



Compte rendu

**Développement en C# du Service Windows
qui clôturera automatiquement
les fiches des visiteurs.**

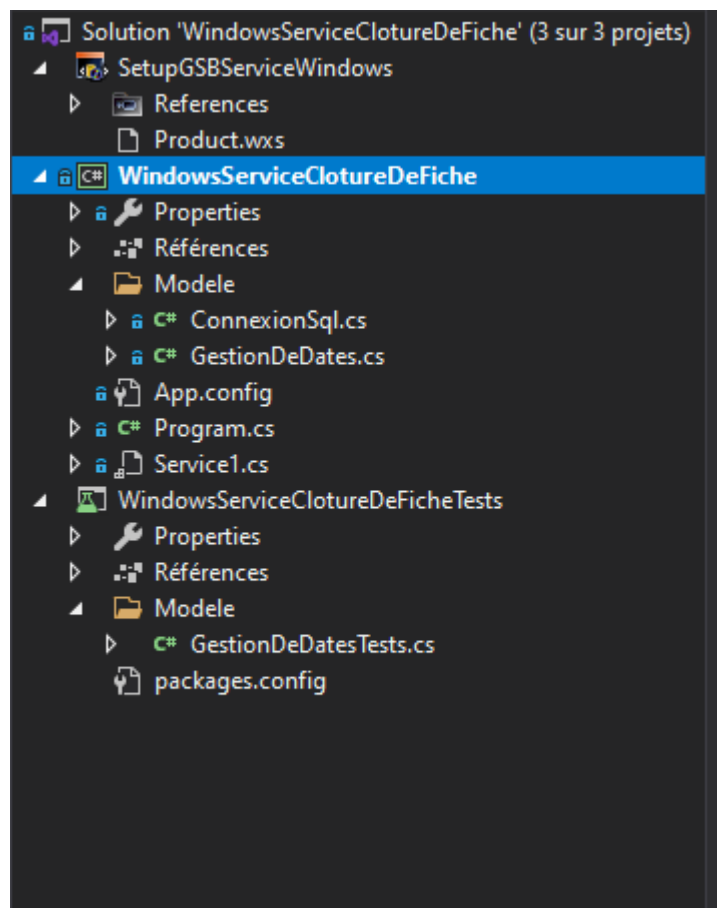
Table des matières

1. Logiciels utilisés	3
2. Arborescence des fichiers.....	3
3. Création du code source	4
3.1. Classe de connexion.....	4
3.2. Classe de gestion de dates.....	6
3.3. Main Service Windows	8
4. Installateur avec WIX.....	10
5. Tests unitaires.....	10
6. Gestion de version	12
7. Compétences	12

1. Logiciels utilisés

- IDE Visual Studio 2019
- Phpmyadmin
- .net Framework 4.7.2
- Extension Wix Toolset pour visual studio
- Sandcastle Help File Builder (Documentation technique)
- Windows 64bit
- Git et Github

2. Arborescence des fichiers



3. Création du code source

3.1. Classe de connexion

```
public class ConnexionSql
{
    // propriétés
    private bool finCurseur=true; // fin du curseur atteinte
    private static string chaîneConnection = "Database=gsb_frais;Data Source=localhost;User Id=root;password=";
    private MySqlConnection connexion; // chaîne de connexion
    private MySqlCommand command; // envoi de la requête à la base de données
    private MySqlDataReader reader; // gestion du curseur
    private static ConnexionSql connexionGSB = null;

    // constructeur
    1 référence | it-fernandeztony, il y a 14 jours | 1 auteur, 1 modification
    private ConnexionSql()
    {
        this.connexion = new MySqlConnection(chaîneConnection);
        this.connexion.Open();
    }

    /// <summary>
    /// Créer une connexion
    /// </summary>
    /// <returns>Retourne un objet de type ConnexionSql</returns>
    1 référence | it-fernandeztony, il y a 1 heure | 1 auteur, 1 modification
    public static ConnexionSql GetConnexionSql()
    {
        connexionGSB = null;
        connexionGSB = new ConnexionSql();

        return connexionGSB;
    }
}
```

```
/// <summary>
/// Execution d'une requête select
/// </summary>
/// <param name="chaineRequete">La requête</param>
0 références | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public void ReqSelect(string chaineRequete)
{
    this.command = new MySqlCommand(chaineRequete, this.connexion);
    this.reader = this.command.ExecuteReader();
    this.finCurseur = false;
    this.Suivant();
}

/// <summary>
/// Execution d'une requête update
/// </summary>
/// <param name="chaineRequete">La requête</param>
2 références | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public void ReqUpdate(string chaineRequete)
{
    this.command = new MySqlCommand(chaineRequete, this.connexion);
    this.command.ExecuteNonQuery();
    this.finCurseur = true;
}

/// <summary>
/// Récupération d'un champ
/// </summary>
/// <param name="nomChamp">Nom du champ</param>
/// <returns>Retourne le nom d'un champ</returns>
0 références | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public Object Champ(string nomChamp)
{
    return this.reader[nomChamp];
}
```

```

    /// <summary>
    /// Passage à la ligne suivante du curseur
    /// </summary>
    1 référence | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
    public void Suivant()
    {
        if (!this.finCurseur)
        {
            this.finCurseur = !this.reader.Read();
        }
    }

    /// <summary>
    /// Test de la fin du curseur
    /// </summary>
    /// <returns>Retourne vrai ou faux</returns>
    0 références | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
    public Boolean Fin()
    {
        return this.finCurseur;
    }

    /// <summary>
    /// Fermeture de la connexion
    /// </summary>
    1 référence | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
    public void Close()
    {
        this.connexion.Close();
    }
}
}

```

3.2. Classe de gestion de dates

```

    /// <summary>
    /// Classe de gestion de date
    /// </summary>
    10 références | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
    public abstract class GestionDeDates
    {
        /// <summary>
        /// Retourne le mois précédent le mois actuel
        /// </summary>
        /// <returns>Retourne un mois sous la forme -MM- de type string</returns>
        3 références | 0/1 ayant réussi | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
        public static string GetMoisPrecedent()
        {
            DateTime dateDuJour = DateTime.Today; // Permet d'avoir la date d'aujourd'hui

            return GetMoisPrecedent(dateDuJour); // Appel de la méthode GetMoisPrecedant avec paramètre
        }

        /// <summary>
        /// Retourne le mois précédent la date entrée en paramètre
        /// </summary>
        /// <param name="date">Une date</param>
        /// <returns>Retourne un mois sous la forme -MM- de type string</returns>
        2 références | 0/1 ayant réussi | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
        public static string GetMoisPrecedent(DateTime date)
        {
            return date.AddMonths(-1).ToString("MM");
        }
    }
}

```

```

/// <summary>
/// Retourne le mois suivant du mois actuel
/// </summary>
/// <returns>Retourne un mois sous la forme -MM- de type string</returns>
1 référence | 0/1 ayant réussi | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public static string GetMoisSuivant()
{
    return GetMoisSuivant(DateTime.Today); // Appel de la méthode GetMoisSuivant qui prend un paramètre
}

/// <summary>
/// Retourne le mois suivant de la date entrée en paramètre
/// </summary>
/// <param name="date">Une date</param>
/// <returns>Retourne un mois sous la forme -MM- de type string</returns>
2 références | 0/1 ayant réussi | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public static string GetMoisSuivant(DateTime date)
{
    return date.AddMonths(1).ToString("MM");
}

/// <summary>
/// Retourne vrai si la date du jour se trouve entre les deux jours entrés en paramètre
/// </summary>
/// <param name="jour1">Le jour le plus petit</param>
/// <param name="jour2">Le jour le plus grand</param>
/// <returns>Retourne vrai ou faux</returns>
3 références | 0/1 ayant réussi | it-fernandeztony, Il y a 1 heure | 1 auteur, 1 modification
public static Boolean Entre(int jour1, int jour2)
{
    return Entre(jour1, jour2, DateTime.Today);
}

```

```

/// <summary>
/// Retourne vrai si la date entrée en paramètre se trouve entre les deux jours
/// </summary>
/// <param name="jour1">Le jour le plus petit</param>
/// <param name="jour2">Le jour le plus grand</param>
/// <param name="date">Une date</param>
/// <returns>Retourne vrai ou faux</returns>
2 références | 0/1 ayant réussi | it-fernandeztony, Il y a 2 heures | 1 auteur, 1 modification
public static Boolean Entre(int jour1, int jour2, DateTime date)
{
    int jour = date.Day;

    if (jour1 <= jour & jour <= jour2)
    {
        return true;
    }
    else
    {
        return false;
    }
}

```

3.3. Main Service Windows

```
0 références | it-fernandeztony, il y a 2 heures | 1 auteur, 3 modifications
static class Program
{
    private static System.Timers.Timer unTimer;
    /// <summary>
    /// Point d'entrée principal de l'application.
    /// </summary>
    0 références | it-fernandeztony, il y a 8 jours | 1 auteur, 2 modifications
    static void Main()
    {
        ServiceBase[] ServicesToRun;
        ServicesToRun = new ServiceBase[]
        {
            new Service1()
        };
        ServiceBase.Run(ServicesToRun);
        SetTimer();
        Console.ReadLine();
        unTimer.Stop();
        unTimer.Dispose();
    }
}
```

Procédure création d'un Timer

```
/// <summary>
/// Création du timer
/// </summary>
1 référence | it-fernandeztony, il y a 8 jours | 1 auteur, 1 modification
private static void SetTimer()
{
    // Créer un timer avec un interval de 2 secondes
    unTimer = new System.Timers.Timer(2000);
    // Ajout d'un événement lorsque le timer est écoulé
    unTimer.Elapsed += OnTimedEvent;
    unTimer.AutoReset = true;
    unTimer.Enabled = true;
}
```


Procédure événement fin de Timer

```
/// <summary>
/// Événement fin de timer
/// </summary>
/// <param name="source"></param>
/// <param name="e"></param>
1 référence | it-fernandeztony, Il y a 3 heures | 1 auteur, 2 modifications
private static void OnTimedEvent(Object source, ElapsedEventArgs e)
{
    // Connexion à la bdd et récupération du curseur:

    ConnexionSql crs = ConnexionSql.GetConnexionSql();

    // On vérifie qu'on est bien entre le 1 et le 10 du mois:
    if (GestionDeDates.Entre(1, 10) == true)
    {
        // Récupération des fiches du mois précédent et maj de celles-ci:
        // Récupération du mois précédent et son année
        string moisPrecedent = GestionDeDates.GetMoisPrecedent();
        string annee = DateTime.Today.AddMonths(-1).ToString("yyyy");
        string mois = annee + moisPrecedent;
        crs.ReqUpdate("UPDATE fichefrais SET idetat='CL' WHERE mois =" + mois + " AND idetat='CR'");
    }
    // Si on est après le 20 du mois:
    if (GestionDeDates.Entre(20, 30) == true)
    {
        ;
        // Récupération des fiches du mois précédent et maj de celles-ci:
        string moisPrecedent = GestionDeDates.GetMoisPrecedent();
        string annee = DateTime.Today.AddMonths(-1).ToString("yyyy");
        string mois = annee + moisPrecedent;

        crs.ReqUpdate("UPDATE fichefrais SET idetat='RB' WHERE mois = " + mois + " AND idetat='VA'");
    }
    crs.Close();
}
```










4. Installateur avec WIX

```
<Wix xmlns="http://schemas.microsoft.com/wix/2006/wi">
  <Product Id="*" Name="SetupGSBServiceWindows" Language="1033" Version="1.0.0.0" Manufacturer="GSB" UpgradeCode="fcef2b98-6667-4f5f-ab22-008377180cf0">
    <Package InstallerVersion="200" Compressed="yes" InstallScope="perMachine" Manufacturer="Fernandez Tony" />
    <Property Id="ARPCONTACT" Value="Fernandez Tony" />
    <MajorUpgrade DowngradeErrorMessage="A newer version of [ProductName] is already installed." />
    <MediaTemplate />
    <Feature Id="ProductFeature" Title="SetupGSBServiceWindows" Level="1">
      <ComponentGroupRef Id="ProductComponents" />
    </Feature>
  </Product>
  <Fragment>
    <Directory Id="TARGETDIR" Name="SourceDir">
      <Directory Id="ProgramFilesFolder">
        <Directory Id="INSTALLFOLDER" Name="SetupGSBServiceWindows" />
      </Directory>
    </Directory>
  </Fragment>
  <Fragment>
    <ComponentGroup Id="ProductComponents" Directory="INSTALLFOLDER">
      <!-- Réaliser à l'aide de https://github.com/Rob79/How-to-create-a-Windows-Service-MSI-Installer-Using-WiX/blob/master/WindowsService.Installer/Product.wxs -->
      <Component Id="ProductComponent">
        <File
          Id="WindowsServiceClotureDeFicheEXE"
          Name="WindowsServiceClotureDeFiche.exe"
          DiskId="1"
          Source="$(var.WindowsServiceClotureDeFiche.TargetPath)"
          Vital="yes"
          KeyPath="yes" />
        <ServiceInstall Id="ServiceInstaller" Type="ownProcess" Vital="yes"
          Name="GSBFicheDeFrais" DisplayName="GSBFicheDeFrais"
          Description="Modifie les fiches de frais" Start="auto"
          Account="LocalSystem" ErrorControl="normal"
          Arguments="/start GSBFicheDeFrais"
          Interactive="no" />
      </Component>
    </ComponentGroup>
  </Fragment>
  <ServiceControl Id="StartService" Stop="both" Start="install" Remove="uninstall"
    Name="GSBFicheDeFrais" Wait="yes" />
</Wix>
```

Source de l'aide : Source : <https://github.com/Rob79/How-to-create-a-Windows-Service-MSI-Installer-Using-WiX/blob/master/WindowsService.Installer/Product.wxs>

5. Tests unitaires

Un projet nommé « WindowsServiceClotureDeFicheTests » a été créé pour ces tests.

Explorateur de tests				Rechercher dans l'Explorateur de tests	
Test	Durée	Caractéris...	Message d'erreur	Récapitulatif du groupe	
WindowsServiceClotureDeFicheTests (..) 	10 ms			WindowsServiceClotureDeFicheTests	
WindowsServiceClotureDeFicheTests	10 ms			Tests dans le groupe	
GestionDeDatesTests (6) 	10 ms			Durée totale: 10 ms	
EntreTest 	9 ms			Résultats	
EntreTest1 	1 ms			 6 Réussite	
GetMoisPrecedentTest 	< 1 ms				
GetMoisPrecedentTest1 	< 1 ms				
GetMoisSuivantTest 	< 1 ms				
GetMoisSuivantTest1 	< 1 ms				

```
[TestClass()]
0 références
public class GestionDeDatesTests
{
    [TestMethod()]
    ✓ | 0 références
    public void GetMoisPrecedentTest()
    {
        string expected = "02";
        string actual = GestionDeDates.GetMoisPrecedent();
        Assert.AreEqual(expected, actual);
    }

    [TestMethod()]
    ✓ | 0 références
    public void GetMoisPrecedentTest1()
    {
        string expected = "01";
        DateTime date = DateTime.Parse("6/02/2021");
        string actual = GestionDeDates.GetMoisPrecedent(date);
        Assert.AreEqual(expected, actual);
    }

    [TestMethod()]
    ✓ | 0 références
    public void GetMoisSuivantTest()
    {
        string expected = "04";
        string actual = GestionDeDates.GetMoisSuivant();
        Assert.AreEqual(expected, actual);
    }

    [TestMethod()]
    ✓ | 0 références
    public void GetMoisSuivantTest1()
    {
        string expected = "02";
        DateTime date = DateTime.Parse("6/01/2021");
        string actual = GestionDeDates.GetMoisSuivant(date);
        Assert.AreEqual(expected, actual);
    }

    [TestMethod()]
    ✓ | 0 références
    public void EntreTest()
    {
        bool expected = true; // false;
        int jour1 = 15; // 2;
        int jour2 = 30; // 15;
        bool actual = GestionDeDates.Entre(jour1, jour2);
        Assert.AreEqual(expected, actual);
    }

    [TestMethod()]
    ✓ | 0 références
    public void EntreTest1()
    {
        bool expected = false; // true;
        int jour1 = 7; // 2;
        int jour2 = 15;
        DateTime date = DateTime.Parse("6/01/2021");
        bool actual = GestionDeDates.Entre(jour1, jour2, date);
        Assert.AreEqual(expected, actual);
    }
}
```

6. Gestion de version

Git :

Répertoire de travail	5
Index du commit	
master Projet-GSB2-commit/master Suppression d'un commit et création d'un nouveau (mise en ligne d'un mot de passe par erreur) Ajout du main, ajout d'un installateur.	 it-fernandeztony Il y a 9 jours e04a7f1
Création des méthode de classe GestionDeDates avec tests unitaires.	 it-fernandeztony Il y a 14 jours 782c26b
Modification de la classe Connexion SQL. Création de la classe abstraite GestionDeDates.	 it-fernandeztony Il y a 14 jours 4cbc41f
Démarrage. Ajout de la classe connexionSQL.	 it-fernandeztony Il y a 14 jours 77f9f12
Projet-GSB2-commit/main Initial commit	 it-fernandeztony Il y a 15 jours 20eca1a

Github :

Commit

<https://github.com/it-fernandeztony/Projet-GSB2-commit.git>

7. Compétences

A4.1.1 , Proposition d'une solution applicative.

A4.1.2 , Conception ou adaptation de l'interface utilisateur d'une solution applicative.

A4.1.3 , Conception ou adaptation d'une base de données.

A4.1.6 , Gestion d'environnements de développement et de test.

A4.1.7 , Développement, utilisation ou adaptation de composants logiciels.

A4.1.8 , Réalisation des tests nécessaires à la validation d'éléments adaptés ou développés.

A4.1.9 , Rédaction d'une documentation technique.

A4.2.1 , Analyse et correction d'un dysfonctionnement, d'un problème de qualité de ...

A4.2.2 , Adaptation d'une solution applicative aux évolutions de ses composants.

A4.2.3 , Réalisation des tests nécessaires à la mise en production d'éléments mis à jour.

A4.2.4 , Mise à jour d'une documentation technique.