Devoir n°2 – Projet Traders

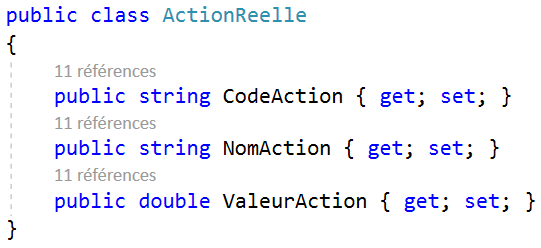
**Enoncé.**

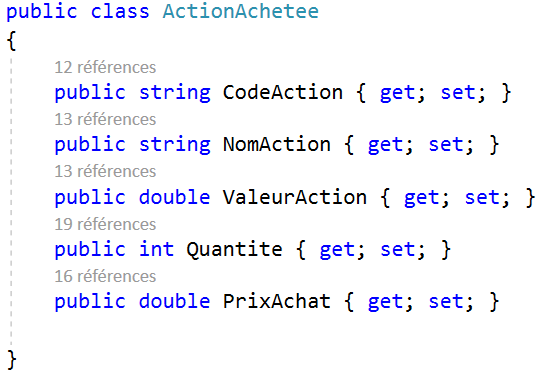
On souhaite gérer dans cette application le portefeuille d’actions de chaque trader.

Pour cela, nous utiliserons 2 collections :

1. Un dictionnaire dans lequel on stocke en clé le nom du trader et en « value » la liste des actions qu’il possède : ActionAchetee
2. Une liste de toutes les actions : ActionReelle

La définition de chaque classe vous est fournie ci-dessous.





A chaque instant, l’utilisateur va pouvoir vendre ou acheter des actions pour un trader.

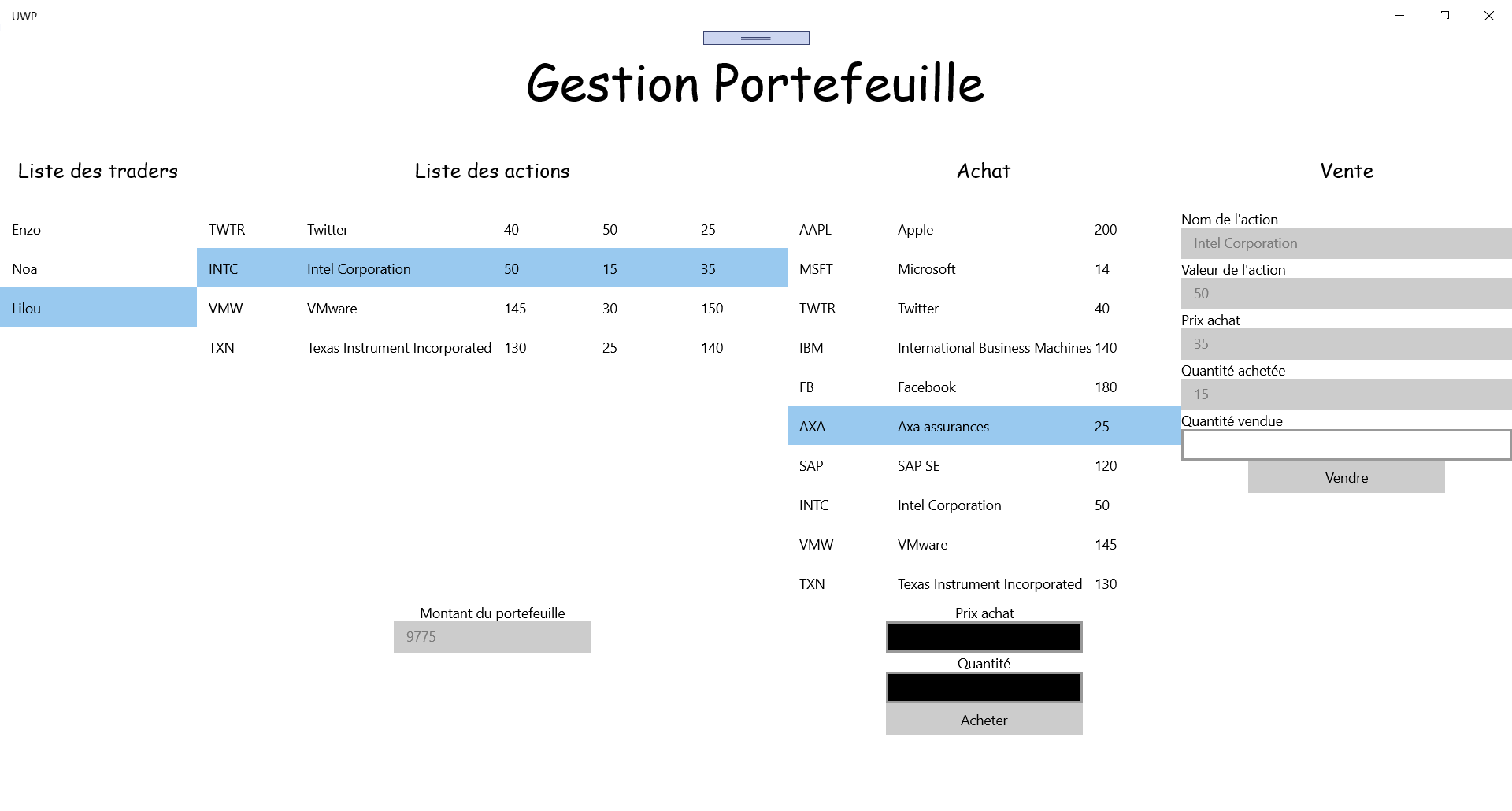
Lors d’un achat 🡺 on calculera le nouveau montant de son portefeuille = somme(quantite \* prix achat)

Lors d’une vente 🡺

1. On vérifiera que la quantité vendue ne dépasse pas la quantité possédée = quantité achetée
2. Si la quantité vendue est égale à la quantité achetée 🡺 on recalcule le montant du portefeuille et on supprime l’action car en ayant TOUT vendu, le trader ne possède plus l’action en question.
3. Si la quantité vendue est inférieur à la quantité achetée 🡺 on recalcule le montant du portefeuille et on diminue la quantité de l’action en question.

Voici l’interface principale 🡺

TextBlock



ListView

TextBox

TextBox

Button

Button

TextBox

TextBox

TextBox

TextBlock

TextBlock

TextBlock

TextBlock

ListView

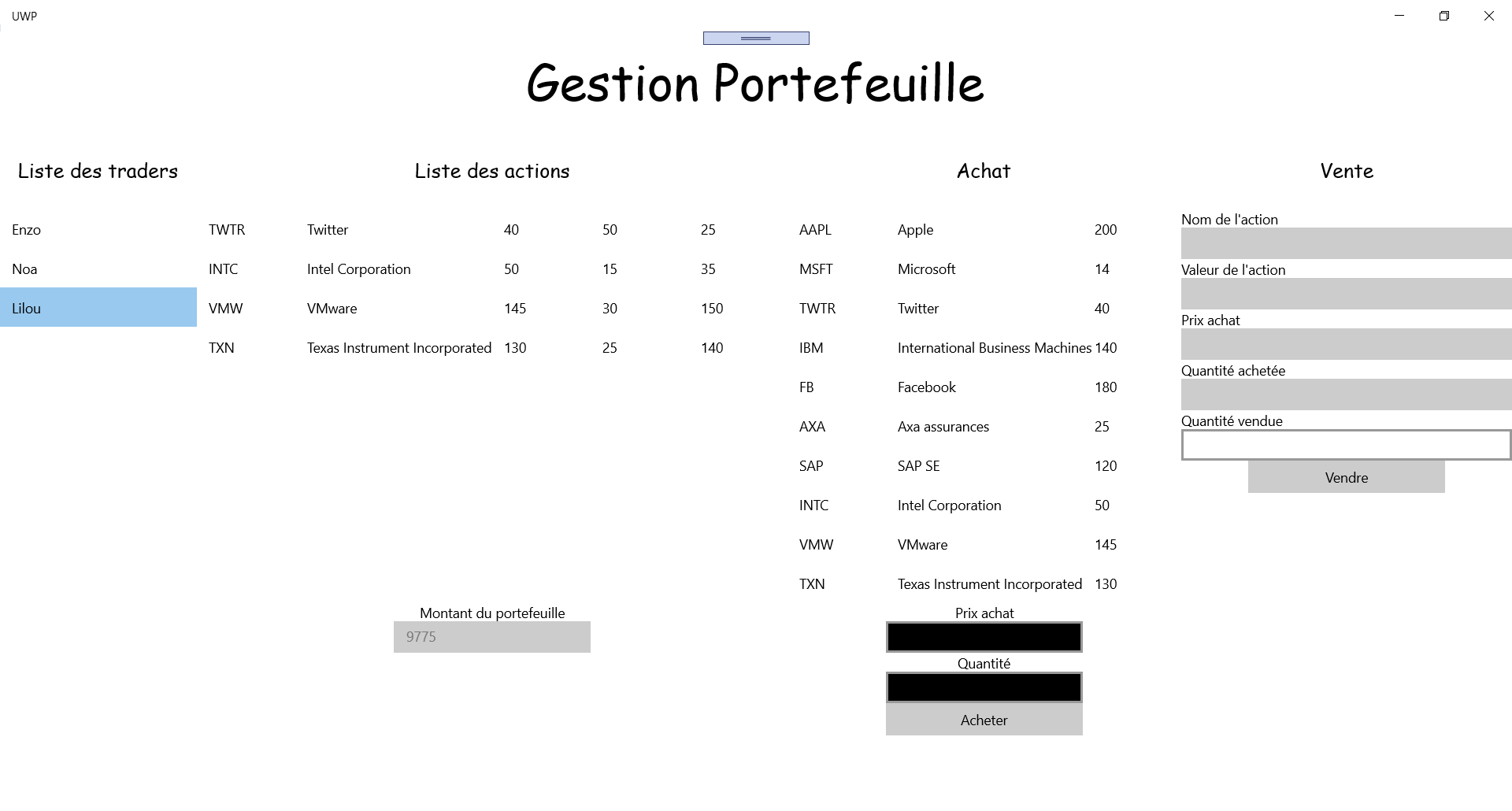
ListView

Au chargement de la page 🡺

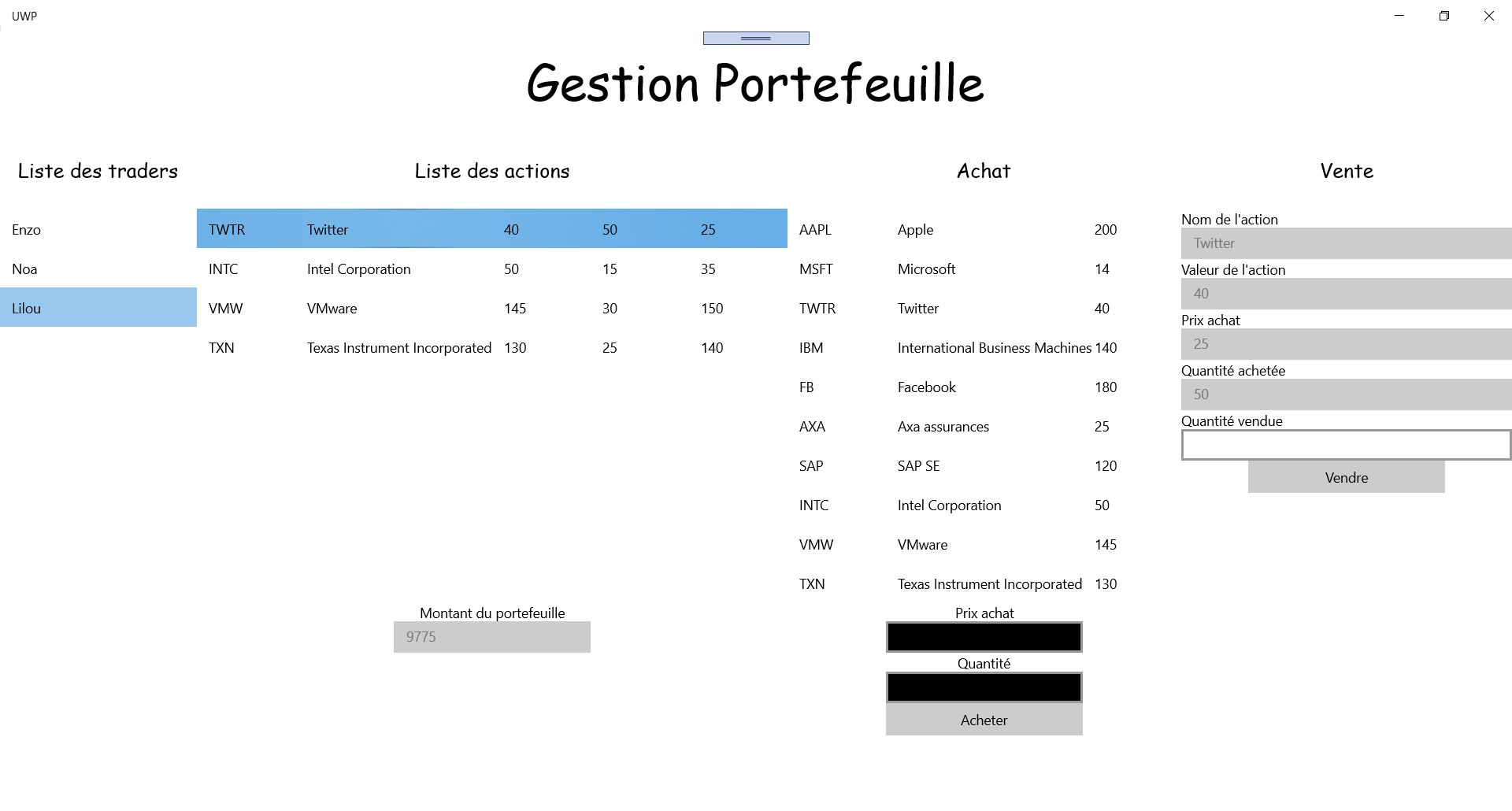
On affiche uniquement la liste des traders et la liste de toutes les actions (ActionRelle)



Lors de la sélection d’un trader, on affiche la liste de ses actions. Celles qu’il possède (ActionAchetee) et on calcule le montant de son portefeuille.



Lors de la sélection d’une action, on affiche les informations correspondantes dans les différents contrôles graphiques.

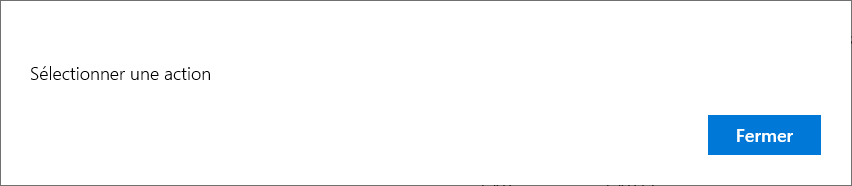


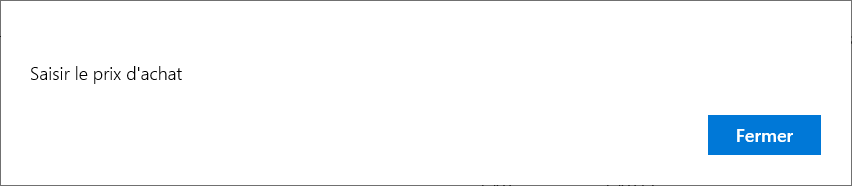
A partir de là, l’utilisateur peut vendre ou acheter.

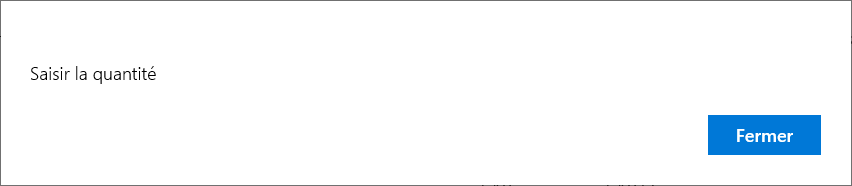
# ACHAT

On peut acheter une nouvelle action : c’est-à-dire une action que l’on ne possède pas déjà. Mais on peut aussi acheter une action que l’on a déjà. Dans ce cas, le prix d’achat et la quantité achetée pourront être différents.

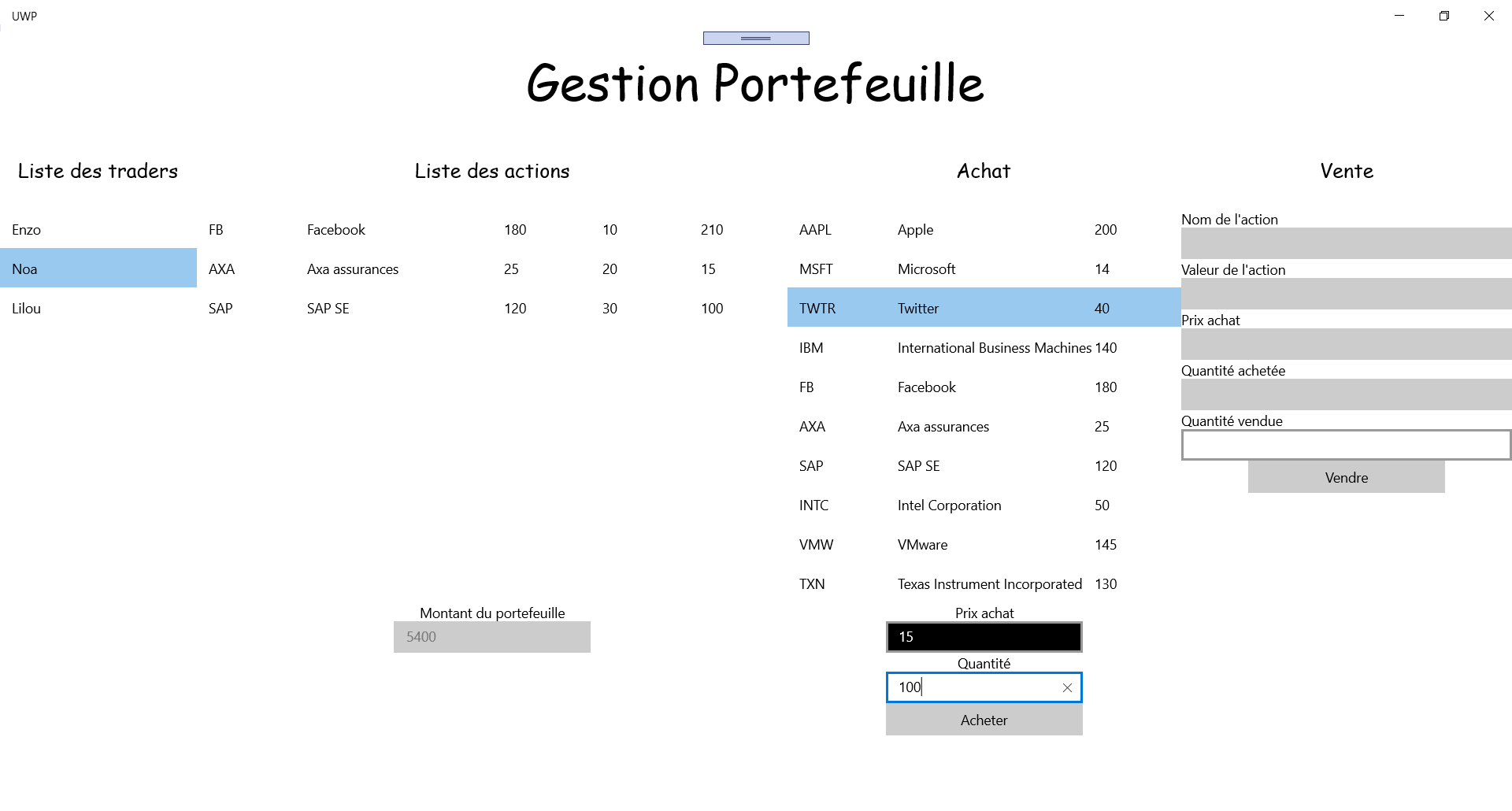
Lors d’un clic sur le bouton « Acheter », on vérifie quelques saisies



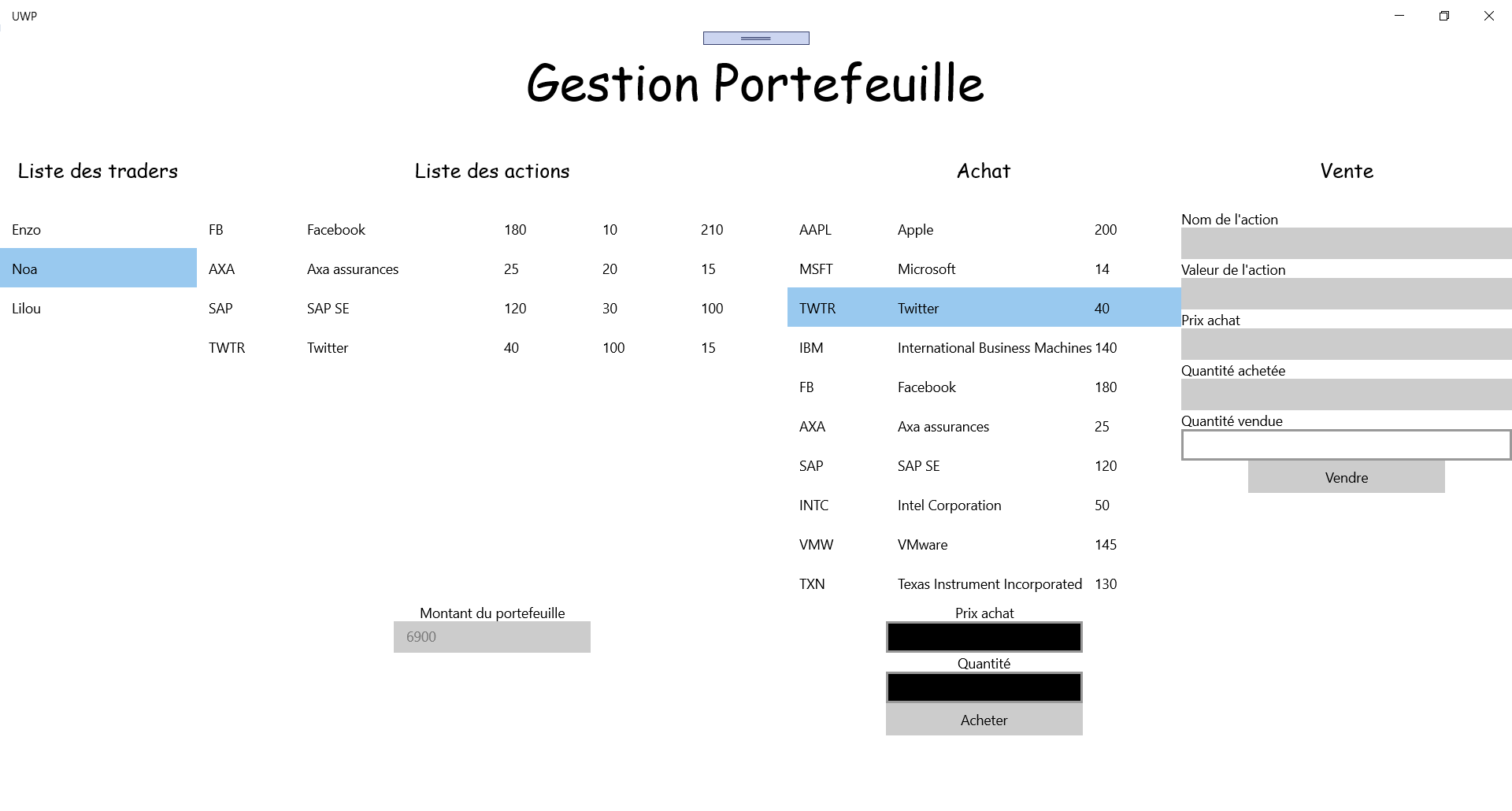




Exemple de l’achat d’une nouvelle action 🡺

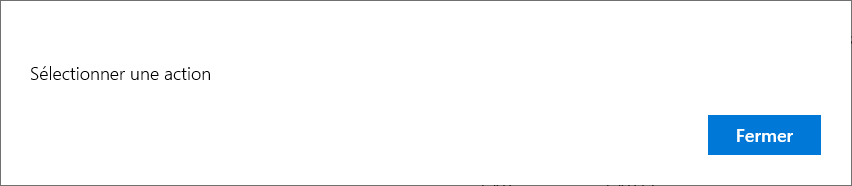


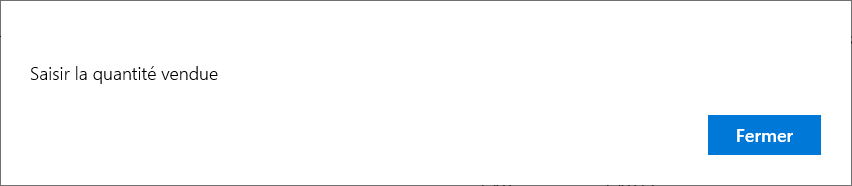
Lors d’un clic sur le bouton « Acheter »

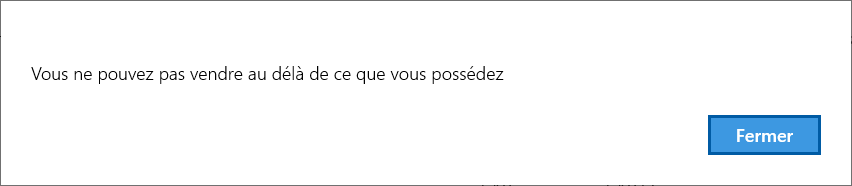


# VENTE

Lors de la vente d’une action il faudra vérifier quelques informations (voir règles de gestion au début du document).



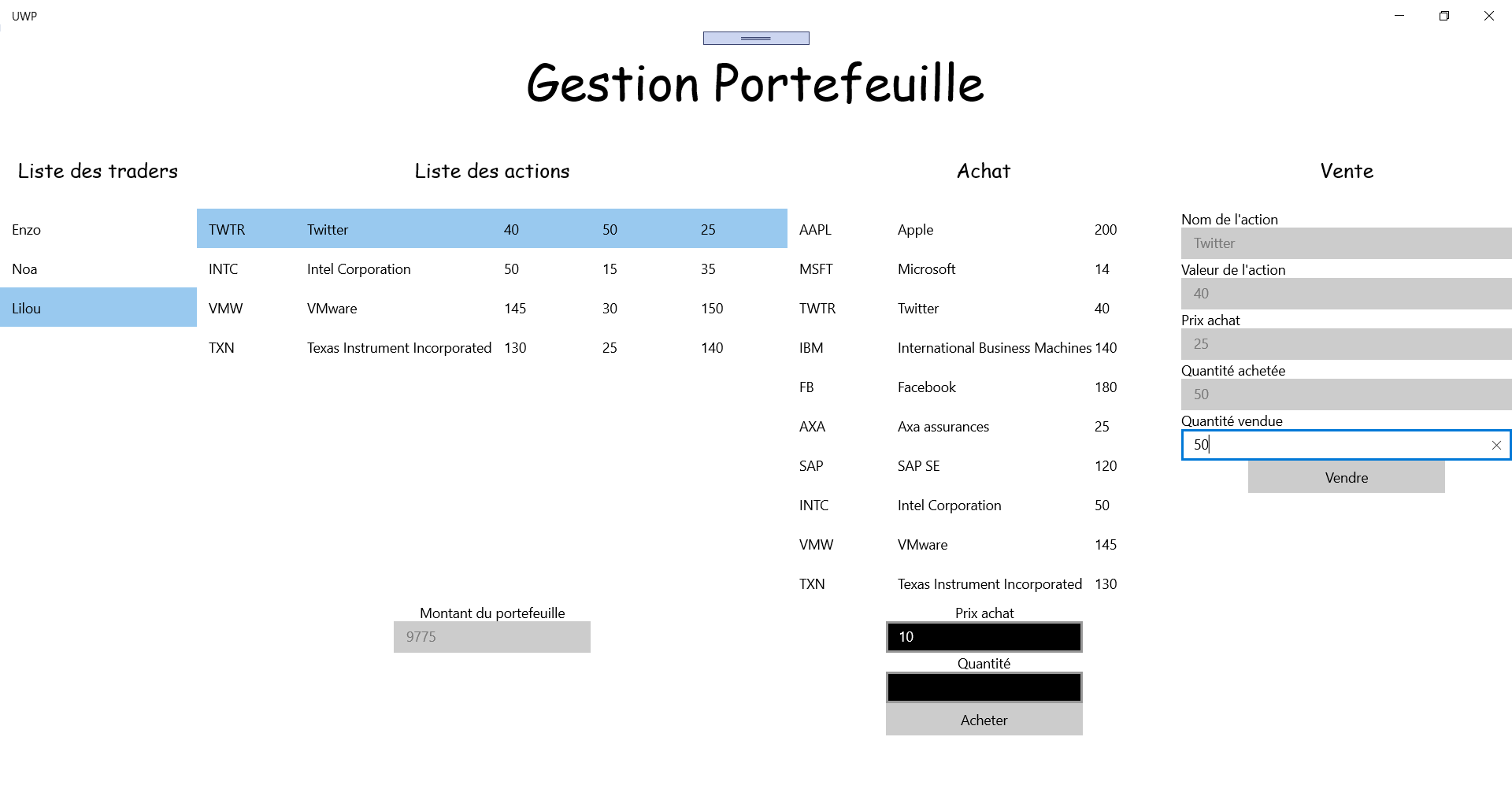




Si l’utilisateur décide de TOUT vendre 🡺

1. On supprime l’action
2. On recalcule le montant du portefeuille

Exemple



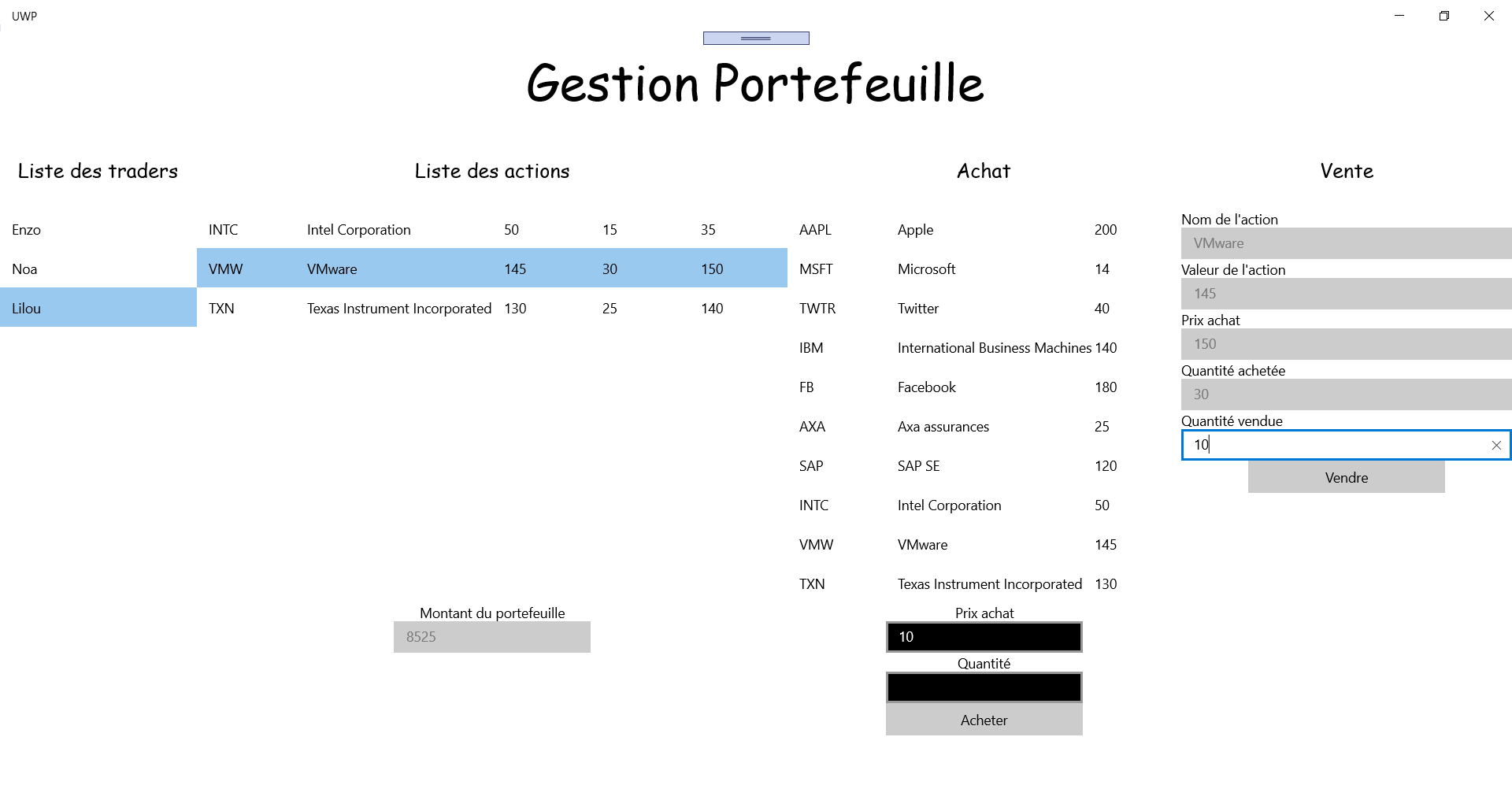


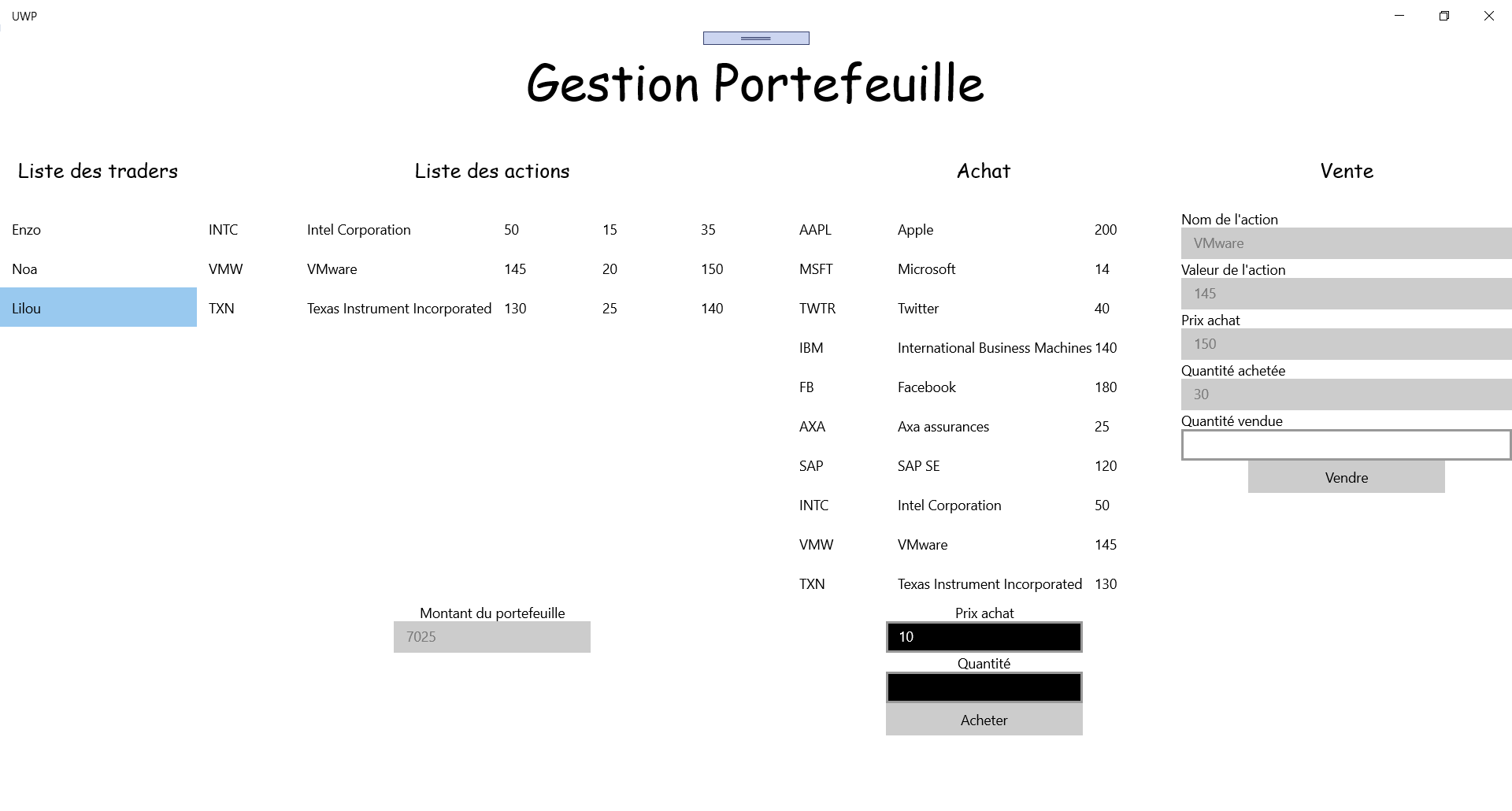
L’action est supprimée

Si l’utilisateur décide de vendre une partie 🡺

1. On diminue la quantité de l’action
2. On recalcule le montant du portefeuille

Exemple





# Jeu d’essais

## Pour toutes les actions

#region jeu d'essais pour la liste de toutes les actions réelles

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "AAPL",

NomAction = "Apple",

ValeurAction = 200

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "MSFT",

NomAction = "Microsoft",

ValeurAction = 14

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "TWTR",

NomAction = "Twitter",

ValeurAction = 40

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "IBM",

NomAction = "International Business Machines",

ValeurAction = 140

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "FB",

NomAction = "Facebook",

ValeurAction = 180

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "AXA",

NomAction = "Axa assurances",

ValeurAction = 25

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "SAP",

NomAction = "SAP SE",

ValeurAction = 120

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "INTC",

NomAction = "Intel Corporation",

ValeurAction = 50

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "VMW",

NomAction = "VMware",

ValeurAction = 145

}

);

lesActionsReelles.Add

(

new ActionReelle()

{

CodeAction = "TXN",

NomAction = "Texas Instrument Incorporated",

ValeurAction = 130

}

);

#endregion

## Pour le dico

#region jeu d'essais pour le dico

dico.Add

("Enzo", new List<ActionAchetee>()

{

new ActionAchetee()

{

CodeAction = "AAPL",

NomAction = "Apple",

ValeurAction = 200,

PrixAchat = 210,

Quantite = 10

},

new ActionAchetee()

{

CodeAction = "MSFT",

NomAction = "Microsoft",

ValeurAction = 140,

PrixAchat = 100,

Quantite = 50

},

new ActionAchetee()

{

CodeAction = "TWTR",

NomAction = "Twitter",

ValeurAction = 40,

PrixAchat = 35,

Quantite = 20

},

new ActionAchetee()

{

CodeAction = "IBM",

NomAction = "International Business Machines",

ValeurAction = 140,

PrixAchat = 110,

Quantite = 40

}

}

);

dico.Add

("Noa", new List<ActionAchetee>()

{

new ActionAchetee()

{

CodeAction = "FB",

NomAction = "Facebook",

ValeurAction = 180,

PrixAchat = 210,

Quantite = 10

},

new ActionAchetee()

{

CodeAction = "AXA",

NomAction = "Axa assurances",

ValeurAction = 25,

PrixAchat = 15,

Quantite = 20

},

new ActionAchetee()

{

CodeAction = "SAP",

NomAction = "SAP SE",

ValeurAction = 120,

PrixAchat = 100,

Quantite = 30

}

}

);

dico.Add

("Lilou", new List<ActionAchetee>()

{

new ActionAchetee()

{

CodeAction = "TWTR",

NomAction = "Twitter",

ValeurAction = 40,

PrixAchat = 25,

Quantite = 50

},

new ActionAchetee()

{

CodeAction = "INTC",

NomAction = "Intel Corporation",

ValeurAction = 50,

PrixAchat = 35,

Quantite = 15

},

new ActionAchetee()

{

CodeAction = "VMW",

NomAction = "VMware",

ValeurAction = 145,

PrixAchat = 150,

Quantite = 30

},

new ActionAchetee()

{

CodeAction = "TXN",

NomAction = "Texas Instrument Incorporated",

ValeurAction = 130,

PrixAchat = 140,

Quantite = 25

}

}

);

#endregion