

Arthur Murillo TD2 | TP4

**Compte rendu IHM – TP5**

***Questions de TD :***

***Partie A Programme deux.exe : Confirmation de fermeture***

1. L’évènement à intercepter est wxEVT\_CLOSE\_WINDOW
   1. Le gestionnaire d’évènements est déclaré dans le fichier .h de la classe Principale, dans la déclaration de la classe.
   2. Le gestionnaire d’évènements est défini dans le fichier .cpp de la classe comme toute autre méthode.
   3. Pour le corps de la procédure, on a :

Void Principale2 ::siQuitter(wxCloseEvent& evenement /\* On intercepte l’évènement\*/) {

/\*On fait apparaitre la boite de dialogue, qui attendra le choix de l’utilisateur. Answer stockera l’identifiant du bouton (oui/non) cliqué pour qu’on décide de l’action à suivre en fonction du bouton\*/

Int answer = MessageBox (« Quitter le programme ?», « Confirmation de fermeture », wxYES\_NO) ;

/\* Si l’identifiant dans answer correspond à celui de wxYES, on détruit la fenêtre, sinon on ne fait rien (la boite de dialogue disparait cependant) et la fenêtre principale reste ouverte\*/

If (answer == wxYES)

{

Destroy() ; // Détruire la fenetre

}

}

***Partie B – un.exe***

1. Dictionnaires des états et évènements du système :

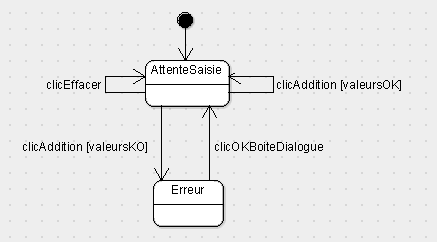
***ETATS :***

|  |  |
| --- | --- |
| **Nom** | **Signification** |
| En attente de Saisie | La fenêtre est en attente de la saisie des nombres et d’une action sur l’un des deux boutons |
| Erreur | La saisie n’est pas correcte (opérande manquante/ erronée), la boite de dialogue s’enclenche |

***EVENEMENTS :***

|  |  |
| --- | --- |
| **Nom** | **Signification** |
| clicAddition | Clic sur le bouton « Addition ! » pour actionner la tentative d’addition des opérandes saisies |
| clicEffacer | Clic sur le bouton « Effacer ! » pour actionner l’effacement des opérandes saisies |
| clicOKBoiteDialogue | Clic sur le bouton « OK »de la boite de dialogue apparue en cas d’échec de l’addition |

1. Diagramme état transition :



1. T\_etatsEvenements

|  |  |  |  |
| --- | --- | --- | --- |
| **nomEtat/ Evenement** | Clic Addition | Clic Effacer | clicOKBoiteDialogue |
| attenteSaisie | Si bon, affichage résultat, sinon transition sur erreurSaisie | Pas de changement | / |
| erreurSaisie | / | / | Transition sur état attenteSaisie |

1. Clic Addition -> wxCommandEvent géré par le gestionnaire d’évènements onClicAddition par ex.

Clic Effacer -> wxCommandEvent géré par le gestionnaire d’évènements onClicEffacer par ex.

clicOKBoiteDialogue -> wxCommandEvent géré par la boite de dialogue dans onClicAddition en cas de mauvaises valeurs par ex.

1. T\_etatsEvenements 2.0

|  |  |  |  |
| --- | --- | --- | --- |
| **Elément d’interface** | Bouton « Addition ! » | Bouton « Effacer ! » | Bouton « OK » sur boîte de dialogue |
| **nomEtat/ Evènement** | Clic Addition | Clic Effacer | clicOKBoiteDialogue |
| attenteSaisie | Si bon, affichage résultat, sinon transition sur erreurSaisie | Pas de changement | / |
| erreurSaisie | / | / | Transition sur état attenteSaisie |