DUT Informatique Paris Descartes

Antoine WANG

MASTERMIND

Visual Basic

Antoine WANG

Table des matières

[Présentation de l’application 2](#_Toc451963279)

[Schémas d’ordonnancement des feuilles 3](#_Toc451963280)

[Feuilles et Codes 4](#_Toc451963281)

[Fenêtre accueil avec son code 4](#_Toc451963282)

[Fenêtre pattern avec son code 7](#_Toc451963283)

[Fenêtre de jeu avec son code 8](#_Toc451963284)

[Fenêtre des statistiques avec son code 15](#_Toc451963285)

[Code du module 17](#_Toc451963286)

[Exemple d’inscription 19](#_Toc451963287)

[Conclusion 20](#_Toc451963288)

# Présentation de l’application

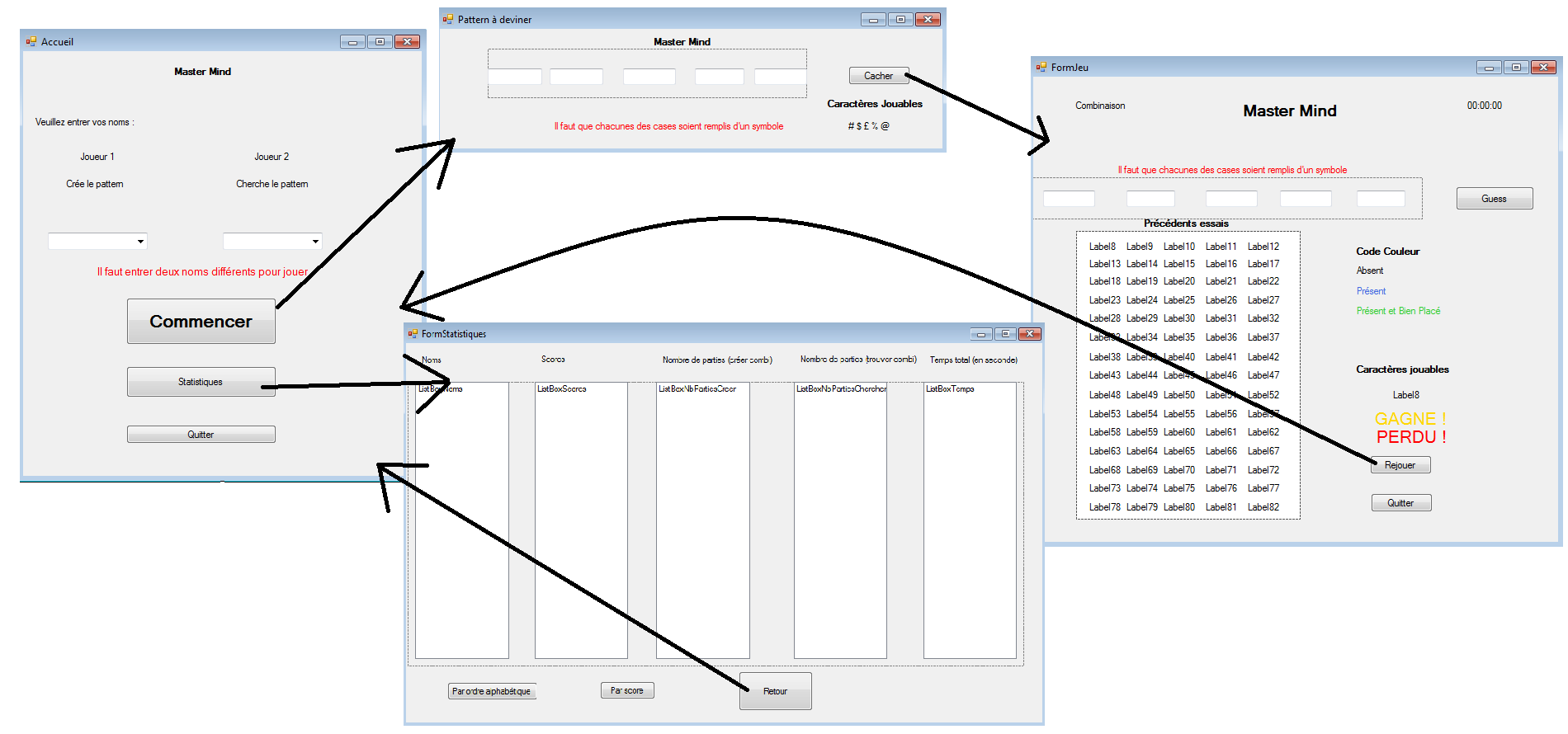
L’application que nous avons élaborée est une interprétation du jeu du Mastermind. Le jeu se joue à deux avec un joueur qui entre un code et l’autre devra le trouver. Dès l’ouverture de l’application, nous avons une fenêtre d’accueil qui apparait avec la possibilité d’entrer les noms des deux joueurs. Leurs noms devront être différents afin de commencer la partie.

En cliquant sur le bouton commencer, une nouvelle fenêtre apparaitra et le joueur 1 entrera en action. Il devra entrer les caractères possibles dans chaque case. Chaque case sera composé d’un caractère et une fois cette condition remplie, il pourra cliquer sur le bouton cacher pour permettre le joueur 2 de tenter la recherche. Un message apparaitra si le joueur 1 ne remplis pas ces conditions.

Une fois le bouton cacher actionné, une nouvelle fenêtre apparaitra et le joueur 2 pour commencer à chercher le pattern. Il timer se lance, je joueur 2 pourra donc commencer sa recherche en mettant les caractères dans les cases. Le joueur 2 dispose en tout de 15 coups afin de trouver la solution. Il devra appuyer sur le bouton guess pour voir si la combinaison entré est la bonne ou non. Toutes les combinaisons qu’il tentera seront affichées afin de l’aider à trouver la solution finale. Des couleurs l’aideront également à deviner le pattern. Bleu pour un caractère présent mais mal placé, vert pour un caractère présent et bien place et si le caractère n’est pas présent, aucune couleur ne s’affichera. Les couleurs resteront évidemment au niveau de l’affichage de ses coups précédemment joué. S’il trouve la combinaison, il gagnera un point et un message apparaitra. S’il ne trouve pas, le joueur 1 gagnera un point et un message apparaitra également. S’ils ne veulent pas continuer à jouer, il suffira d’appuyer sur le bouton quitter sinon, il suffira d’appuyer sur le bouton rejouer qui les amènera directement sur la fenêtre d’accueil avec le nom des joueurs automatiquement inversé pour une revanche.

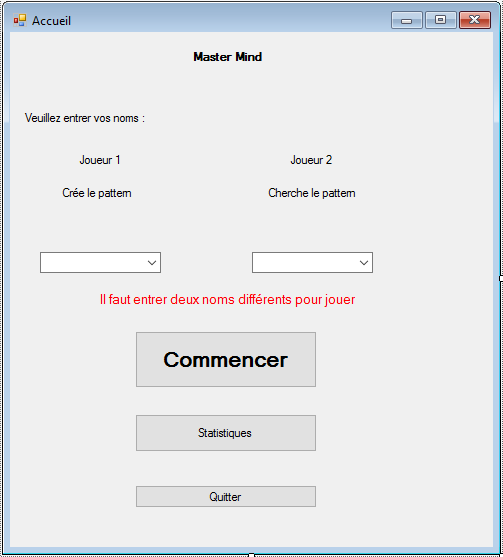
Ils pourront ensuite prendre connaissance des statistiques en cliquant simplement sur le bouton statistique. Cette fenêtre s’affichera et nous pourront distinguer les différents joueurs inscris avec leurs nombres de points, leurs nombres de parties en tant que créateur du pattern mais également en tant que chercheur du pattern. Il y aura également le temps total en train de chercher le pattern. Les joueurs pourront également faire un tri par nom mais aussi un tri par score en cliquant sur les boutons indiqués. Le nouvel affichage selon le tri voulu sera immédiatement affiché.

# Schémas d’ordonnancement des feuilles



# Feuilles et Codes

## Fenêtre accueil avec son code



Imports System.IO

Public Class FormAccueil

Dim nombrenoms As Integer

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim test As Boolean = False

If ComboBox1.Text IsNot "" And ComboBox2.Text IsNot "" Then

If ComboBox1.Text <> ComboBox2.Text Then

test = True

End If

End If

If test Then

If ComboBox1.Items.Contains(ComboBox1.Text) = False And ComboBox1.Items.Contains(ComboBox2.Text) = False Then

ComboBox1.Items.Add(ComboBox1.Text)

ComboBox1.Items.Add(ComboBox2.Text)

ComboBox2.Items.Add(ComboBox1.Text)

ComboBox2.Items.Add(ComboBox2.Text)

End If

FormPattern.Show()

Me.Hide()

End If

If test = False Then

LabelError.Visible = True

End If

End Sub

Private Sub ComboBox1\_TextChanged(sender As Object, e As EventArgs) Handles ComboBox1.TextChanged, ComboBox2.TextChanged

LabelError.Visible = False

End Sub

Private Sub ButtonStatistiques\_Click(sender As Object, e As EventArgs) Handles ButtonStatistiques.Click

If getnbj() - 1 > 0 Then

statsjoueurs()

End If

FormStatistiques.Show()

Me.Hide()

End Sub

Private Sub ButtonQuitter\_Click(sender As Object, e As EventArgs) Handles ButtonQuitter.Click

Dim choix As Integer = MsgBox("Voulez vous vraiment quitter ?", MsgBoxStyle.YesNo, "Quitter")

If choix = 6 Then

Application.Exit()

Ecrire()

End If

End Sub

Sub statsjoueurs()

FormStatistiques.ListBoxScores.Items.Clear()

FormStatistiques.ListBoxNoms.Items.Clear()

FormStatistiques.ListBoxNbPartiesCreer.Items.Clear()

FormStatistiques.ListBoxNbPartiesChercher.Items.Clear()

FormStatistiques.ListBoxTemps.Items.Clear()

For i As Integer = 0 To getnbj() - 1

FormStatistiques.ListBoxNoms.Items.Add(getJoueuri(i).nom)

FormStatistiques.ListBoxScores.Items.Add(getJoueuri(i).score)

FormStatistiques.ListBoxNbPartiesCreer.Items.Add(getJoueuri(i).nbPartieJoueur1)

FormStatistiques.ListBoxNbPartiesChercher.Items.Add(getJoueuri(i).nbPartieJoueur2)

FormStatistiques.ListBoxTemps.Items.Add(getJoueuri(i).temps)

Next

End Sub

Sub Ecrire()

Dim sw As StreamWriter = New StreamWriter("z:\Score.txt")

sw.Write("", False)

For i As Integer = 0 To getnbj() - 1

sw.WriteLine(getJoueuri(i).nom)

sw.WriteLine(getJoueuri(i).score)

sw.WriteLine(getJoueuri(i).nbPartieJoueur1)

sw.WriteLine(getJoueuri(i).nbPartieJoueur2)

sw.WriteLine(getJoueuri(i).temps)

Next

sw.Close()

End Sub

Private Sub FormAccueil\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

recuperer()

For i As Integer = 0 To getnbj() - 1

ComboBox1.Items.Add(getJoueuri(i).nom)

ComboBox2.Items.Add(getJoueuri(i).nom)

Next

End Sub

Sub recuperer()

Dim sr As StreamReader = New StreamReader("z:\Score.txt")

Dim lignes As Integer = File.ReadAllLines("z:\Score.txt").Length

For i As Integer = 0 To lignes / 5

For j As Integer = 0 To 4

créerJoueur(sr.ReadLine(), sr.ReadLine(), sr.ReadLine(), sr.ReadLine(), sr.ReadLine())

Next

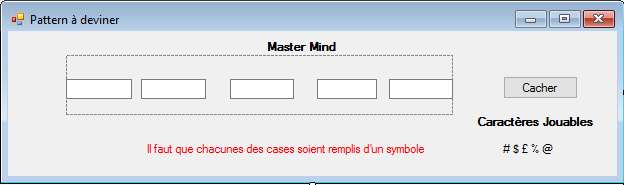
Next

sr.Close()

End Sub

End Class

## Fenêtre pattern avec son code



Public Class FormPattern

Dim CaractèresAutorisés As String = "#$£%@"

Dim Jeu As String

Private Sub TextBox1\_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged, TextBox2.TextChanged, TextBox3.TextChanged, TextBox4.TextChanged, TextBox5.TextChanged

If CaractèresAutorisés.Contains(sender.text) Then

LabelError.Visible = False

Exit Sub

Else

sender.text = ""

LabelError.Visible = False

End If

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

Dim test As Boolean = True

For Each txbx As TextBox In PanelCacher.Controls

If txbx.Text = "" Then

test = False

LabelError.Visible = True

End If

Next

If test = True Then

For Each txbx As TextBox In PanelCacher.Controls

Jeu = Jeu + txbx.Text

Next

Me.Hide()

FormJeu.LabelComb.Text = Jeu

FormJeu.Show()

FormJeu.Timer1.Start()

End If

End Sub

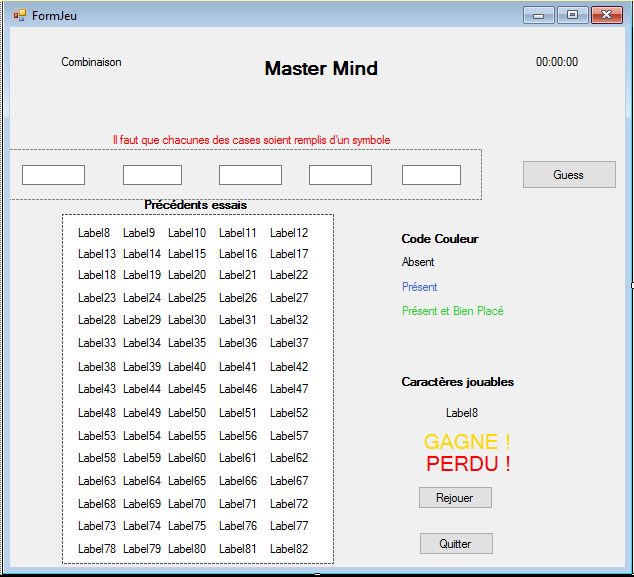
Private Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

LabelCaractère.Text = CaractèresAutorisés

End Sub

End Class

## Fenêtre de jeu avec son code



Public Class FormJeu

Dim CoupsRestants As Integer = 15

Dim Combinaison(4) As String

Dim Resultats(4, 14) As Label

Dim compt As Integer = 0

Dim CombinaisonGagnante(4) As String

Dim time As Integer = 0

Dim indice As Integer = 4

Private Sub FormJeu\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Me.Text = "Il vous reste " + CoupsRestants.ToString + " coups restants" + " 00:00:00"

LabelCaractère.Text = FormPattern.LabelCaractère.Text

CombinaisonGagnante(0) = FormPattern.TextBox1.Text

CombinaisonGagnante(1) = FormPattern.TextBox2.Text

CombinaisonGagnante(2) = FormPattern.TextBox3.Text

CombinaisonGagnante(3) = FormPattern.TextBox4.Text

CombinaisonGagnante(4) = FormPattern.TextBox5.Text

Resultats(0, 0) = Me.Label8

Resultats(1, 0) = Me.Label9

Resultats(2, 0) = Me.Label10

Resultats(3, 0) = Me.Label11

Resultats(4, 0) = Me.Label12

Resultats(0, 1) = Me.Label13

Resultats(1, 1) = Me.Label14

Resultats(2, 1) = Me.Label15

Resultats(3, 1) = Me.Label16

Resultats(4, 1) = Me.Label17

Resultats(0, 2) = Me.Label18

Resultats(1, 2) = Me.Label19

Resultats(2, 2) = Me.Label20

Resultats(3, 2) = Me.Label21

Resultats(4, 2) = Me.Label22

Resultats(0, 3) = Me.Label23

Resultats(1, 3) = Me.Label24

Resultats(2, 3) = Me.Label25

Resultats(3, 3) = Me.Label26

Resultats(4, 3) = Me.Label27

Resultats(0, 4) = Me.Label28

Resultats(1, 4) = Me.Label29

Resultats(2, 4) = Me.Label30

Resultats(3, 4) = Me.Label31

Resultats(4, 4) = Me.Label32

Resultats(0, 5) = Me.Label33

Resultats(1, 5) = Me.Label34

Resultats(2, 5) = Me.Label35

Resultats(3, 5) = Me.Label36

Resultats(4, 5) = Me.Label37

Resultats(0, 6) = Me.Label38

Resultats(1, 6) = Me.Label39

Resultats(2, 6) = Me.Label40

Resultats(3, 6) = Me.Label41

Resultats(4, 6) = Me.Label42

Resultats(0, 7) = Me.Label43

Resultats(1, 7) = Me.Label44

Resultats(2, 7) = Me.Label45

Resultats(3, 7) = Me.Label46

Resultats(4, 7) = Me.Label47

Resultats(0, 8) = Me.Label48

Resultats(1, 8) = Me.Label49

Resultats(2, 8) = Me.Label50

Resultats(3, 8) = Me.Label51

Resultats(4, 8) = Me.Label52

Resultats(0, 9) = Me.Label53

Resultats(1, 9) = Me.Label54

Resultats(2, 9) = Me.Label55

Resultats(3, 9) = Me.Label56

Resultats(4, 9) = Me.Label57

Resultats(0, 10) = Me.Label58

Resultats(1, 10) = Me.Label59

Resultats(2, 10) = Me.Label60

Resultats(3, 10) = Me.Label61

Resultats(4, 10) = Me.Label62

Resultats(0, 11) = Me.Label63

Resultats(1, 11) = Me.Label64

Resultats(2, 11) = Me.Label65

Resultats(3, 11) = Me.Label66

Resultats(4, 11) = Me.Label67

Resultats(0, 12) = Me.Label68

Resultats(1, 12) = Me.Label69

Resultats(2, 12) = Me.Label70

Resultats(3, 12) = Me.Label71

Resultats(4, 12) = Me.Label72

Resultats(0, 13) = Me.Label73

Resultats(1, 13) = Me.Label74

Resultats(2, 13) = Me.Label75

Resultats(3, 13) = Me.Label76

Resultats(4, 13) = Me.Label77

Resultats(0, 14) = Me.Label78

Resultats(1, 14) = Me.Label79

Resultats(2, 14) = Me.Label80

Resultats(3, 14) = Me.Label81

Resultats(4, 14) = Me.Label82

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles ButtonGuess.Click

Dim test As Boolean = True

Dim bon As Integer = 0

For Each txbx As TextBox In PanelGuess.Controls

If txbx.Text = "" Or CoupsRestants <= 0 Then

test = False

LabelError.Visible = True

Exit For

End If

Next

If test = True Then

Combinaison(0) = TextBox1.Text

Combinaison(1) = TextBox2.Text

Combinaison(2) = TextBox3.Text

Combinaison(3) = TextBox4.Text

Combinaison(4) = TextBox5.Text

bon = 0

For Each txbx As TextBox In PanelGuess.Controls

Resultats(indice, compt).Text = txbx.Text

If (txbx.Text = CombinaisonGagnante(indice)) Then

txbx.BackColor = Color.Green

Resultats(indice, compt).ForeColor = Color.Green

bon = bon + 1

ElseIf CombinaisonGagnante.Contains(txbx.Text) Then

txbx.BackColor = Color.Blue

Resultats(indice, compt).ForeColor = Color.Blue

Else

txbx.BackColor = Color.White

Resultats(indice, compt).ForeColor = Color.Black

End If

Resultats(indice, compt).Visible = True

indice = indice - 1

Next

indice = 4

If bon = 5 Then

Timer1.Stop()

LabelVictoite.Visible = True

ButtonGuess.Enabled = False

créerJoueur(FormAccueil.ComboBox1.Text, 0, 1, 0, 0)

créerJoueur(FormAccueil.ComboBox2.Text, 1, 0, 1, time)

Exit Sub

End If

compt = compt + 1

CoupsRestants = CoupsRestants - 1

Me.Text = "Il vous reste " & CoupsRestants & " coups restants " & (time \ 3600).ToString("00") & ":" & ((time Mod 3600) \ 60).ToString("00") & ":" & ((time Mod 3600) Mod 60).ToString("00")

If CoupsRestants <= 0 Then

créerJoueur(FormAccueil.ComboBox1.Text, 1, 1, 0, 0)

créerJoueur(FormAccueil.ComboBox2.Text, 0, 0, 1, time)

LabelDéfaite.Visible = True

ButtonGuess.Enabled = False

Exit Sub

End If

End If

End Sub

Private Sub TextBox1\_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged, TextBox2.TextChanged, TextBox3.TextChanged, TextBox4.TextChanged, TextBox5.TextChanged

If LabelCaractère.Text.Contains(sender.text) Then

LabelError.Visible = False

Exit Sub

Else

sender.text = ""

LabelError.Visible = False

End If

End Sub

Private Sub ButtonQuitter\_Click(sender As Object, e As EventArgs) Handles ButtonQuitter.Click

Dim choix As Integer = MsgBox("Voulez vous vraiment quitter ?", MsgBoxStyle.YesNo, "Quitter")

If choix = 6 Then

Application.Exit()

FormAccueil.Ecrire()

End If

End Sub

Private Sub ButtonRejouer\_Click(sender As Object, e As EventArgs) Handles ButtonRejouer.Click

Erase Resultats

FormAccueil.Show()

Me.Close()

time = 0

LabelTemps.Text = "00:00:00"

CoupsRestants = 15

ButtonGuess.Enabled = True

For Each txbx As TextBox In PanelGuess.Controls

txbx.Text = ""

txbx.BackColor = Color.White

Next

For Each lbl As Label In PanelReponse.Controls

lbl.Text = ""

lbl.ForeColor = Color.White

lbl.Visible = False

Next

For Each txbx As TextBox In FormPattern.PanelCacher.Controls

txbx.Text = ""

Next

LabelDéfaite.Visible = False

LabelVictoite.Visible = False

Dim tmp As String = FormAccueil.ComboBox1.Text

FormAccueil.ComboBox1.Text = FormAccueil.ComboBox2.Text

FormAccueil.ComboBox2.Text = tmp

End Sub

Private Sub Timer1\_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick

time = time + 1

Dim h As Integer = time \ 3600

Dim m As Integer = (time Mod 3600) \ 60

Dim s As Integer = (time Mod 3600) Mod 60

Me.Text = "Il vous reste " & CoupsRestants & " coups restants " & h.ToString("00") & ":" & m.ToString("00") & ":" & s.ToString("00")

LabelTemps.Text = h.ToString("00") & ":" & m.ToString("00") & ":" & s.ToString("00")

End Sub

Private Sub Enregistrer\_Score\_Gagnantj2()

If Not FormStatistiques.ListBoxNoms.Items.Contains(FormAccueil.ComboBox1.Text) Then

FormStatistiques.ListBoxNoms.Items.Add(FormAccueil.ComboBox1.Text)

FormStatistiques.ListBoxScores.Items.Add(0)

FormStatistiques.ListBoxTemps.Items.Add(0)

FormStatistiques.ListBoxNbPartiesCreer.Items.Add(1)

FormStatistiques.ListBoxNbPartiesChercher.Items.Add(0)

Else

Dim i As Integer

i = FormStatistiques.ListBoxNoms.Items.IndexOf(FormAccueil.ComboBox1.Text)

FormStatistiques.ListBoxNbPartiesCreer.Items.Item(i) = FormStatistiques.ListBoxNbPartiesCreer.Items(i) + 1

End If

If Not FormStatistiques.ListBoxNoms.Items.Contains(FormAccueil.ComboBox2.Text) Then

FormStatistiques.ListBoxNoms.Items.Add(FormAccueil.ComboBox2.Text)

FormStatistiques.ListBoxScores.Items.Add(1)

FormStatistiques.ListBoxTemps.Items.Add(time)

FormStatistiques.ListBoxNbPartiesCreer.Items.Add(0)

FormStatistiques.ListBoxNbPartiesChercher.Items.Add(1)

Else

Dim i As Integer

i = FormStatistiques.ListBoxNoms.Items.IndexOf(FormAccueil.ComboBox2.Text)

FormStatistiques.ListBoxNbPartiesChercher.Items.Item(i) = FormStatistiques.ListBoxNbPartiesChercher.Items(i) + 1

FormStatistiques.ListBoxTemps.Items(i) = FormStatistiques.ListBoxTemps.Items(i) + time

FormStatistiques.ListBoxScores.Items(i) = FormStatistiques.ListBoxScores.Items(i) + 1

End If

End Sub

Private Sub Enregistrer\_Score\_Gagnantj1()

If Not FormStatistiques.ListBoxNoms.Items.Contains(FormAccueil.ComboBox2.Text) Then

FormStatistiques.ListBoxNoms.Items.Add(FormAccueil.ComboBox2.Text)

FormStatistiques.ListBoxScores.Items.Add(0)

FormStatistiques.ListBoxTemps.Items.Add(0)

FormStatistiques.ListBoxNbPartiesCreer.Items.Add(1)

FormStatistiques.ListBoxNbPartiesChercher.Items.Add(0)

Else

Dim i As Integer

i = FormStatistiques.ListBoxNoms.Items.IndexOf(FormAccueil.ComboBox2.Text)

FormStatistiques.ListBoxNbPartiesChercher.Items.Item(i) = FormStatistiques.ListBoxNbPartiesChercher.Items(i) + 1

End If

If Not FormStatistiques.ListBoxNoms.Items.Contains(FormAccueil.ComboBox1.Text) Then

FormStatistiques.ListBoxNoms.Items.Add(FormAccueil.ComboBox1.Text)

FormStatistiques.ListBoxScores.Items.Add(1)

FormStatistiques.ListBoxTemps.Items.Add(time)

FormStatistiques.ListBoxNbPartiesCreer.Items.Add(0)

FormStatistiques.ListBoxNbPartiesChercher.Items.Add(1)

Else

Dim i As Integer

i = FormStatistiques.ListBoxNoms.Items.IndexOf(FormAccueil.ComboBox1.Text)

FormStatistiques.ListBoxNbPartiesCreer.Items.Item(i) = FormStatistiques.ListBoxNbPartiesCreer.Items(i) + 1

FormStatistiques.ListBoxTemps.Items(i) = FormStatistiques.ListBoxTemps.Items(i) + time

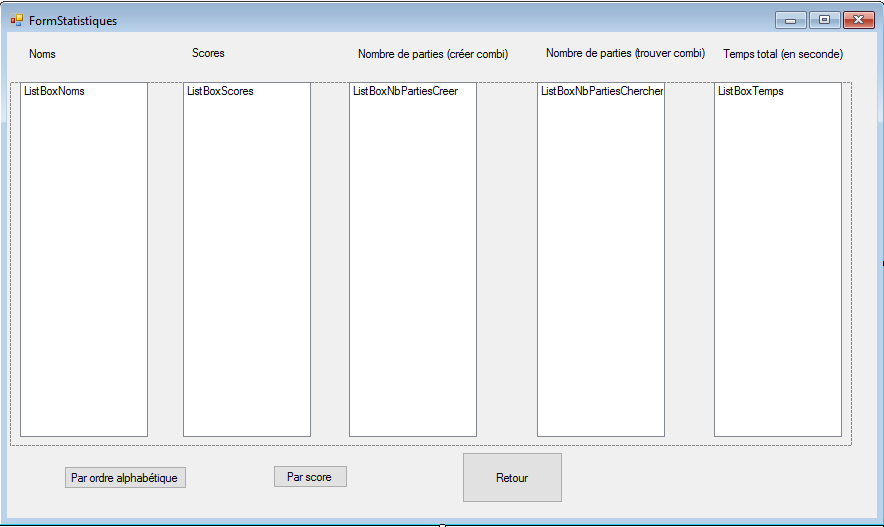
FormStatistiques.ListBoxScores.Items(i) = FormStatistiques.ListBoxScores.Items(i) + 1

End If

End Sub

End Class

## Fenêtre des statistiques avec son code



Public Class FormStatistiques

Private Sub ButtonRetour\_Click(sender As Object, e As EventArgs) Handles ButtonRetour.Click

FormAccueil.Show()

Me.Hide()

End Sub

Private Sub ListBoxNbPartiesCreer\_TabIndexChanged(sender As Object, e As EventArgs) Handles ListBoxNbPartiesCreer.SelectedIndexChanged, ListBoxNbPartiesChercher.SelectedIndexChanged, ListBoxNoms.SelectedIndexChanged, ListBoxScores.SelectedIndexChanged, ListBoxTemps.SelectedIndexChanged

Dim tmp As Integer = sender.SelectedIndex

ListBoxNbPartiesCreer.SelectedIndex = tmp

ListBoxNbPartiesChercher.SelectedIndex = tmp

ListBoxNoms.SelectedIndex = tmp

ListBoxScores.SelectedIndex = tmp

ListBoxTemps.SelectedIndex = tmp

End Sub

Private Sub ButtonTriParScore\_Click(sender As Object, e As EventArgs) Handles ButtonTriParScore.Click

Dim a As String

For i As Integer = 0 To ListBoxScores.Items.Count - 2 Step 1

For j As Integer = i + 1 To ListBoxScores.Items.Count - 1 Step 1

If ListBoxScores.Items(i) < ListBoxScores.Items(j) Then

For Each c As ListBox In PanelListBox.Controls

a = c.Items(i)

c.Items(i) = c.Items(j)

c.Items(j) = a

Next

End If

Next

Next

End Sub

Private Sub ButtonTriAlphabetique\_Click(sender As Object, e As EventArgs) Handles ButtonTriAlphabetique.Click

ListBoxNoms.Items.Clear()

ListBoxNbPartiesChercher.Items.Clear()

ListBoxScores.Items.Clear()

ListBoxNbPartiesCreer.Items.Clear()

ListBoxScores.Items.Clear()

Joueurs.trierParAlphabetique()

FormAccueil.statsjoueurs()

Me.Refresh()

End Sub

Private Sub ListBoxNbPartiesChercher\_TabIndexChanged(sender As Object, e As EventArgs) Handles ListBoxNbPartiesChercher.TabIndexChanged, ListBoxNbPartiesCreer.TabIndexChanged, ListBoxNoms.TabIndexChanged, ListBoxScores.TabIndexChanged, ListBoxTemps.TabIndexChanged

Dim tmp As Integer = sender.Tabindex

ListBoxNbPartiesCreer.TabIndex = tmp

ListBoxNbPartiesChercher.TabIndex = tmp

ListBoxNoms.TabIndex = tmp

ListBoxScores.TabIndex = tmp

ListBoxTemps.TabIndex = tmp

End Sub

Private Sub FormStatistiques\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

End Sub

End Class

# Code du module

Module Joueurs

Dim joueurs(5) As joueur

Dim nbJoueurs As Integer = 0

Structure joueur

Dim nom As String

Dim score As Integer

Dim nbPartieJoueur1 As Integer

Dim nbPartieJoueur2 As Integer

Dim temps As Integer

End Structure

Sub créerJoueur(nom As String, score As Integer, nbJoueur1 As Integer, nbJoueur2 As Integer, temps As Integer)

Dim trouvé As Boolean = False

For i As Integer = 0 To UBound(joueurs)

If joueurs(i).nom = nom Then

trouvé = True

joueurs(i).score = joueurs(i).score + score

joueurs(i).nbPartieJoueur1 = joueurs(i).nbPartieJoueur1 + nbJoueur1

joueurs(i).nbPartieJoueur2 = joueurs(i).nbPartieJoueur2 + nbJoueur2

joueurs(i).temps = joueurs(i).temps + temps

End If

Next

If trouvé = False Then

Dim persn As joueur

persn.nom = nom

persn.score = score

persn.nbPartieJoueur1 = nbJoueur1

persn.nbPartieJoueur2 = nbJoueur2

persn.temps = temps

If (nbJoueurs + 1 = joueurs.Length) Then

ReDim Preserve joueurs(UBound(joueurs) + 5)

End If

joueurs(nbJoueurs) = persn

nbJoueurs = nbJoueurs + 1

End If

End Sub

Function getnbj() As Integer

Return nbJoueurs

End Function

Sub enregistrerjoueur2gagnant()

End Sub

Function existe(name As String) As Integer

For i As Integer = 0 To UBound(joueurs) Step (+1)

If joueurs(i).nom = name Then

Return i

End If

Next

Return vbNull

End Function

Function getJoueuri(i As Integer) As joueur

Return joueurs(i)

End Function

Public Sub trierParAlphabetique()

Dim a As joueur

For i = 0 To nbJoueurs - 1

For j = i To nbJoueurs - 1

If Compare(joueurs(i).nom, joueurs(j).nom) = True Then

a = joueurs(i)

joueurs(i) = joueurs(j)

joueurs(j) = a

End If

Next

Next

End Sub

Function Compare(Chaine1 As String, Chaine2 As String) As Boolean

If Trim(Chaine1) > Trim(Chaine2) Then

Return True

Else : Return False

End If

End Function

End Module

# Exemple d’inscription

L'enregistrement d'une inscription se fait lorsqu'une partie se termine, lorsque le joueur 1 perd ou gagne, les informations telles que le score à la fin, le temps et le nom du joueur sont ajouté dans une nouvelle variable de type de la structure Joueur que nous avons créé dans le module Joueur, on ajoute ce nouveau Joueur dans un tableau de joueurs lui aussi contenu dans le module Joueur.

Lorsqu'une personne quitte l'application normalement (en utilisant le bouton quitter se trouvant sur la feuille d'accueil ou sur celui se trouvant sur la feuille où le jeu se déroule une fois celui-ci fini, donc en n'appuyant pas sur rejouer) tous les joueurs sont sauvegarder dans un fichier texte que nous avons choisi de créer dans le dossier Z:, son nom est Score.txt et son chemin d'accès est donc Z:\Score.txt.

Pour enregistrer sur le fichier texte, nous avons utilisé une variable de type StreamWriter qui permet d'écrire sur un fichier texte, afin de faciliter la lecture du fichier texte lors de la réouverture de l'application MasterMind, nous avons décidé d'écrire une donnée par ligne dans cet ordre :

-Le nom du Joueur

-Le score du Joueur

-Le nombre de fois où il a joué une partie en Créant le pattern

-Le nombre de fois où il a joué une partie en Cherchant le pattern

-Le temps cumulé de toutes les parties où il a cherché des patterns dans toutes les parties auxquelles il a participé.

Enfin, à l'aide d'une variable de type StreamReader nous pouvons relire le fichier texte à l'ouverture de l'application et ranger tous les précédents joueurs dans le tableau de joueurs afin de garder les statistiques malgré une fermeture du jeu ainsi que des différents noms dans les combobox dans le formulaire d'accueil (Les noms de tous les précédents joueurs seront présents dans la comboBox en liste déroulante).

# Conclusion

Pour conclure, la création de cette application a été une réussite. Nous avons réussi à terminer l’application dans les temps grâce à notre parfaite entente et notre travail d’équipe très efficace. Nous avons su se partager parfaitement les taches que chacun devra accomplir afin de d’avancer sans problème et sans nous mettre des bâtons dans les roues.

Nous avons également beaucoup appris sur le langage VisualBasic en y passant beaucoup de temps dessus. Grâce au cours mais également à des recherches sur le net, nous n’avons pas rencontré d’énorme problème dans le code.