CR du TP 5 : TEST IHM	
Sylvie Auneau et Nicolas Evano	date 28 octobre 2011

Test d'IHM avec WindowTester-Pro

1)Mise en oeuvre de windows tester pro:

ajout du plug-in eclipse windows tester pro (sous l'environnement eclipse):
help>Intall New Software
compléter le TextField "Work with" avec L'URL suivante:

http://dl.google.com/eclipse/inst/windowtester/latest/3.6

select all
clic sur "finish" bouton.
eclipse télécharge le "plug-in" puis redémarre.
quand eclipse redemarre un nouveau type de lanceur est disponible si le plugin a été correctement installé.

2)Test d'un carnet d'addresse :

l'application carnet d'adresse nous permet de gérer des contacts via une IHM Swing.

CR du TP 5 : TEST IHM	
Sylvie Auneau et Nicolas Evano	date 28 octobre 2011

-test de l'IHM carnet d'adresse avec windowTester-Pro

2.1 L'ensemble des textField permettant d'ajouter un contact doivent être désactivés au démarrage de l'application.

La fonction record de window-test pro nous permet de générer automatiquement un Junit test cependant ce test doit être repris pour pouvoir être fonctionnel.
En effet il nous a fallu rajouter la ligne suivante pour que le test soit fonctionnel:

```
ui.assertThat(new LabeledTextLocator("ZIP").isEnabled( false ) );
```

le test généré est: `StartUpTextFieldDesabled`

CR du TP 5 : TEST IHM	
Sylvie Auneau et Nicolas Evano	date 28 octobre 2011

2.2 L'ensemble des champs permettant d'ajouter un contact doivent être activés aprés un clic sur le bouton "new"

La fonction record de window-test pro nous permet de générer automatiquement un Junit test cependant ce test doit être repris pour pouvoir être fonctionnel.
En effet il nous a fallu rajouter la ligne suivante pour que le test soit fonctionnel:

```
ui.assertThat(new LabeledTextLocator("ZIP").isEnabled( false ) );
```

le test généré est: `AllTestFieldsEnabledAfterClickOnNewButton.`

2.3 Le bouton "Detete" doit être activer uniquement si un contact est sélectionné dans la list

Modification du code pour rendre la "JList" utilisée exploitable avec "windowTester-Pro" ajout de la ligne suivante dans la méthode *initComponent* de la classe *AddressListPanel*:

```
addressList.setName("JListAddress");
```

Ajout d'un getter et d'un setter sur le membre `btnSaveAddress` de la classe `AddressActionPanel`
désactivation du bouton "*Delete*" dans la méthode `initComponent` de la classe `AddressActionPanel`.
Ajout des lignes suivantes dans la méthode `valueChanged` de la classe `AddresseFrame`:
-Dans le corps du if:

```
addressActionPanel.getBtnDeleteAddress().setEnabled( true );
```

-Dans le corps du else:

```
addressActionPanel.getBtnDeleteAddress().setEnabled( false );
```

Modification du test généré par windowTester-Pro pour le rendre fonctionnel avec l'ajout de la ligne suivante:

```
ui.click( new JListLocator("Sylvie, Auneau", new
NamedWidgetLocator( "JListAddress" ) ) );
```

le test généré est: `TestEnabledDesabledDeleteButton`

CR du TP 5 : TEST IHM	
Sylvie Auneau et Nicolas Evano	date 28 octobre 2011

2.4 Le bouton “Save” doit être activé uniquement aprés un clic sur le bouton “New”

Génération du cas de test avec windowTester-Pro:
`SaveButtonOnlyAvailableIfNewButtonClickedBefore`
Le test est non passant.
Modification code pour corriger le probléme:
Ajout d’un getter et d’un setter sur le membre privé `btnSaveAddress` de la classe `AddressActionPanel`
désactivation du bouton “*Delete*” dans la méthode `initComponent` de la classe `AddressActionPanel`.
Ajout de la ligne suivante en fin du corps de `newAddress` de la classe `AddressFrame`:

```
addressActionPanel.getBtnSaveAddress().setEnabled( true );
```

Ajout de la ligne suivante en fin du corps de `saveAddress` de la classe `AddressFrame`:

```
addressActionPanel.getBtnSaveAddress().setEnabled( false );
```

réexecution du test pour valider le correctif, le test est passant.

2.5 vérifier que le bouton Delete est désactivé quand aucun élément de la JList de contact n’est sélectionné.

Le test généré précédemment en 2.3 valide cette condition.
Le correctif appliqué en 2.3 sur l’application carnet d’addresse valide cette condition.

2.6 vérifier que le bouton Save est activé quand le champ d’un contact de la JLlist est modifié .

Génération avec windowTester-Pro du cas de test suivant:
`CheckSaveButtonEnalbedAfterListElementSelectedAndEditButtonClicked`
Le Test est non passant.
Modification de la méthode `editAddress` dans la classe `AddressFrame` pour corriger le comportement de l’application.
ré-execution du test, le test est passant.

2.7 vérifier que les boutons “Delete” and “Save” sont désactivés quand on appuie sur le bouton “Cancel” aprés avoir sélectionné un élément de la JList et cliqué sur le bouton “Edit”:

Le Test est non passant.
Modification de la méthode `cancelAddress` dans la classe `AddressFrame` pour corriger le comportement de l’application.
ré-execution du test, le test est passant.