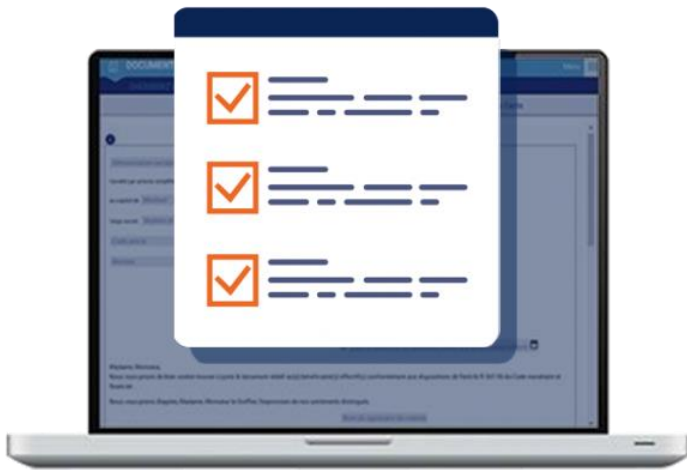

Projet d'IHM – Visual Basic .Net



ANNEE 2019/2020

IUT Paris Descartes

**SOUMARE Mouhamadou 105
RAHARINOSY Mamilala 111**



Sommaire

| | |
|--|----|
| I - Introduction :..... | 3 |
| II - Schéma d'ordonnancement : | 6 |
| III – Exemple du fichier de sauvegarde des inscriptions..... | 9 |
| IV – Formulaires et leur code | 10 |
| V – Conclusion | 39 |

I - Introduction :

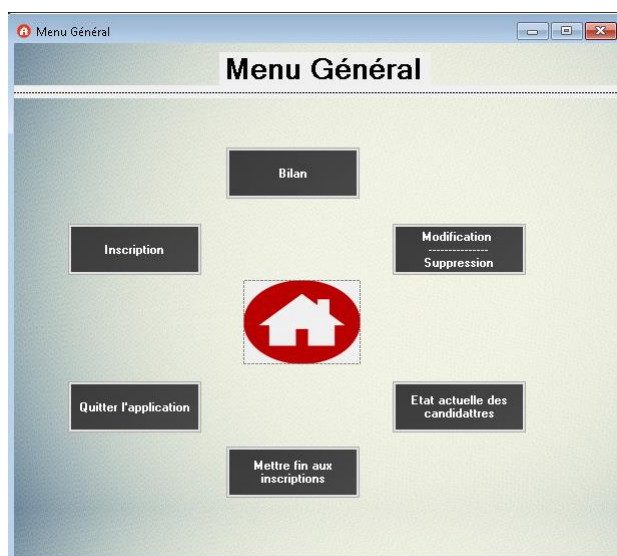
Le projet d'introduction aux interfaces hommes-machines consiste à la création d'une application dans le langage de programmation Visual Basic.net. Ce programme est à réaliser en binôme ou seul. Pour répondre aux attentes du rendu final, on se doit de respecter différents critères et contraintes qui font partie du cycle de développement logiciel. Ce projet reflète une cohésion et un travail concret du groupe au fil des différentes semaines. Il permet en outre de nous mettre face à des difficultés que l'on se doit de surmonter par nous-mêmes, avec autonomie. Cela permet de développer les compétences et les qualités de chacun.

En ce qui concerne l'application, il est codé sur Visual Studio (environnement de développement intégré) via le langage de programmation Vb.net reposant sur la bibliothèque Win-Forms qu'on étudie lors de la période D, semestre 2.

Visual Basic.net est un langage de programmation événementielle. Le développement du programme et plus précisément de la simulation d'une gestion d'un concours avec, est caractérisé par l'avancée réalisée par chaque groupe au niveau de la conception des différents formulaires que l'on doit mettre en place, de l'inscription à un bilan soit individuel soit par matière. Le but principal du projet est donc de créer une application capable de répondre aux besoins d'un acteur lors de la gestion d'un concours, notamment aux niveaux des inscriptions des candidats.

Nous avons 9 formulaires intégrés à l'application ayant chacun leurs spécificités et répondant à un besoin concrets :

1. Le premier formulaire correspond au menu général, c'est l'accueil de l'application qui permet l'accès aux autres formulaires, c'est donc la « racine » de l'application.



2. Le deuxième formulaire correspond a la saisie des données personnels du candidat, il y entre son nom, prénom, âge qui permettent son identification. L'utilisateur peut soit annuler son inscription soit continue et ainsi choisir les épreuves qu'il souhaite passer dans une région donné. Ce formulaire doit être rempli en 1 min, au delà l'application retourne a l'accueil.

The screenshot shows a window titled 'Inscription'. Inside, there are several text input fields labeled 'Nom:', 'Prénom:', 'Adresse:', 'Code Postale:', 'Ville:', and 'Age:'. The 'Age' field has a dropdown arrow. Below the fields are two buttons: 'Abandonner' and 'Continuer'. At the very bottom, there is a small text label 'Informations Obligatoires'.

3. Le troisième formulaire correspond a la sélection de la région de passage du concours et choix des différentes épreuves écrites, orales et facultatif. Ce formulaire doit être rempli en 1 min 30, au delà de ce temps limite imposé, l'application retourne à l'accueil.

The screenshot shows a window titled 'Choix des épreuves'. It has a complex layout with several sections. On the left, there are labels for 'Prénom', 'Nom', and 'Région' (with a dropdown). The main area is titled 'Epreuve du concours' and is divided into three columns: 'Ecrit' (Written), 'Oral', and 'Epreuve facultatif'. Each column contains a list of subjects with checkboxes. The 'Ecrit' column includes subjects like Algorithmique, Base de données, Droit, Expression, Gestion, Langage C, Langage Java, Mathématiques, Programmation Web, Réseau, Système, and Visual Basic.Net. The 'Oral' column includes Anglais, Chinois, Espagnol, Droit, Expression, Gestion, Mathématiques, Réseau, and Système. The 'Epreuve facultatif' column has radio buttons for 'Oui' and 'Non', and a dropdown menu. At the bottom, there are two buttons: 'Annuler' and 'Valider'.

4. Le quatrième formulaire est l'un des aboutissements de l'arborescence de l'application, elle permet l'enregistrement final de la candidature. L'application sauvegarde à travers un fichier de sauvegarde les différentes données utiles à la caractérisation du candidat. L'utilisateur peut soit annuler sa candidature et alors revenir à l'accueil ou valider sa candidature qui mène à la capture de celui-ci dans le système de sauvegarde. Ce formulaire se présente comme un résumé, un récapitulatif. Si le candidat n'entre pas d'adresse, une information non obligatoire, alors l'application rend invisible les labels le représentant. De même pour l'épreuve facultative.

Récapitulatif

Données Saisie

Nom Label2 Région: Label19

Prénom Label4

Ville Label6

Code Postal Label8

Age Label12

Adresse Label10

Epreuves Ecrites:

Epreuves Orales:

Epreuves Facultatif:

Annuler Retour Enregistrer

5. Le cinquième formulaire est accessible à partir de l'accueil, celui-ci permet à l'utilisateur de modifier ou supprimer sa candidature. Il a deux options, soit faire une recherche à travers l'identifiant qui lui est donné par l'application, soit faire une recherche via le nom et prénom qu'il a rentré lors de son inscription. Il y'a deux Combobox, ils sont visibles en fonction du type de recherches que souhaite effectuer l'utilisateur.

Action sur candidat

Modification Ou Suppression

Type de recherche :

☐ Numéro

☐ Identifiant

Identifiant : N° :

Supprimer Modifier

Retour à l'accueil

6. Le sixième formulaire est aussi accessible à partir de la racine de l'application .Celui ci se représente un récapitulatif d'ensemble. On peut y retrouver l'ensemble des candidatures enregistrées avant la clôture de la session d'inscription au concours.

The screenshot shows a window titled "Candidats inscrits" with the subtitle "Etat actuels des inscriptions". It contains five vertical panels, each with a header and a large empty text area below it:

- Identifiant**: LatId
- Nom Prénom**: LatNomPren
- Epreuves écrites**: LatEcrits
- Epreuves orales**: LatOral
- Epreuve facultatif**: LatFac

At the bottom center, there is a button labeled "Retour à l'accueil".

7. Le septième formulaire est accessible via l'accueil et permet de connaître le nombre de candidats enregistrés en cours et propose à l'utilisateur un bilan individuel ou un bilan par matière. Nous avons deux combobox qui sont ici aussi visible ou invisible selon le type de bilan que souhaite afficher l'utilisateur.

The screenshot shows a window titled "Bilan". It contains the following elements:

- A text label "Nombre d'inscriptions enregistrées :" followed by an empty text input field.
- A section titled "Choix du bilan :" containing two radio buttons: "Bilan Individuel" and "Bilan Matière".
- A text label "Matière : :" followed by a dropdown menu.
- At the bottom, two buttons: "Retour à l'accueil" and "Afficher le Bilan".

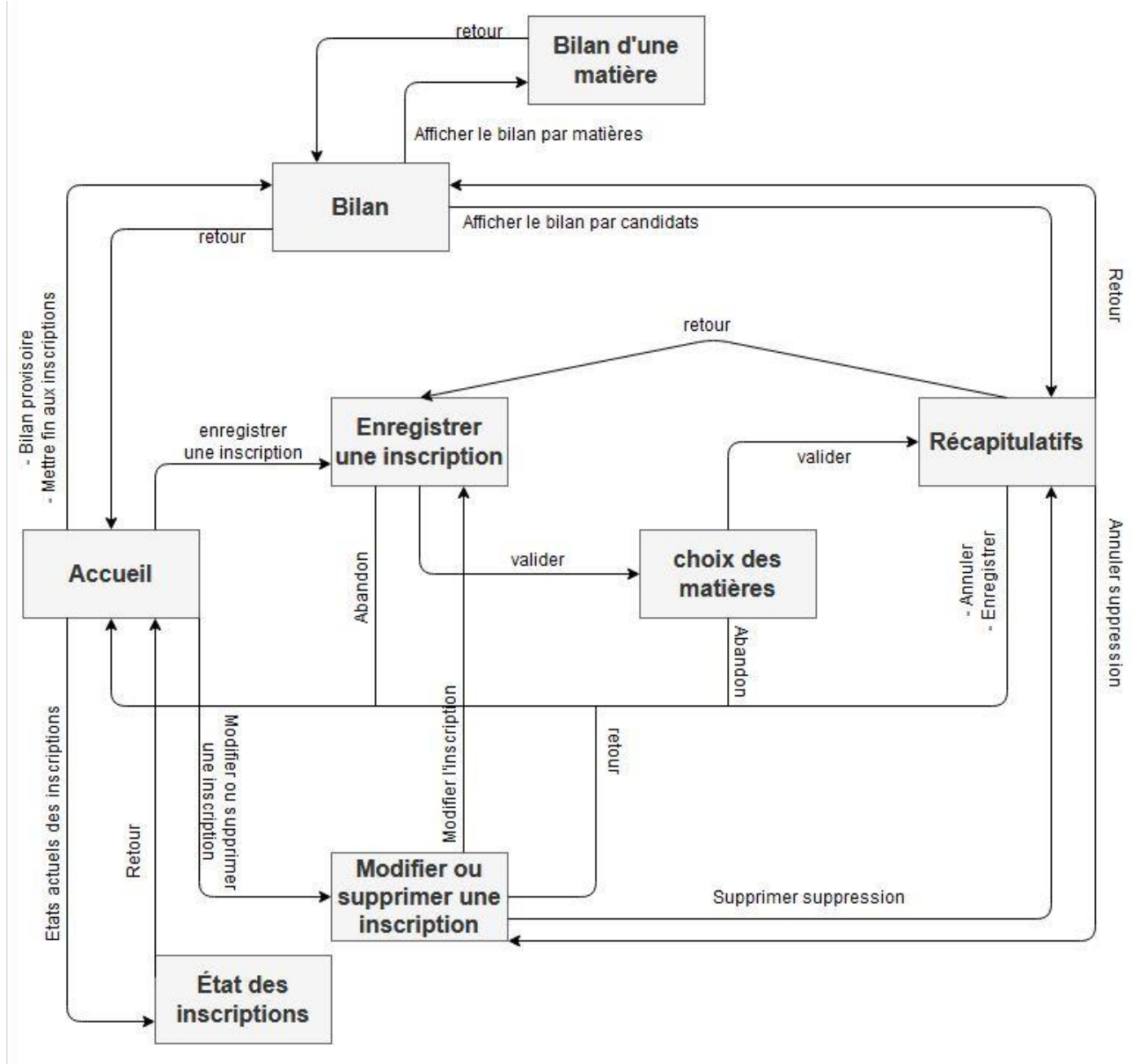
8. Le dernier formulaire est accessible via le formulaire 7 et permet d'avoir une vue d'ensemble sur les candidats ayant choisis tels ou tels matières pour tels ou tels épreuves en fonction de la recherche de l'utilisateur.

The screenshot shows a web application window titled "Bilan Matière". Inside the window, there is a table with the following columns: Id, Nom, Prénom, Région, Age, Ecrit, Oral, and Facultatif. The 'Ecrit', 'Oral', and 'Facultatif' columns contain checkboxes. Below the table, there is a button labeled "Retour à l'accueil".

| | Id | Nom | Prénom | Région | Age | Ecrit | Oral | Facultatif |
|---|----|-----|--------|--------|-----|--------------------------|--------------------------|--------------------------|
| * | | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Retour à l'accueil

II - Schéma d'ordonnancement :



III – Exemple du fichier de sauvegarde des inscriptions

```
vec.txt - Bloc-notes
Fichier Edition Format Affichage Aide
| | @Projet Vb, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null|| System.Collections.Generic.List`1[[Projet_Vb.Module1+Candidat, Projet Vb, Ver
|Corse || ^ |éýýýüýýý |Keller|Clement ||Paris|Base de donnÃes| Bordelais || éýýýüýýý lg||Gauch
Roussillon |éýýýüýýý ||Alkaline!|Moent ||#|Auber ||Bretagne & '|System.Collections.Generic.List`1[Sys
!* ||7 |Droit8
Expression9 Language C#: |Programmation Web|+ ||;|Chinois|< |Anglais|= |MathÃ©matiques
|, || 7 8 |@ ^Language Java = |- ||B |Espagnol|C |Gestion|D |RÃ©seau
|. ||E
Expression1F Language C|G |RÃ©seauH |SystÃme||I |Espagnol|J |Chinois|K |Anglais
|
```

IV – Formulaire et leur code

1. Formulaire 1



```
Imports System.IO
```

```
Public Class Form1
```

```
Private cand As New Candidat
```

```
Dim matieres() As String = {"Algorithmique", "Anglais", "Base de données", "Chinois", "Droit",  
"Espagnol", "Expression", "Gestion", "Langage C", "Langage Java", "Mathématiques",  
"Programmation Web", "Réseau", "Système", "Visual Basic .NET"}
```

```
Dim rgion() As String = {"Auvergne", "Bordelais", "Bourgogne", "Bretagne", "Corse", "Nord",  
"Normandie", "Paris", "Poitou", "Roussillon"}
```

```
Function getcand()
```

```
Return cand
```

```
End Function
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles BtnInscription.Click
```

```
If File.Exists("num.txt") Then
```

```
Dim line As String
```

```
Using reader As StreamReader = New StreamReader("num.txt")
```

```
line = reader.ReadLine.Last
```

```
nbenregistrement = line
```

```
cand.id = line
```

```

EndUsing
Else
File.Create("num.txt").Close()
nbenregistrement = 0
    cand.id = 0
File.WriteAllText("num.txt", cand.id)
EndIf
Dim ins AsNew Form2(Me)
ins.Show(Me)
Me.Hide()
EndSub

PrivateSub Button2_Click(sender AsObject, e AsEventArgs) HandlesBtnModifier.Click
DimmodifsupprAsNew Form5
modifsuppr.Show(Me)
Me.Hide()
EndSub
PrivateSub Button6_Click(sender AsObject, e AsEventArgs) HandlesBtnEtatactuel.Click
DimregistreAsNew Form6
registre.Show(Me)
Me.Hide()
EndSub

PrivateSub Button3_Click(sender AsObject, e AsEventArgs) HandlesBtnBilan.Click
DimbilanAsNew Form7
bilan.Show(Me)
Me.Hide()
EndSub

PrivateSub Button5_Click(sender AsObject, e AsEventArgs) HandlesBtnQuitter.Click
Application.Exit()
EndSub

PrivateSub Form1_Load(sender AsObject, e AsEventArgs) Handles MyBase.Load
    lecture()
EndSub

PrivateSubBtnFin_Click(sender AsObject, e AsEventArgs) HandlesBtnFin.Click
    arret = True
If arret = TrueThen
ForEachcandAsCandidatInlisteCandidat
ForEach s AsStringInregion
ForEachsiAsStringInmatieres
ForiAsInteger = 0 To 3
For m AsInteger = 0 To 2
Ifsi = cand.ecrits(i) And s = cand.regionThen
Using a AsStreamWriter = NewStreamWriter("../" & s & si & ".txt")
a.WriteLine(cand.id & " "&cand.nom& " "&cand.prenom& " -----> inscrit à l'épreuve écrite")
EndUsing
ElseIfsi = cand.orals(m) And s = cand.regionThen
Using a AsStreamWriter = NewStreamWriter("../" & s & si & ".txt")
a.WriteLine(cand.id & " "&cand.nom& " "&cand.prenom& " -----> inscrit à l'épreuve oral")
EndUsing
EndIf
Next
Next
Next
Next
Next
EndIf
MsgBox("Fermeture de cette session d'inscriptions aux concours", MsgBoxStyle.Information)
Application.Exit()
EndSub
EndClas

```

2. Formulaire 2

The image shows a web browser window with a title bar that says "Inscription". The main content area has a large heading "Inscription" at the top. Below the heading, there are six form fields, each preceded by an asterisk (*) indicating they are mandatory. The fields are: "Nom:" with a single-line text input; "Prenom:" with a single-line text input; "Adresse:" with a single-line text input; "Code Postale:" with a single-line text input; "Ville:" with a single-line text input; and "Age:" with a dropdown menu. At the bottom of the form area, there are two buttons: "Abandonner" on the left and "Continuer" on the right. Below the buttons, there is a small asterisk (*) followed by the text "Informations Obligatoires".

Inscription

* Nom:

* Prenom:

Adresse:

* Code Postale:

* Ville:

* Age:

* Informations Obligatoires

```

PublicClassForm2
Private temps, restant, nbtoursAsInteger
PrivatecandAsNewCandidat
Private modifier AsBoolean = False
Private nouvelle As Form1

SubNew(ByRefnvlleinscAs Form1)
InitializeComponent()
Me.nouvelle = nvlleinsc
cand = Me.nouvelle.getcand
EndSub
SubNew(ByRefRmodificationAs Form5)

InitializeComponent()
Me.cand = Rmodification.getCand
Me.modifier = True
Me.Show(Rmodification)
Me.Owner.Hide()
cmdNom.Text = cand.nom
CmdPren.Text = cand.prenom
CmdAdres.Text = cand.adress
CmdCp.Text = cand.cp
        ComboBox1.Text = cand.age
CmdVille.Text = cand.ville

EndSub

DimnbenregistrementAsInteger = 0
FunctiongetCand()
Returncand
EndFunction

PrivateSubcmdNom_KeyPress(sender AsObject, e AsKeyPressEventArgs) HandlescmdNom.KeyPress
Ife.KeyChar = vbBackThenExitSub
IfNotChar.IsLetter(e.KeyChar) Then
e.Handled = True
EndIf
cmdNom.MaxLength = 15

EndSub

PrivateSubcmdPrenom_KeyPress(sender AsObject, e AsKeyPressEventArgs) HandlesCmdPren.KeyPress
Ife.KeyChar = vbBackThenExitSub
IfNotChar.IsLetter(e.KeyChar) Then
e.Handled = True
EndIf
cmdNom.MaxLength = 20

EndSub

PrivateSubcdmNom_lostfocus(sender AsObject, e AsEventArgs) HandlescmdNom.LostFocus,
CmdPren.LostFocus, CmdVille.LostFocus
If (cmdNom.TextLength> 0) Then
cmdNom.Text = cmdNom.Text.Substring(0, 1).ToUpper() + cmdNom.Text.Substring(1).ToLower()
EndIf

If (CmdPren.TextLength> 0) Then
CmdPren.Text = CmdPren.Text.Substring(0, 1).ToUpper() + CmdPren.Text.Substring(1).ToLower()
EndIf

If (CmdVille.TextLength> 0) Then
CmdVille.Text = CmdVille.Text.Substring(0, 1).ToUpper() + CmdVille.Text.Substring(1).ToLower()
EndIf

```

EndSub

```
PrivateSub CmdAdres_KeyPress(sender As Object, e As KeyPressEventArgs) Handles CmdAdres.KeyPress
    CmdAdres.MaxLength = 30
EndSub
```

```
PrivateSub CmdCp_KeyPress(sender As Object, e As KeyPressEventArgs) Handles CmdCp.KeyPress
    If e.KeyChar = vbBack Then Exit Sub
    If Not Char.IsDigit(e.KeyChar) Then
        e.Handled = True
    End If
    CmdCp.MaxLength = 5
EndSub
```

```
PrivateSub Button1_Click(sender As Object, e As EventArgs) Handles BtnAbando.Click
    If MsgBox("Etes-vous certain de vouloir quitter l'inscription ?", vbYesNo, "Demande de confirmation") = vbYes Then
        Me.Owner.Show()
        Me.Close()
    Else
        Me.Show()
    End If
EndSub
```

```
PrivateSub cmdNom_Enter(sender As Object, e As EventArgs) Handles cmdNom.Enter
    LblNom.ForeColor = Color.Black
```

EndSub

```
PrivateSub CmdPren_Enter(sender As Object, e As EventArgs) Handles CmdPren.Enter
    LblPren.ForeColor = Color.Black
```

EndSub

```
PrivateSub CmdCp_Enter(sender As Object, e As EventArgs) Handles CmdCp.Enter
    LblCp.ForeColor = Color.Black
```

EndSub

```
PrivateSub CmdVille_Enter(sender As Object, e As EventArgs) Handles CmdVille.Enter
    LblVille.ForeColor = Color.Black
```

EndSub

```
PrivateSub ComboBox1_Enter(sender As Object, e As EventArgs) Handles ComboBox1.Enter
    LblAge.ForeColor = Color.Black
```

EndSub

```
PrivateSub ComboBox1_KeyPress(sender As Object, e As KeyPressEventArgs) Handles ComboBox1.KeyPress
    If e.KeyChar = vbBack Then Exit Sub
    If Not Char.IsDigit(e.KeyChar) Then
        e.Handled = True
    End If
    If e.KeyChar = vbBack Then Exit Sub
    If Not Char.IsLetter(e.KeyChar) Then
        e.Handled = True
    End If
EndSub
```

EndSub

```
PrivateSub CmdVille_KeyPress(sender As Object, e As KeyPressEventArgs) Handles CmdVille.KeyPress
```

```

If e.KeyChar = vbBack Then Exit Sub
If Not Char.IsLetter(e.KeyChar) Then
e.Handled = True
End If
End Sub

Private Sub Button2_Click(sender As Object, e As EventArgs) Handles BtnContinuer.Click

Dim valid As Boolean = True
Dim selec As Boolean = False

LblNom.ForeColor = Color.Black
LblPren.ForeColor = Color.Black
LblCp.ForeColor = Color.Black
LblVille.ForeColor = Color.Black

If cmdNom.TextLength = 0 Then
LblNom.ForeColor = Color.Red
valid = False
End If

If cmdPren.TextLength = 0 Then
LblPren.ForeColor = Color.Red
valid = False
End If
If cmdVille.TextLength = 0 Then
LblVille.ForeColor = Color.Red
valid = False
End If
If cmdCp.TextLength = 0 Then
LblCp.ForeColor = Color.Red
valid = False
End If
If ComboBox1.SelectedItem = 0 Then
LblAge.ForeColor = Color.Red
valid = False
End If

If cmdCp.TextLength = 0 Then
LblCp.ForeColor = Color.Red
valid = False
ElseIf cmdCp.Text.Substring(0, 2).Equals("00") Then
LblCp.ForeColor = Color.Red
MsgBox("les deux premiers chiffres doivent être différents de zéro")
valid = False
End If

selec = False
If valid Then
cand.nom = cmdNom.Text
cand.prenom = cmdPren.Text
cand.adress = cmdAdres.Text
cand.age = ComboBox1.Text
cand.ville = cmdVille.Text
cand.cp = cmdCp.Text
If cmdAdres.Text = "" Then
cand.adress = ""
Else
cand.adress = cmdAdres.Text
End If
Timer1.Stop()
Dim choix As New Form3(Me)
choix.Show(Me)

```

```

Me.Hide()
EndIf
EndSub


PrivateSub Timer1_Tick(sender AsObject, e AsEventArgs) Handles Timer1.Tick
nbtours += 1
Ifnbtours \ 10 = restantThen
restant += 1
EndIf
If temps - restant = 0 Then
Me.Owner.Show()
Me.Close()
MsgBox("L'inscription doit s'effectuer en 1 min !", MsgBoxStyle.Information)
EndIf
Me.Text = "Donnéespersonnels" + "      " + TimeOfDay + "      " + (temps - restant).ToString() +
"s"
EndSub

PrivateSub Form2_Load(sender AsObject, e AsEventArgs) Handles MyBase.Load
Timer1.Start()
    temps = 60
nbtours = 0
    restant = 0
EndSub

Functiongetmodifier()
Return modifier
EndFunction
EndClass

```


3. Formulaire 3

 Choix des épreuves

Choix des épreuves

Prénom :

Prénom

Nom :

Nom

Région

Annuler

Valider

Epreuve du concours

Ecrit

☐ Algorithmique

☐ Base de données

☐ Droit

☐ Expression

☐ Gestion

☐ Langage C

☐ Langage Java

☐ Mathématiques

☐ Programmation Web

☐ Réseau

☐ Système

☐ Visual Basic.Net

Epreuve facultatif:

☐ Oui

☐ Non

Oral

☐ Anglais

☐ Chinois

☐ Espagnol

☐ Droit

☐ Expression

☐ Gestion

☐ Mathématiques

☐ Réseau

☐ Système

17

```

Imports System.Text.RegularExpressions

Public Class Form3

    Dim count As Integer = 0
    Dim cpt As Integer = 0
    Private modifier As Boolean = False
    Private inscription As Form2
    Private cand As Candidat
    Private nbtours, temps, restant As Integer
    Dim matieres() As String = {"Droit", "Expression", "Gestion", "Mathématiques", "Programmation Web", "Réseau", "Système", "Visual Basic .NET"}
    Dim ma() As String = {"Algorithmique", "Anglais", "Base de données", "Chinois", "Espagnol", "Langage C", "Langage Java", "Programmation Web"}
    Private Sub Form3_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Timer1.Start()
        temps = 90
        restant = 0
        nbtours = 0
        LblNom2.Text = cand.nom
        LblPren2.Text = cand.prenom

        Dim EPREUVESE() As String = {"Algorithmique", "Base de données", "Droit", "Expression", "Gestion", "Langage C", "Langage Java", "Mathématiques", "Programmation Web", "Réseau", "Système", "Visual Basic .NET"}
        Dim preuvesO() As String = {"Anglais", "Chinois", "Espagnol", "Droit", "Expression", "Mathématiques", "Réseau", "Système"}
        GrpFac.Visible = False

    End Sub
    Function getCand()
        Return cand
    End Function
    Sub New(ByRef ins As Form2)

        InitializeComponent()
        Me.modifier = ins.getmodifier()
        Me.inscription = ins
        cand = Me.inscription.getCand()
        If Me.modifier Then
            count = 0
            cpt = 0
            For Each ck As CheckBox In GrpEcrit.Controls
                For Each ecrit As String In cand.ecrits
                    If ck.Text = cand.ecrits(ecrit) Then
                        ck.Checked = True
                        count = count + 1
                    End If
                Next
            Next
            For Each ck As CheckBox In GrpOral.Controls
                For Each orals As String In cand.orals
                    If ck.Text = cand.orals(orals) Then
                        ck.Checked = True
                        cpt = cpt + 1
                    End If
                Next
            Next

            CbxRegion.Text = cand.region
            If cand.fac = "" Then
                RbNon.Checked = True
            Else

```

```

RbOui.Checked=True
CbxFacultatif.SelectedItem=cand.fac
EndIf
Else
cand.ecrits=New List(OfString)
cand.orals=New List(OfString)
EndIf

Me.modifier=False
EndSub

PrivateSubCKEdroit_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKEdroit.CheckedChanged
IfCKEdroit.Checked=TrueThen
CkOdroit.Checked=False
CkOdroit.Enabled=False
    count = count +1
ElseIfCKEdroit.Checked=FalseThen
    count = count -1

CkOdroit.Enabled=True
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCkEexpr_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkEexpr.CheckedChanged
IfCkEexpr.Checked=TrueThen
CkOExpress.Checked=False
CkOExpress.Enabled=False
    count = count +1
ElseIfCkEexpr.Checked=FalseThen
CkOExpress.Enabled=True
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCkEGest_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkEGest.CheckedChanged
IfCkEGest.Checked=TrueThen
CkOGest.Checked=False
CkOGest.Enabled=False
    count = count +1
Else
CkOGest.Enabled=True
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCKEmath_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKEmath.CheckedChanged
IfCKEmath.Checked=TrueThen
CkOMath.Checked=False
CkOMath.Enabled=False
    count = count +1
ElseIfCKEmath.Checked=FalseThen
CkOMath.Enabled=True

```

```

        count = count -1
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCkEResa_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkEResa.CheckedChanged
IfCkEResa.Checked=TrueThen
CkORes.Checked=False
CkORes.Enabled=False
        count = count +1
ElseIfCkEResa.Checked=FalseThen
CkORes.Enabled=True
        count = count -1
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCkESyst_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkESyst.CheckedChanged
IfCkESyst.Checked=TrueThen
CkOSyst.Checked=False
CkOSyst.Enabled=False
        count = count +1
ElseIfCkESyst.Checked=FalseThen
CkOSyst.Enabled=True
        count = count -1
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCkOdroit_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkOdroit.CheckedChanged
IfCkOdroit.Checked=TrueThen
CKEdroit.Checked=False
CKEdroit.Enabled=False
cpt=cpt+1
ElseIfCkOdroit.Checked=FalseThen
CKEdroit.Enabled=True
cpt=cpt-1
EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubCkOExpress_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkOExpress.CheckedChanged
IfCkOExpress.Checked=TrueThen
CkEexpr.Checked=False
CkEexpr.Enabled=False
cpt=cpt+1
ElseIfCkOExpress.Checked=FalseThen
CkEexpr.Enabled=True
cpt=cpt-1
EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubCkOGest_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkOGest.CheckedChanged
IfCkOGest.Checked=TrueThen

```

```

CkEGest.Checked=False
CkEGest.Enabled=False
cpt=cpt+1
ElseIfCkOGest.Checked=FalseThen
CkEGest.Enabled=True
cpt=cpt-1

EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubCkOMath_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkOMath.CheckedChanged
IfCkOMath.Checked=TrueThen
CKEmath.Checked=False
CKEmath.Enabled=False
cpt=cpt+1
ElseIfCkOMath.Checked=FalseThen
CKEmath.Enabled=True
cpt=cpt-1

EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubCkORes_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkORes.CheckedChanged
IfCkORes.Checked=TrueThen
CkEResa.Checked=False
CkEResa.Enabled=False
cpt=cpt+1
ElseIfCkORes.Checked=FalseThen
CkEResa.Enabled=True
cpt=cpt-1

EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubCkOSyst_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCkOSyst.CheckedChanged
IfCkOSyst.Checked=TrueThen
CkESyst.Checked=False
CkESyst.Enabled=False
cpt=cpt+1
ElseIfCkOSyst.Checked=FalseThen
CkESyst.Enabled=True
cpt=cpt-1
EndIf
TxtNbreOrals.Text=cpt.ToString
EndSub

PrivateSubckEAlgo_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesckEAlgo.CheckedChanged
IfckEAlgo.Checked=TrueThen
    count = count +1
ElseIfckEAlgo.Checked=FalseThen
    count = count -1
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

```

```

PrivateSubckEbd_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesckEbd.CheckedChanged
IfckEbd.Checked=TrueThen
    count = count +1
Else
    count = count -1
EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCKELgeC_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKELgeC.CheckedChanged
IfCKELgeC.Checked=TrueThen
    count = count +1
Else
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCKElgeJava_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKElgeJava.CheckedChanged
IfCKElgeJava.Checked=TrueThen
    count = count +1
Else
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCKEPweb_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKEPweb.CheckedChanged
IfCKEPweb.Checked=TrueThen
    count = count +1
Else
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSubCKEVB_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesCKEVB.CheckedChanged
IfCKEVB.Checked=TrueThen
    count = count +1
Else
    count = count -1

EndIf
TxtNbreEcrits.Text=count.ToString
EndSub

PrivateSub TextBox1_TextChanged(sender AsObject, e
AsEventArgs)HandlesTxtNbreEcrits.TextChanged
Ifcount.ToString=4Then

TxtNbreEcrits.Text="limiteatteinte"
ForEach ck AsCheckBoxInGrpEcrit.Controls

Ifck.Checked=FalseThen
ck.Enabled=False

```

```

EndIf
Next

ElseIf count.ToString < 4 Then
    ForEach ck As CheckBox In GrpEcrit.Controls
        If ck.Enabled = False Then
            If ck.Checked = False Then
                ck.Enabled = True
            EndIf
        EndIf
    ForEach ak As CheckBox In GrpOral.Controls
        If ak.Checked = True Then
            If ck.Text = ak.Text Then
                ck.Enabled = False
            EndIf
        EndIf
    Next
Next

EndIf
EndSub

Private Sub CkOAng_CheckedChanged(sender As Object, e As EventArgs) Handles CkOAng.CheckedChanged
    If CkOAng.Checked = True Then
        cpt = cpt + 1
    Else
        cpt = cpt - 1
    EndIf
    TxtNbOrals.Text = cpt.ToString
EndSub

Private Sub CkOChinois_CheckedChanged(sender As Object, e As EventArgs) Handles CkOChinois.CheckedChanged
    If CkOChinois.Checked = True Then
        cpt = cpt + 1
    Else
        cpt = cpt - 1
    EndIf
    TxtNbOrals.Text = cpt.ToString
EndSub

Private Sub CkOesp_CheckedChanged(sender As Object, e As EventArgs) Handles CkOesp.CheckedChanged
    If CkOesp.Checked = True Then
        cpt = cpt + 1
    Else
        cpt = cpt - 1
    EndIf
    TxtNbOrals.Text = cpt.ToString
EndSub

Private Sub TextBox2_TextChanged(sender As Object, e As EventArgs) Handles TxtNbOrals.TextChanged
    If cpt.ToString = 3 Then
        TxtNbOrals.Text = "limite atteinte"
        ForEach ck As CheckBox In GrpOral.Controls
            If ck.Checked = False Then
                ck.Enabled = False
            EndIf
        Next
    EndIf
EndSub

```

```

Next

ElseIf cpt.ToString < 3 Then
    ForEach ck As CheckBox In GrpOral.Controls
        If ck.Enabled = False Then
            If ck.Checked = False Then
                ck.Enabled = True
            EndIf
        EndIf
    ForEach ak As CheckBox In GrpEcrit.Controls
        If ak.Checked = True Then
            If ck.Text = ak.Text Then
                ck.Enabled = False
            EndIf
        EndIf
    Next
EndIf

Next
EndIf

Next
EndIf

If cpt.ToString = 3 And count.ToString = 4 Then
    GrpFac.Visible = True
    CbxFacultatif.Visible = False
Else
    GrpFac.Visible = False
EndIf

EndSub

Private Sub RadioButton1_CheckedChanged(sender As Object, e
As EventArgs) Handles RbOui.CheckedChanged
    Dim l As New List(Of String)
    If RbOui.Checked = True Then
        ForEach ck As CheckBox In GrpEcrit.Controls
            ForEach ak As CheckBox In GrpOral.Controls
                If ck.Checked = False And ak.Checked = False Then
                    If ck.Enabled = False And ak.Enabled = False Then
                        CbxFacultatif.Items.Add(ck.Text)
                        l.Add(ck.Text)
                    EndIf
                EndIf
            Next
        EndIf
    EndIf
    Next

Next

ForEach ak As CheckBox In GrpOral.Controls
    If ak.Checked = False And ak.Enabled = False Then

        ForEach m As String In l
            If ak.Text <> m Then
                CbxFacultatif.Items.Add(ak.Text)
            EndIf
        EndIf
    EndIf
Next

EndIf
Next

```



```

CbxFacultatif.Visible=True

Else
CbxFacultatif.Visible=False
EndIf

GrpOral.Enabled=False
GrpEcrit.Enabled=False
EndSub

PrivateSub RadioButton2_CheckedChanged(sender AsObject, e
AsEventArgs)HandlesRbNon.CheckedChanged
GrpOral.Enabled=True
GrpEcrit.Enabled=True
ForEach ck AsCheckBoxInGrpOral.Controls
Ifck.Enabled=FalseThen
Ifck.Checked=FalseThen
ck.Enabled=True
EndIf
ForEachakAsCheckBoxInGrpEcrit.Controls
Ifak.Checked=TrueThen
Ifck.Text=ak.TextThen
ck.Enabled=False
EndIf
EndIf
Next
EndIf

Next
ForEach ck AsCheckBoxInGrpEcrit.Controls
Ifck.Enabled=FalseThen
Ifck.Checked=FalseThen
ck.Enabled=True
EndIf
ForEachakAsCheckBoxInGrpOral.Controls
Ifak.Checked=TrueThen
Ifck.Text=ak.TextThen
ck.Enabled=False
EndIf
EndIf
Next
EndIf
Next
EndSub

PrivateSub Button1_Click(sender AsObject, e AsEventArgs)HandlesBtnAnnuler.Click
MsgBox("Etes-vous sûr d'annuler l'inscription ?",vbYesNo,"Annulation")
IfvbYesThen
Me.Owner.Owner.Show()

Me.Owner.Close()
Me.Close()
Else
Me.Show()
EndIf
EndSub

PrivateSub Button2_Click(sender AsObject, e AsEventArgs)HandlesBtnValider.Click
DimvalideAsBoolean=True

If count >4Then

```

```

valide =False
MsgBox("le nombre d'épreuves écrites est supérieur à 4")
EndIf
Ifcpt>3Then
valide=False
MsgBox("le nombre d'épreuves orales est supérieur à 3")
EndIf
IfCbRegion.Text=""Then
MsgBox("vous devez choisir une région")
EndIf
Ifvalide=TrueThen

        Timer1.Stop()
ForEach ck AsCheckBoxInGrpEcrit.Controls
Ifck.Checked=TrueThen
cand.ecrits.Insert(0,ck.Text)
EndIf
Next
ForEach ck AsCheckBoxInGrpOral.Controls
Ifck.Checked=TrueThen
cand.orals.Insert(0,ck.Text)
EndIf
Next
cand.region=CbxRegion.Text
IfRbNon.CheckedThen
cand.fac=""
Else
cand.fac=CbxFacultatif.Text
EndIf

DimrecapitulatifAsNew Form4(Me)
recapitulatif.Show(Me)
Me.Hide()

EndIf
EndSub

PrivateSub Timer1_Tick(sender AsObject, e AsEventArgs)Handles Timer1.Tick
nbtours+=1
Ifnbtours\10=restantThen
restant+=1
EndIf
Me.Text="Choix des épreuves et de la région      "+TimeOfDay+"      "+(temps -
restant).ToString()+"s"
If temps -restant=0Then
Me.Owner.Owner.Show()
Me.Owner.Close()
Me.Close()
MsgBox("vous devez choisir vos épreuves et la région de passage en 1min30 !")
EndIf
EndSub
EndClass

```

4. Formulaire 4

Récapitulatif

Données Saisie

Nom Label2 Région: Label19

Prénom Label4

Ville Label6

Code Postal Label8

Age Label12

Adresse Label 10

Epreuves Ecrites:

Epreuves Orales:

Epreuves Facultatif:

Annuler Retour Enregistrer

```

Imports System.IO

Public Class Form4
    Private cand As Candidat
    Private choixpreuves As Form3
    Private bilan As Boolean = False

    Sub New(ByRef choixfaits As Form3)
        InitializeComponent()
        Me.choixpreuves = choixfaits
        Me.cand = choixfaits.getCand()
    EndSub

    Sub New(ByRef modifier As Form5)

        InitializeComponent()
        Me.cand = modifier.getCand
        Me.Show(modifier)
        Me.Owner.Hide()
        MsgBox("La suppression est irréversible ! Ete-vous sur de votre choix?", vbYesNo, "Confirmation
de suppression")
        If vbYes Then

            Module1.listeCandidat.Remove(cand)
            MsgBox("le candidat a bien été supprimé !", MsgBoxStyle.OkOnly)
            Me.Owner.Owner.Show()
            Me.Owner.Close()
            Me.Close()
        Else
            Me.Close()
            Me.Owner.Show()
        EndIf

    EndSub

    Sub New(ByRef bilan As Form7)

        InitializeComponent()
        Me.cand = bilan.getCand()
        Me.bilan = True

    EndSub

    Private Sub Form4_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        LblNom.Text = cand.nom
        LblPren.Text = cand.prenom
        LblVille.Text = cand.ville
        LblCp.Text = cand.cp
        If cand.adress = "" Then
            LblAddress.Visible = False
            Label9.Visible = False
        Else
            LblAddress.Text = cand.adress
            Label9.Visible = True
        EndIf
        LblAge.Text = cand.age
        LblRegion.Text = cand.region
        For Each s As String In cand.ecrits
            LvEcrits.Items.Add(s)
        Next
        For Each s As String In cand.orals

```

```

LvOral.Items.Add(s)
Next
If cand.fac = "" Then
    TxtFac.Visible = False
    Label15.Visible = False
Else
    TxtFac.Text = cand.fac
    Label15.Visible = True
EndIf

If Not File.Exists(c) Then
    Dim d As FileStream
    d = File.Create(C)
d.Close()
EndIf

EndSub

PrivateSub Button1_Click(sender As Object, e As EventArgs) Handles BtnAnnuler.Click
    MsgBox("Etes-vous sûr d'annuler l'inscription ?", vbYesNo, "Annulation")
    If vbYes Then
        Me.Owner.Owner.Owner.Show()
        Me.Owner.Owner.Close()
        Me.Owner.Close()
        Me.Close()
    Else
        Me.Show()
    EndIf
EndSub

PrivateSub Button2_Click(sender As Object, e As EventArgs) Handles BtnEnregistrer.Click
    cand.id += 1
    nbenregistrement += 1
    Using s As StreamWriter = New StreamWriter("num.txt")
        s.WriteLine(cand.id)
    EndUsing
    listeCandidat.Add(cand)
    enregistrer()

    Me.Owner.Owner.Owner.Show()
    Me.Owner.Owner.Close()
    Me.Close()
EndSub

PrivateSub Button3_Click(sender As Object, e As EventArgs) Handles BtnRetour.Click
    If bilan = False Then
        Me.Owner.Owner.Show()
        Me.Owner.Owner.Close()
        Me.Close()
    Else
        Me.Owner.Show()
        Me.Close()
    EndIf
EndSub

EndClass

```

5. Formulaire 5

Windows application window titled "Action sur candidat". The main title of the form is "Modification Ou Suppression".

Form fields and controls:

- Section "Type de recherche :":
 - Radio button "Numéro"
 - Radio button "Identifiant"
- Search area:
 - Label "Identifiant :"
 - Label "N° :"
 - Dropdown menu
- Action buttons:
 - "Supprimer"
 - "Modifier"
 - "Retour à l'accueil"

```
PublicClass Form5
PrivatecandAsCandidat
FunctiongetCand()
Returncand
EndFunction
```

```
PrivateSub Form5_Load(sender AsObject, e AsEventArgs) Handles MyBase.Load
cbxnompren.Items.Clear()
cbxNum.Items.Clear()
ForEachcandAsCandidatInlisteCandidat
cbxnompren.Items.Add(cand.nom&" "&cand.prenom)
cbxNum.Items.Add(cand.id)
Next
cbxNum.Visible = False
cbxnompren.Visible = False
txtnopren.Visible = False
Txtnum.Visible = False
EndSub
```

```
PrivateSubRdBtnNum_CheckedChanged(sender AsObject, e AsEventArgs) HandlesRdBtnNum.CheckedChanged
IfRdBtnNum.Checked = TrueThen
cbxNum.Visible = True
Txtnum.Visible = True
Else
cbxNum.Visible = False
Txtnum.Visible = False
EndIf
EndSub
```

```
PrivateSubRdBtnNomPren_CheckedChanged(sender AsObject, e AsEventArgs)
HandlesRdBtnNomPren.CheckedChanged
IfRdBtnNomPren.Checked = TrueThen
cbxnompren.Visible = True
txtnopren.Visible = True
Else
cbxnompren.Visible = False
txtnopren.Visible = False
EndIf
EndSub
```

```

PrivateSub btnSupp_Click(sender As Object, e As EventArgs) Handles btnSupp.Click
    If RdBtnNomPren.Checked = True Then
        For Each s As Candidat In listeCandidat
            If (s.nom & " " & s.prenom) = cbxnompren.Text Then
                cand = s
                Dim recapitulatif As New Form4(Me)
                Exit For
            End If
        Next

        ElseIf RdBtnNum.Checked = True Then
            For Each s As Candidat In listeCandidat
                If s.id = cand.id Then
                    cand = s
                    Dim recapitulatif As New Form4(Me)
                    Exit For
                End If
            Next
        End If
    End Sub

PrivateSub BtnModif_Click(sender As Object, e As EventArgs) Handles BtnModif.Click
    If RdBtnNomPren.Checked = True Then
        For Each s As Candidat In listeCandidat
            If (s.nom & " " & s.prenom) = cbxnompren.Text Then
                cand = s
                Dim modification As New Form2(Me)
                modification.Show(Me)
                Exit For
            End If
        Next

        ElseIf RdBtnNum.Checked = True Then
            For Each s As Candidat In listeCandidat
                If s.id = cbxNum.Text Then
                    cand = s
                    Dim modification As New Form2(Me)
                    modification.Show(Me)
                    Exit For
                End If
            Next
        End If
    End Sub

PrivateSub Button1_Click(sender As Object, e As EventArgs) Handles BtnRetour.Click
    Me.Owner.Show()
    Me.Close()
End Sub
End Class

```

6. Formulaire 6

```
Imports System.IO
```

```
Public Class Form6
```

```
Private cand As Candidat
```

```
Dim abbreviation As String
```

```
Private Sub Form6_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
Dim ecrits, orals As String
```

```
Dim fac As String = ""
```

```
ForEach cand As Candidat In listeCandidat.OrderBy(Function(x) x.nom).ToList()
```

```
ecrits = (abrev(cand.ecrits(0)) & " "&abrev(cand.ecrits(1)) & " "&abrev(cand.ecrits(2)) & " "&abrev(cand.ecrits(3)))
```

```
orals = (abrev(cand.orals(0)) & " "&abrev(cand.orals(1)) & " "&abrev(cand.orals(2)))
```

```
LstId.Items.Add(cand.id)
```

```
LstNomPren.Items.Add(cand.nom & " "&cand.prenom)
```

```
LstFac.Items.Add(abrev(cand.fac))
```

```
LstEcrits.Items.Add(ecrits)
```

```
LstOral.Items.Add(orals)
```

```
Next
```

```
End Sub
```

```
Function abrev(epreuves As String)
```

```
Dim abbreviation As String = ""
```

```
Select Case epreuves
```

```
Case = "Algorithmique"
```

```
abbreviation = "ALG"
```

```
Exit Select
```

```
Case = "Anglais"
```

```
abbreviation = "ANG"
```

```
Exit Select
```

```
Case = "Base de données"
```

```
abbreviation = "BDD"
```

```
Exit Select
```

```
Case = "Chinois"
```



```

abreviation = "CHI"
ExitSelect
Case = "Droit"
abreviation = "DRO"
ExitSelect
Case = "Espagnol"
abreviation = "ESP"
ExitSelect
Case = "Expression"
abreviation = "EXP"
ExitSelect
Case = "Gestion"
abreviation = "GES"
ExitSelect
Case = "Langage C"
abreviation = "LGC"
ExitSelect
Case = "Langage Java"
abreviation = "JAV"
ExitSelect
Case = "Mathématiques"
abreviation = "MAT"
ExitSelect
Case = "Programmation Web"
abreviation = "WEB"
ExitSelect
Case = "Réseau"
abreviation = "RES"
ExitSelect
Case = "Système"
abreviation = "SYS"
ExitSelect
Case = "Visual Basic .NET"
abreviation = "VBN"
ExitSelect
EndSelect

Returnabreviation
EndFunction

PrivateSubBtnRetour_Click(sender AsObject, e AsEventArgs) HandlesBtnRetour.Click
Me.Owner.Show()
Me.Close()
EndSub

Private Sub LstId_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
LstId.SelectedIndexChanged
    With LstId
        LstEcrits.SelectedIndex = .SelectedIndex
        LstNomPren.SelectedIndex = .SelectedIndex
        LstOral.SelectedIndex = .SelectedIndex
        LstFac.SelectedIndex = .SelectedIndex

    End With

End Sub
EndClass

```

7. Formulaire 7



```
PublicClass Form7
Private cand As Candidat
PrivateSub Form7_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    TxtNbInscr.Text = nbenregistrement
    Dim matieres() As String = {"Algorithmique", "Anglais", "Base de données", "Chinois", "Droit",
    "Espagnol", "Expression", "Gestion", "Langage C", "Langage Java", "Mathématiques",
    "Programmation Web", "Réseau", "Système", "Visual Basic .NET"}
    Dim region() As String = {"Auvergne", "Bordelais", "Bourgogne", "Bretagne", "Corse", "Nord",
    "Normandie", "Paris", "Poitou", "Roussillon"}
    CbxCand.Items.Clear()
    CbxMat.Items.Clear()
    For Each cand As Candidat In listeCandidat
        CbxCand.Items.Add(cand.nom & " " & cand.prenom)

    Next
    For Each s As String In matieres
        CbxMat.Items.Add(s)
    Next
    CbxMat.Visible = False
    CbxCand.Visible = False
    LblCand.Visible = False
    LblMatiere.Visible = False

EndSub

PrivateSub RadioButton1_CheckedChanged(sender As Object, e As EventArgs)
    Handles RbIndividuel.CheckedChanged
    If RbIndividuel.Checked Then
        CbxCand.Visible = True
        LblCand.Visible = True
    Else
        CbxCand.Visible = False
        LblCand.Visible = False
    EndIf
EndSub
```

```

PrivateSub RadioButton2_CheckedChanged(sender AsObject, e AsEventArgs)
HandlesRbMatiere.CheckedChanged
IfRbMatiere.CheckedThen
Lb1Matiere.Visible = True
CbxMat.Visible = True
Else
Lb1Matiere.Visible = False
CbxMat.Visible = False
EndIf
EndSub

PrivateSub Button1_Click(sender AsObject, e AsEventArgs) HandlesBtnRetour.Click
Me.Owner.Show()
Me.Close()
EndSub
Functiongetcand()
Returncand
EndFunction
PrivateSub Button2_Click(sender AsObject, e AsEventArgs) HandlesBtnAfficher.Click
Dim bool AsBoolean = False

IfRbIndividuel.CheckedThen
IfCbxCand.Text<>""Then
ForEachcandiAsCandidatInlisteCandidat
If (candi.nom&" "&candi.prenom) = CbxCand.TextThen
bool = True
cand = candi
DimrecapitulatifAsNew Form4(Me)
Me.Hide()
recapitulatif.BtnAnnuler.Visible = False
recapitulatif.BtnRetour.Visible = False
recapitulatif.Show(Me)
ExitFor
EndIf
Next
Else
MsgBox("candidatintrouvable")
EndIf
ElseIfRbMatiere.CheckedThen
IfCbxMat.Text<>""Then
DimbilanMatiereAsNew Form8(Me)

ForEachcandAsCandidatInlisteCandidat
ForiAsInteger = 0 To 3
Ifcand.ecrits(i) = CbxMat.TextThen
bilanMatiere.DataGridView1.Rows.Add(cand.id, cand.nom, cand.prenom,
cand.region, cand.age, True, False, False)

EndIf
Next
ForiAsInteger = 0 To 2
Ifcand.oral(i) = CbxMat.TextThen
bilanMatiere.DataGridView1.Rows.Add(cand.id, cand.nom, cand.prenom,
cand.region, cand.age, False, True, False)

EndIf
Next
Ifcand.fac = CbxMat.TextThen
bilanMatiere.DataGridView1.Rows.Add(cand.id, cand.nom, cand.prenom,
cand.region, cand.age, False, False, True)
EndIf
Next
bilanMatiere.Show(Me)

```

```
Me.Hide()  
Else  
MsgBox("introuvable!")  
EndIf  
EndIf  
EndSub  
EndClass
```

8. Formulaire 8

| | Id | Nom | Prénom | Région | Age | Ecrit | Oral | Facultatif |
|---|----|-----|--------|--------|-----|--------------------------|--------------------------|--------------------------|
| * | | | | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

```
PublicClassForm8
PrivatebilanAs Form7

SubNew(bilanAs Form7)

' Cet appel est requis par le concepteur.
InitializeComponent()

' Ajoutez une initialisation quelconque après l'appel InitializeComponent().
Me.bilan = bilan

EndSub
PrivateSub Button1_Click(sender AsObject, e AsEventArgs) Handles Button1.Click
Me.Owner.Show()
Me.Close()
EndSub
EndClass
```

9. Module

```
Imports System.IO
Imports System.Runtime.Serialization.Formatters.Binary

Public Module Module1
    Public a As StreamReader
    Public b As String
    Public c As String = "sauvegarde.txt"
    Public arret As Boolean

    <Serializable(>>
    Public Structure Candidat
        Public id, cp, age As Integer
        Public nom, prenom, adress, ville, fac, region As String
        Public ecrits, orals As List(Of String)

    End Structure

    Public nbEnregistrement As Integer
    Public listeCandidat As New List(Of Candidat)
    Private cand As Candidat

    Public Sub enregistrer()
        Dim stream As FileStream = File.Create("vec.txt")
        Dim formatter As New BinaryFormatter
        formatter.Serialize(stream, listeCandidat)
        stream.Close()
    End Sub

    Public Sub lecture()
        If File.Exists("vec.txt") Then
            Dim stream As FileStream = File.OpenRead("vec.txt")
            Dim formatter As New BinaryFormatter

            listeCandidat = formatter.Deserialize(Stream)
            stream.Close()
        End If
    End Sub

End Module
```

V – Conclusion

Le projet Vb.net nous a permis de découvrir et comprendre au mieux un nouveau langage. Concernant les différentes améliorations applicables a notre solution :

On peut d'abord améliorer notre code en factorisant mieux certaines parties du code, ce qui permettraient une lecture plus fluide et rapide ainsi qu'un code plus performant et optimisé. Le code du formulaire trois aurait pu et aurait du être beaucoup plus court que ce que l'on propose. Cependant, on s'est rendu compte de cela trop tard et avons décidé de le laisser tel quel.

On pourrait aussi améliorer la relation entre les différents formulaires.

Enfin , il y a quelques bugs a corriger comme par exemple le formulaire de choix des épreuves , on est obligé de finir avec le choix de l'épreuve oral pour que la groupbox contenant l'épreuve facultatif s'affiche.

Avec un peu plus de temps et un ordinateur non tombé en panne je pense que nous aurions pu palier à ces quelques problèmes. Toutefois ce projet nous a beaucoup enseigné et s'est révéler très intéressant à suivre.