Compte Rendu de Projet :

Table des matières

[I. Introduction : 2](#_Toc40430969)

[II. Réalisation du projet : 3](#_Toc40430970)

[a) Tâches de l’étudiant : 3](#_Toc40430971)

[b) Interconnexion du réseau : 3](#_Toc40430972)

[c) Communication : 3](#_Toc40430973)

[d) Informations de connexion au serveur BDD : 4](#_Toc40430974)

[e) Code de connexion à la base de données : 4](#_Toc40430975)

[III. Étude du Wi-Fi : 6](#_Toc40430976)

[a) L’antenne du routeur : 7](#_Toc40430977)

[IV. Réalisation de l’application de supervision : 9](#_Toc40430978)

[a) Lancer une nouvelle partie : 9](#_Toc40430979)

[b) Superviser : 10](#_Toc40430980)

[c) Envoyer un indice : 12](#_Toc40430981)

[d) Finaliser la partie : 13](#_Toc40430982)

[e) Classer les équipes : 14](#_Toc40430983)

# Introduction :

#### Rappel du cahier des charges :

Afin de superviser la partie durant le déroulement de celle-ci, une application de supervision est à fournir.

L’installation de celle-ci sera sur un ordinateur situé en dehors de la salle de l’escape game.

L’application de supervision doit respecter le cahier des charges suivant :

* Création d’une partie : Le superviseur saisi le nom de l’équipe et lance manuellement la partie. Un chronomètre d’une durée d’une heure doit ensuite se lancer automatiquement.
* Visualiser la partie : Depuis l’application, le superviseur doit avoir accès aux images provenant de la caméra de surveillance installée au sein de la salle du jeu.
* Interagir avec les joueurs : Le superviseur a la possibilité d’envoyer un message écrit à l’afficheur situé dans la salle du jeu. Ce message peut être prédéfini ou être nouvellement saisi.
* Connaitre les résultats : A la fin de la partie, qu’elle soit gagnée ou perdue, le superviseur peut connaître le score de l’équipe actuelle. De plus, cette équipe doit pouvoir être comparée avec d’autres équipes. Un taux de réussite en pourcentage sera calculé et affiché.

La communication entre les différents équipements locaux nécessaires à l’escape game se fera grâce à une intercommunication locale basée sur le protocole IP et gérée par un routeur Wi-Fi.

# Réalisation du projet :

## Tâches de l’étudiant :

Au sein du projet, les parties me concernant sont :

* Le développement de l’application de supervision en respectant le cahier des charges.
* L’établissement et la mise en place de l’intercommunication réseaux entre les différents matériels.

Le développement de l’application doit être pour un ordinateur sous Windows. J’ai ainsi choisi le langage C# et l’IDE Visual Studio comme environnement de développement. Le choix de ces deux derniers m’était libre.

La communication réseau doit être en Wi-Fi. L’afficheur LCD ne comportant pas de carte Wi-Fi, je vais ajouter un adaptateur RS232 / Wi-Fi de type SOLLAE CSW-H85K présent en nombre suffisant au sein de la section.

## Interconnexion du réseau :

L’intégralité des matériels ci-dessus doivent être adressés en IP statiques.

## Communication :

La communication réseau entre les différents matériels se fera par le Wi-Fi en utilisant le protocole TCP/IP.

La communication entre l’adaptateur SOLLAE RS232/Wi-Fi et l’afficheur LCD se fera grâce au protocole RS232.

Enfin, la communication entre la base de données hébergeant les indices et les équipes avec l’application de supervision se fera par l’intermédiaire de localhost (127.0.0.1). En effet, la base de données MySQL sera installée localement sur le même poste que l’application de supervision.

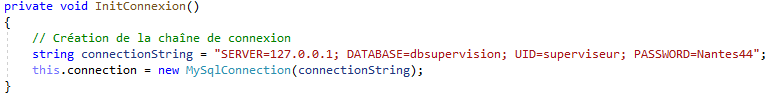
## Informations de connexion au serveur BDD :

Pour se connecter au serveur de base de données hébergé en local sur l’ordinateur, les informations suivantes sont requises :

* + Adresse IP : **127.0.0.1**
  + Base de données : **dbsupervision**
  + Utilisateur : **superviseur**
  + Mot de passe : **Nantes44**

## Code de connexion à la base de données :

Afin que l’application puisse communiquée avec la base de données, il est nécessaire d’employé le code suivant (C#) :



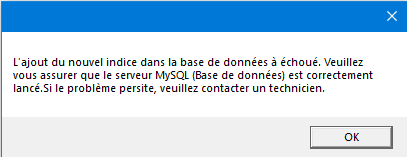
La méthode d’initialisation de connexion ci-dessus permet de renseigner l’adresse du serveur MySQL, l’utilisateur et le mot de passe nécessaire pour la connexion avec la base de données. Je définie ensuite l’élément *connection* étant un type *MySqlConnection()* (déclarer au préalable).

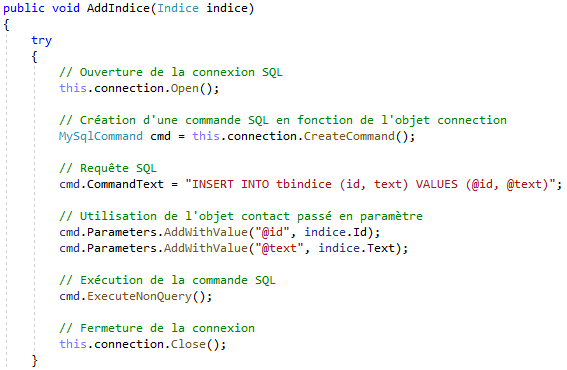
La méthode suivante permet quant à elle l’exécution de la requête SQL depuis l’application. Cette méthode est identique qu’il s’agisse de l’envoi de données ou de la récupération.

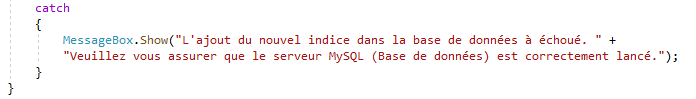
L’exemple suivant permet l’ajout d’un indice dans la base de données, donc un envoi depuis l’application.

L’utilisation des objets passés en paramètres varie en fonction des paramètres renseignés dans la requête SQL.

En cas de problème, il est judicieux de mettre en place un message d’information en cas de mauvais envoi. Celui-ci s’effectue donc avec la condition « try » / « catch ». On essaye ainsi l’exécution de la requête dans le « try ». Si cette action est impossible, on entre dans le « catch ». L’élément « catch » contient un élément de type *MessageBox.Show* qui permet d’afficher, dans une nouvelle vignette, du texte à l’utilisateur.







L’ouverture de la connexion SQL par la méthode *Open()* exécute la chaine de connexion.

A l’inverse, ma méthode *Close()* ferme la connexion établie avec la base de données.

# Étude du Wi-Fi :

Dans l’objectif d’intégrer les différents domaines scolaires au sein du projet, il m’est demandé de réaliser une étude du réseau Wi-Fi que je vais devoir mettre en place.

La référence du routeur Wi-Fi auquel sera connecté : l’ordinateur de supervision, la caméra IP, l’adaptateur SOLLAE (afficheur LCD géant) et le Raspberry (sous-système médaillons) est Netgear WRN612.

Je vais donc présenter ci-dessous le descriptif technique de l’antenne intégrée à ce matériel :

* Normes Wi-Fi prises en charge *: IEEE 802.11n version 2.0, IEEE 802.11n, IEEE 802.11g, IEEE 802.11b 2.4 GHz*
* Intervalles de fréquences : *2.412–2.462 GHz (US), 2.412–2.472 GHz (Europe ETSI)*

Afin de limiter les interférences dues aux autres ondes possiblement émises, ci-dessous un tableau d’écartement conseillé par le constructeur. L’écart est compris entre l’appareil émettant les interférences et l’enveloppe externe du routeur Wi-Fi.

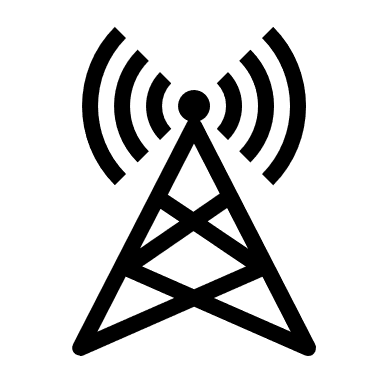
|  |  |
| --- | --- |
| ***Périphérique :*** | ***Distance minimale conseillée :*** |
| Four à micro-ondes | *9 mètres* |
| Moniteur pour bébé (analogique) | *6 mètres* |
| Moniteur pour bébé (numérique) | *12 mètres* |
| Téléphone filaire (analogique) | *6 mètres* |
| Téléphone filaire (numérique) | *9 mètres* |
| Périphériques Bluetooth | *6 mètres* |
| Périphériques ZigBee (électroménager connecté) | *6 mètres* |

En principe, aucun de ces éléments ne devrait se trouver dans le champ des ondes Wi-Fi. Cependant, il est important de prendre en compte les différentes perturbations provoquées par la salle et sa configuration.

Un plan de la salle a donc été établi, celui-ci devrait être relativement proche de la réalité :



*Caméra IP*



*Afficheur LCD*

*4m*

***Routeur Wi-Fi***

*<< Salle de jeu >>*



*<< Salle de supervision >>*

*Sous-système médaillons*

*2m*

*5m*

## L’antenne du routeur :

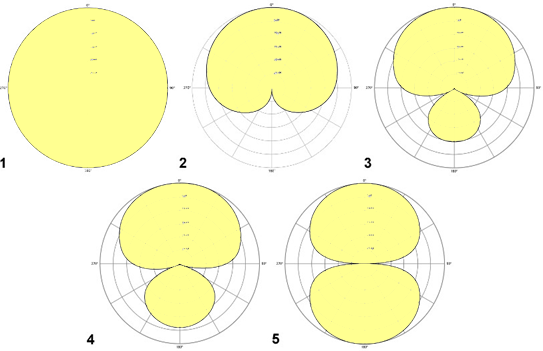
L’antenne intégrée à ce routeur est de type **ANT2511AC**.

Elle dispose de caractéristiques plus détaillées que celles du routeur. Après plusieurs recherches sur l’antenne individuelle, les informations techniques sont recueillies dans le tableau ci-dessous :

|  |  |
| --- | --- |
| Mode Wi-Fi : | 802.11 a/b/g/n/ac |
| Plage de fréquences : | 2,4 à 2,5 GHz et 4,9 à 5,85 GHz |
| Gain : | 2,5 dBi (2,4 GHz) et 4 dBi (5 GHz) |
| Puissance d’émission : | 20 dBm |
| Rayon d’ajustement : | Horizontal 360° Vertical 75° (2.4 GHz) et Horizontal 360° Vertical 22,5° (5 GHz) |
| Polarisation : | Linéaire ; Verticale |
| Connecteurs : | Prise RP-SMA (SMA inversé mâle) |
| Direction : | Omnidirectionnelle. |

Premièrement, nous allons étudier la directivité de l’antenne. Il s’agit de la portée d’émission des ces ondes (et parallèlement de réception). Comme pour les microphones, il existe plusieurs types de directivités.

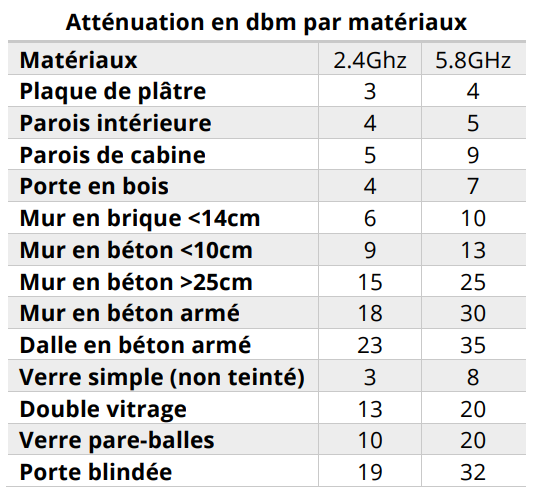
Voici les principaux :



|  |  |
| --- | --- |
| **1** | *Omnidirectionnel* |
| **2** | *Cardioïde* |
| **3** | *Supercardioïde* |
| **4** | *Hypercardioïde* |
| **5** | *Bidirectionnel* |

Ces diagrammes nous permettent de déterminer l’endroit idéal pour placer l’antenne. Notre antenne étant omnidirectionnelle, il serait plus judicieux de la placée au centre de la pièce.

Deux plages de fréquences sont disponibles avec cette antenne, le 2,4 GHz ou le 5 GHz. Cependant, celle-ci ayant destination dans un bâtiment public (mairie), il est probable que d’autres équipements fonctionnent sur le 5 GHz (alarme intrusion, vidéosurveillance, alarme incendie…). De plus, tous les appareils informatiques n’acceptent pas le 5 GHz. Le choix de la plage de fréquence se porte ainsi sur le 2,4 GHz pour des raisons de perturbations et de compatibilité.

Nous allons désormais chercher à savoir si tous les appareils ayant besoin du Wi-Fi le recevront correctement. Pour ce faire, il est nécessaire d’utiliser les faits ci-dessous :

*Formule de calcul d’atténuation en fonction de la distance :*

**92,45+20\*LOG10(*Freq en GHz*)+20\*LOG10(*Dist en km*)**

|  |
| --- |
| Interprétation du bilan : |
| -10 à -77 Excellent |
| -78 à -86 Bon |
| -87 à -92 Faible |
| <-93 Nul |

Sur le plan de la salle représenté en page précédente, les trois points (sous-système médaillons, caméra IP et afficheur LCD) ont besoin d’une connectivité suffisante au Wi-Fi.

Afin de valider cela, nous allons calculer leur atténuation comme il suit :

#### Caméra IP :

Atténuation en l’air sec : 92,45+20\*LOG10(**2,4**)+20\*LOG10(**0,002**) = **46 dBm**

|  |  |
| --- | --- |
| Source | 20 dBm |
| 2m d’air | -46 dBm |
| Mur en brique | -6 dBm |
| Bilan | -32 dBm |
| Qualité du signal | Excellent |

#### Afficheur LCD :

Atténuation en l’air sec : 92,45+20\*LOG10(**2,4**)+20\*LOG10(**0,005**) = **54 dBm**

|  |  |
| --- | --- |
| Source | 20 dBm |
| 5m d’air | -54 dBm |
| Mur en brique | -6 dBm |
| Bilan | -40 dBm |
| Qualité du signal | Excellent |

#### Sous-système médaillons :

Atténuation en l’air sec : 92,45+20\*LOG10(**2,4**)+20\*LOG10(**0,003**) = **49 dBm**

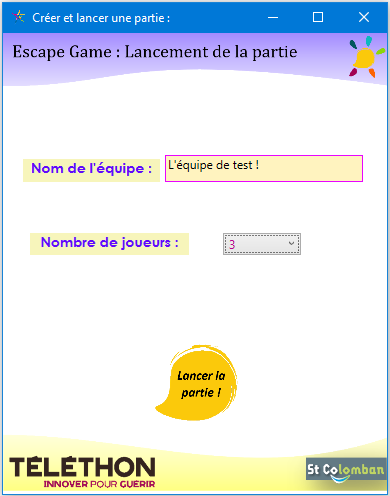
|  |  |
| --- | --- |
| Source | 20 dBm |
| 3m d’air | -49 dBm |
| Mur en brique | -6 dBm |
| Bilan | -35 dBm |
| Qualité du signal | Excellent |

# Réalisation de l’application de supervision :

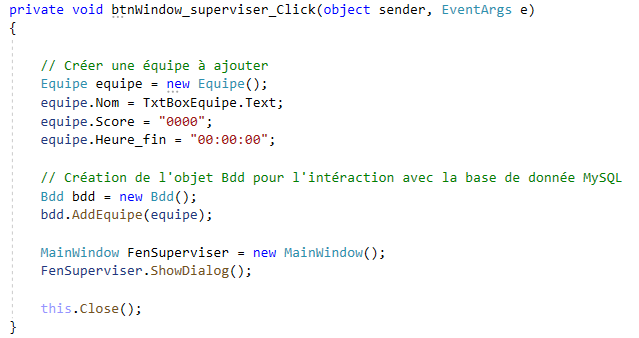
Afin que le superviseur de la partie puisse visualiser et aiguiller l’équipe qui joue, il est nécessaire d’avoir une application dédiée qui communique entre les éléments d’interactions utilisateur et superviseur.

## Lancer une nouvelle partie :

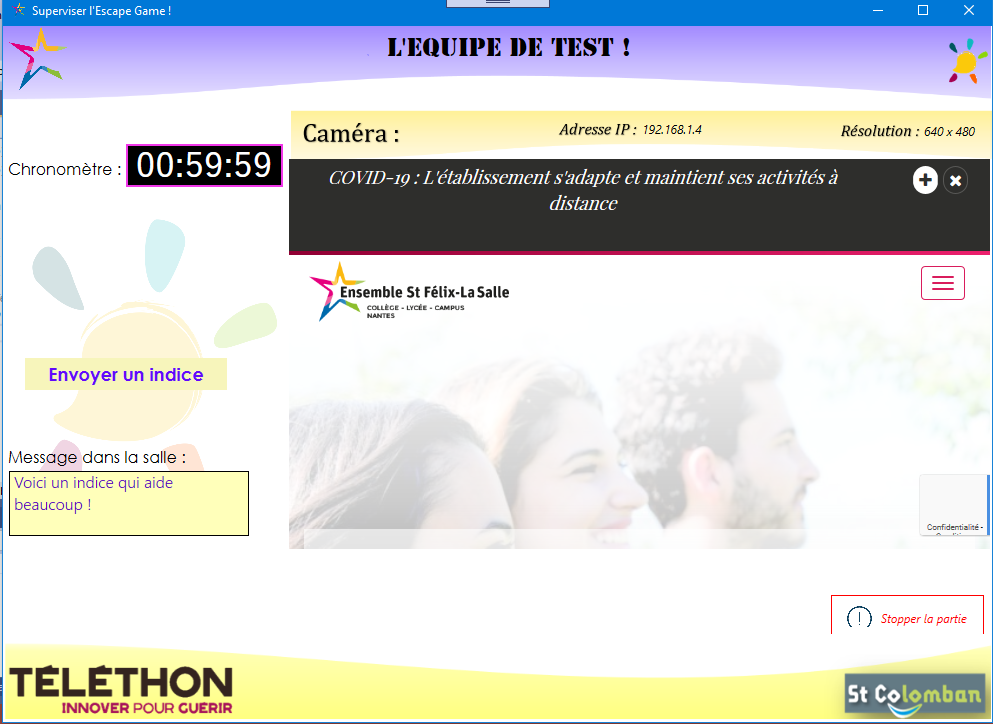
Dès le lancement de l’application par double-clique sur l’icône, cette page s’affiche :





Le superviseur est invité à saisir le nom choisi par l’équipe dans le champ correspondant. Ensuite, le clic sur « **Lancer la partie !** » permet d’enregistrer l’équipe dans la base de données et d’appeler la fenêtre principale de l’application. Pour ce faire, j’utilise le code suivant :

## Superviser :

La fenêtre principale de l’application est celle de supervision. Depuis celle-ci, on peut visualiser l’image en direct provenant de la caméra IP (située dans la salle de l’escape game), appeler la fenêtre d’envoi d’indice afin d’envoyer un indice à l’afficheur. Le superviseur a également la possibilité de surveiller le chronomètre.

*\*Image de la caméra IP\**

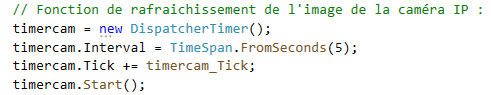
Premièrement, j’utilise l’élément *WebBrowser* pour récupérer l’image de la caméra IP :

L’image de la caméra D-Link est fournie par l’intermédiaire de son adresse IP. Afin de pouvoir l’afficher dans l’application, l’utilisation de l’élément *WebBrowser* est la plus adaptée. Il s’agit en effet d’un navigateur web simplifié et allégé fourni par Microsoft et facilement intégrable dans un code en XAML.



L’image fournie étant fixe, il est nécessaire d’utiliser une fonction de rafraichissement pour que celle-ci soit fluide :

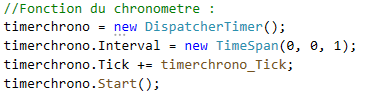


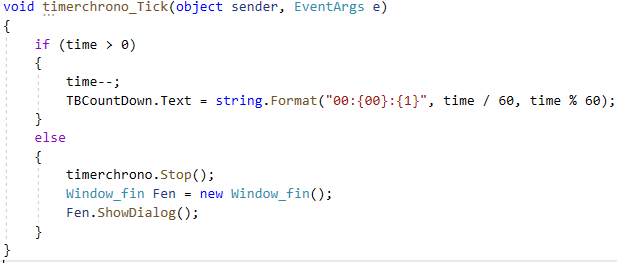


Dans l’exemple précédent, le rafraîchissement de l’image s’effectuait toutes les cinq secondes. Pour une utilisation optimale, je programme celui-ci toute les secondes.

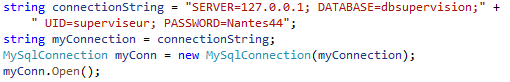
En second lieu, le chronomètre utilise également la minuterie *DispatcherTimer*. Cette minuterie fiable permet son intégration dans de nombreux cas. Basée sur l’intervalle, je l’utilise à deux répétitions au sein de l’application de supervision. Voici le code qui permet de faire fonctionner le chronomètre de la partie et qui y met fin lorsque celui-ci arrive à son terme :

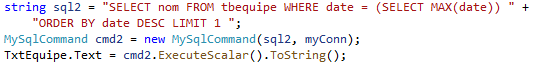




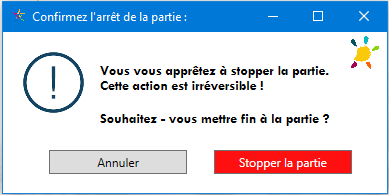


Afin de permettre la récupération du nom de l’équipe et son inscription en haut de la fenêtre principale, j’utilise la communication avec la base de données suivante :





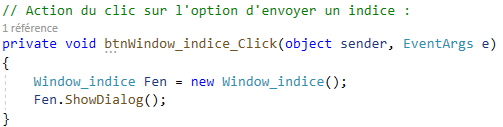
Enfin, si le superviseur souhaite, pour quelconques raisons, mettre fin prématurément à la partie, il suffit de cliquer sur l’option « **Stopper la partie** ». Le clic sur celle-ci ouvre la fenêtre suivante :



L’option « **Annuler** » ferme la fenêtre ci-dessus et reviens à la fenêtre de supervision sans avoir le moindre effet sur la partie en cours. L’option « Stopper la partie » met quant à elle fin à la partie en appelant la fenêtre de fin de partie.

Ensuite, le bouton « **Envoyer un indice** » appel une nouvelle fenêtre qui elle permet de sélectionner un indice prédéfini ou d’en saisir un nouveau. Se référer à la partie « Envoi d’indice ».

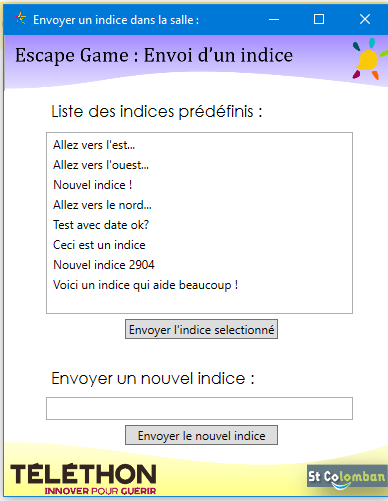
Ci-dessous le code qui est appelé lors du clic sur ce bouton :



## Envoyer un indice :

Le code qui permet d’envoyer un indice sur une adresse IP et ensuite décodé par l’adaptateur Wi-Fi/RS232 n’est actuellement pas présent dans ce rapport. En effet, celui-ci ne pourra se faire avant un retour en classe et l’aide d’un des professeurs. Je ne dispose pas du code ni de la trame requise. Cependant, l’ajout de ce code dans l’ensemble de l’application se fera dans un temps réduis.

Deux options sont alors possibles, l’envoi d’un indice existant en le sélectionnant et l’envoyant. Mais également la création d’un nouvel indice, son enregistrement dans la base de données et l’envoi à l’afficheur. Ce nouvel indice est désormais disponible dans les indices prédéfinis après une réouverture de la fenêtre.



## Finaliser la partie :

La fin de la partie pouvant être provoquée par trois systèmes (la fermeture volontaire depuis le bouton « Stopper la partie », le gain de celle-ci par la réception de l’ordre de fin de partie ou l’échec de celle-ci par l’écoulement du chronomètre), une fenêtre de fin de partie est nécessaire.



## Classer les équipes :

En fin de partie, il était demandé d’afficher un classement des équipes précédentes triées par leur score décroissant. Pour réaliser celui-ci, j’ai choisi de créer une nouvelle *Form* qui permet d’intégrer un élément tableau de type *ListView*. Celui-ci se rempli avec les éléments récupérés par la base de données. Ci-dessous le code qui permet de lister les équipes :



Après l’appel depuis la fenêtre de fin de partie, voici la fenêtre de classement :

