceSelectionne, List<Categorie> desCategories) {
        initComponents();
        this.setLocation(x: 150, y: 150);

        modeleJComboLesCategories = new DefaultComboBoxModel();
        jComboBoxCate.setModel(aModel: modeleJComboLesCategories);
        this.serviceSelectionne = serviceSelectionne;

        if (serviceSelectionne != null) {
            jTextFieldService.setText(v: serviceSelectionne.getDesignation());
        }

        remplirJComBoxCate(desCategories);
    }
}
```

Création de tous les accesseurs afin de récupérer les données entrées par l'utilisateur pour les

2023/2024	Projet JAVA – Etape 08
BTS SIO	Auteurs : Loric Worms / Stefen Rialland
SIOB	Date de rédaction : 17/04/2024

utiliser dans l'appel de création d'un salarié et appel du Dao afin de créer le salarié.

```
private void jButtonAjouter2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    String code = jTextFieldCode.getText();
    String nom = jTextFieldNom.getText();
    String prenom = jTextFieldPrenom.getText();

    Date utilDateNaiss = jDateChooserNaiss.getDate();
    java.sql.Date naiss = new java.sql.Date(date: utilDateNaiss.getTime());

    Date utilDateEmbauche = jDateChooserEmbauche.getDate();
    java.sql.Date embauche = new java.sql.Date(date: utilDateEmbauche.getTime());

    String fonction = jTextFieldFonction.getText();

    String tauxHoraire = jTextFieldTauxHoraire.getText();
    double horaire = Double.parseDouble(s: tauxHoraire);

    String famille = jTextFieldSitua.getText();

    String nbrEnfants = jTextFieldEnfants.getText();
    int enfants = Integer.parseInt(s: nbrEnfants);

    int service = serviceSelectionne.getCode();

    String libelleCategorie = (String) jComboBoxCate.getSelectedItem();
    String categorie = getNumCategorie(libelleCategorie);

    try {
        DaoSalarie.ajouterSalarie(code,nom,prenom,naiss,embauche,fonction,horaire,famille,enfants,categorie,service);
    } catch (SQLException | IOException ex) {
        Logger.getLogger(name: JDialogAjouter.class.getName()).log(level: Level.SEVERE, msg:null, thrown: ex);
    }
}
```

Et donc la classe de remplissage de la liste déroulante des catégories et récupération du code de la catégorie.

```
public void remplirJComBoxCate(List<Categorie> desCategories) {
    modeleJComboLesCategories.removeAllElements(); // Efface d'abord tous les éléments de la combobox
    // Ajoute ensuite les libellés des catégories à la combobox
    for (Categorie uneCategorie : desCategories) {
        modeleJComboLesCategories.addElement(anObject: uneCategorie.getLibelle());
    }
}

public String getNumCategorie(String libelleCategorie) {
    // Parcourir la liste des catégories pour trouver celle correspondant au libellé
    for (Categorie uneCategorie : desCategories) {
        if (uneCategorie.getLibelle().equals(anObject: libelleCategorie)) {
            return uneCategorie.getCode();
        }
    }
    return null;
}
```

Rendu visuel de l'interface de l'ajout d'un salarié depuis l'éditeur.


2023/2024	Projet JAVA – Etape 08
BTS SIO	Auteurs : Loric Worms / Stefen Rialland
SIOB	Date de rédaction : 17/04/2024


Ajouter un nouveau salarié :

Code :

Nom :

Prenom :

Date naissance : 

Date embauche : 

Fonction :

Taux horaire :

Situation familiale :

Nbr enfants :

Service :

Categorie : Item 1 ▾


Enfin voilà le rendu une fois le projet lancé : L'utilisateur peut remplir les cases de texte vides. Le service est automatiquement récupéré selon celui sélectionné précédemment, et le choix de la catégorie se fait sous la forme d'une liste déroulante.


Ajouter un nouveau salarié :

Code :

Nom :

Prenom :

Date naissance : 

Date embauche : 

Fonction :

Taux horaire :

Situation familiale :

Nbr enfants :

Service :

Categorie : Cadre moyen ▾

Cadre moyen
Cadre supérieur
Employé niveau 1
Employé niveau 2

2023/2024	Projet JAVA – Etape 08
BTS SIO	Auteurs : Loric Worms / Stefen Rialland
SIOB	Date de rédaction : 17/04/2024

L'ajout d'un salarié est donc enfin un succès une fois le bouton cliqué. Le salarié apparaît bien dans la liste de tous les salariés sur la fenêtre principale.

Cependant, il semble y avoir un souci car il nous est impossible de le supprimer :

