Hofmann Jonny

Documentation technique

Dernière modification : mardi, 3 novembre 2020

Division Technique du CEJEF

ILBSYS WPF



**Table des matières**

[Introduction 3](#_Toc55322820)

[1 Projet 3](#_Toc55322821)

[2 Fonctionnement 3](#_Toc55322822)

[2.1 Schéma 3](#_Toc55322823)

[2.2 Explication 3](#_Toc55322824)

[3 Esquisse 4](#_Toc55322825)

[4 Diagrammes 4](#_Toc55322826)

[4.1 Use case 4](#_Toc55322827)

[4.2 Diagramme de séquence 5](#_Toc55322828)

[4.3 Diagramme de classe 5](#_Toc55322829)

[5 Classes et méthodes 6](#_Toc55322830)

[5.1 Utilities 6](#_Toc55322831)

[5.1.1 Variables membre 6](#_Toc55322832)

[5.1.2 Fonction GetAllServers 6](#_Toc55322833)

[5.1.3 Fonction AddServer 6](#_Toc55322834)

[5.1.4 Fonction GetSelectedServerIndex 6](#_Toc55322835)

[5.1.5 Fonction SetSelectedServerIndex 6](#_Toc55322836)

[5.1.6 Fonction GetSelectedServer 7](#_Toc55322837)

[5.1.7 Fonction DeleteServer 7](#_Toc55322838)

[5.1.8 Fonction GetAllHostsAsync 7](#_Toc55322839)

[5.1.9 Fonction SetCurrentHost 7](#_Toc55322840)

[5.2 InfluxDB 8](#_Toc55322841)

[5.2.1 Variables membre 8](#_Toc55322842)

[5.2.2 Fonction SetCurrentServerAddress 8](#_Toc55322843)

[5.2.3 Fonction SetCurrentServerName 8](#_Toc55322844)

[5.2.4 Fonction SetCurrentHost 8](#_Toc55322845)

[5.2.5 Fonction GetRamUsage 8](#_Toc55322846)

[5.2.6 Fonction GetCPUUsage 9](#_Toc55322847)

[5.2.7 Fonction GetUptime 9](#_Toc55322848)

[5.2.8 Fonction GetCPUUsageAverage24h 9](#_Toc55322849)

[5.2.9 Fonction GetRAMUsage24h 9](#_Toc55322850)

[5.2.10 Fonction GetAllHostsAsync 9](#_Toc55322851)

[5.3 Server 10](#_Toc55322852)

[5.3.1 Variables membre 10](#_Toc55322853)

[5.3.2 Constructeur 10](#_Toc55322854)

[5.4 Host 10](#_Toc55322855)

[5.4.1 Variables membre 10](#_Toc55322856)

[5.4.2 Constructeur 10](#_Toc55322857)

[5.5 MainWindow 10](#_Toc55322858)

[5.5.1 Constructeur 11](#_Toc55322859)

[5.5.2 Fonction UpdateHostList 11](#_Toc55322860)

[5.5.3 Fonction cbServers\_SelectionChanged 11](#_Toc55322861)

[5.5.4 Fonction Btn\_AddServer\_Click 11](#_Toc55322862)

[5.5.5 Fonction PopulateServerList 11](#_Toc55322863)

[5.5.6 Fonction UpdateInfoDataGrid 11](#_Toc55322864)

[5.5.7 Fonction dgHosts\_SelectionChanged 11](#_Toc55322865)

[5.6 Info 12](#_Toc55322866)

[5.6.1 Variables membre 12](#_Toc55322867)

[6 Protocol de tests 12](#_Toc55322868)

[6.1 Vérification ajout d’un serveur 12](#_Toc55322869)

[6.2 Sélection d’hôte 12](#_Toc55322870)

[6.3 Informations affichées 12](#_Toc55322871)

[7 Améliorations envisageables 13](#_Toc55322872)

[7.1 Affichage 13](#_Toc55322873)

[7.1.1 Beauté 13](#_Toc55322874)

[7.2 Gestion serveur 13](#_Toc55322875)

[7.2.1 Modification et suppression 13](#_Toc55322876)

[7.2.2 Sauvegarde locale 13](#_Toc55322877)

[7.3 Graphiques 13](#_Toc55322878)

[7.4 Code 13](#_Toc55322879)

[7.4.1 Utilisation des classes 13](#_Toc55322880)

[8 Versionning 13](#_Toc55322881)

[9 Sources et liens utiles 13](#_Toc55322882)

|  |  |
| --- | --- |
| ILBSYS WPF | Microsoft WPF Samples | How Not To Code |
| Application C# |

# Introduction

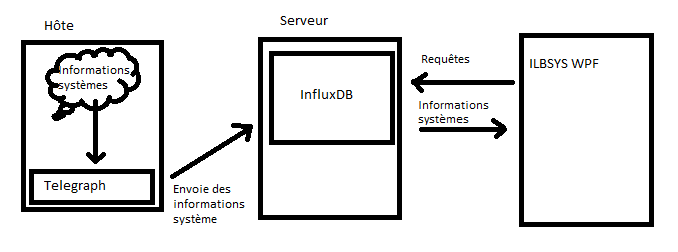
Dans le cadre du module 4 nous devons réaliser une application utilisant le WPF.

# Projet

Comme projet j’ai choisi de faire une application semblable à celle faite en android ce qui est une application qui accède à la base de données InfluxDB que j’ai sur mon serveur pour afficher des statistiques du poste physique sur lequel tournent tous les services de chez moi.

# Fonctionnement

## Schéma



## Explication

Si un haute installe le logiciel Telegraph, il peut envoyer régulièrement toutes ses informations sur le la base de données InfluxDB qui est spécialisée dans le stockage de données en rapport du temps tels que des capteurs de température, statistiques systèmes ou autre. InfluxDB n’est pas une base de données relationnel, pour cela, aucun MCD sera fourni.

Le serveur InfluxDB de test est chez moi sur mon serveur personnel et l’hôte surveillé est ce même serveur. La redirection des ports a déjà été effectuée pour le projet du module 6.

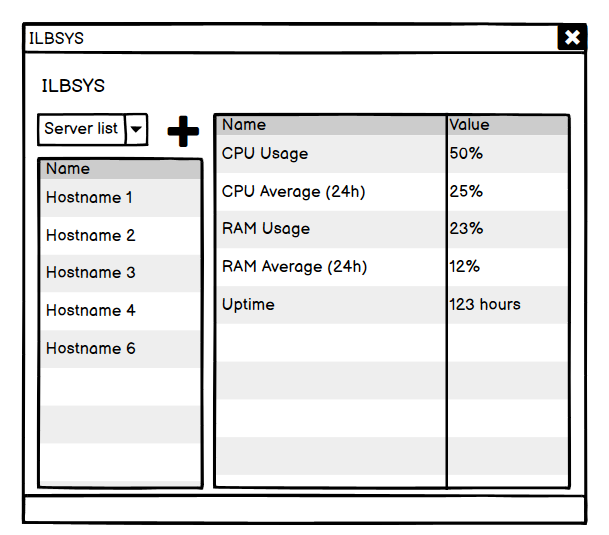
Dans l’application, une liste de serveurs peut être gérée, cette liste de serveurs contient les adresses de serveur qui ont une instance de InfluxDB d’installé.

Les hôtes sont des postes quelconques qui ont Telegraph d’installé et paramétré pour stocker les informations sur un serveur InfluxDB.

Au moment qu’un serveur est sélectionné dans la liste déroulante, la liste des hôtes qui possèdent des données stockées dans sur sa base de donnée seront affichés dans un datagrid.

Lors d’un clique sur un élément du datagrid, les informations de cet hôte sont chargée et affichée.

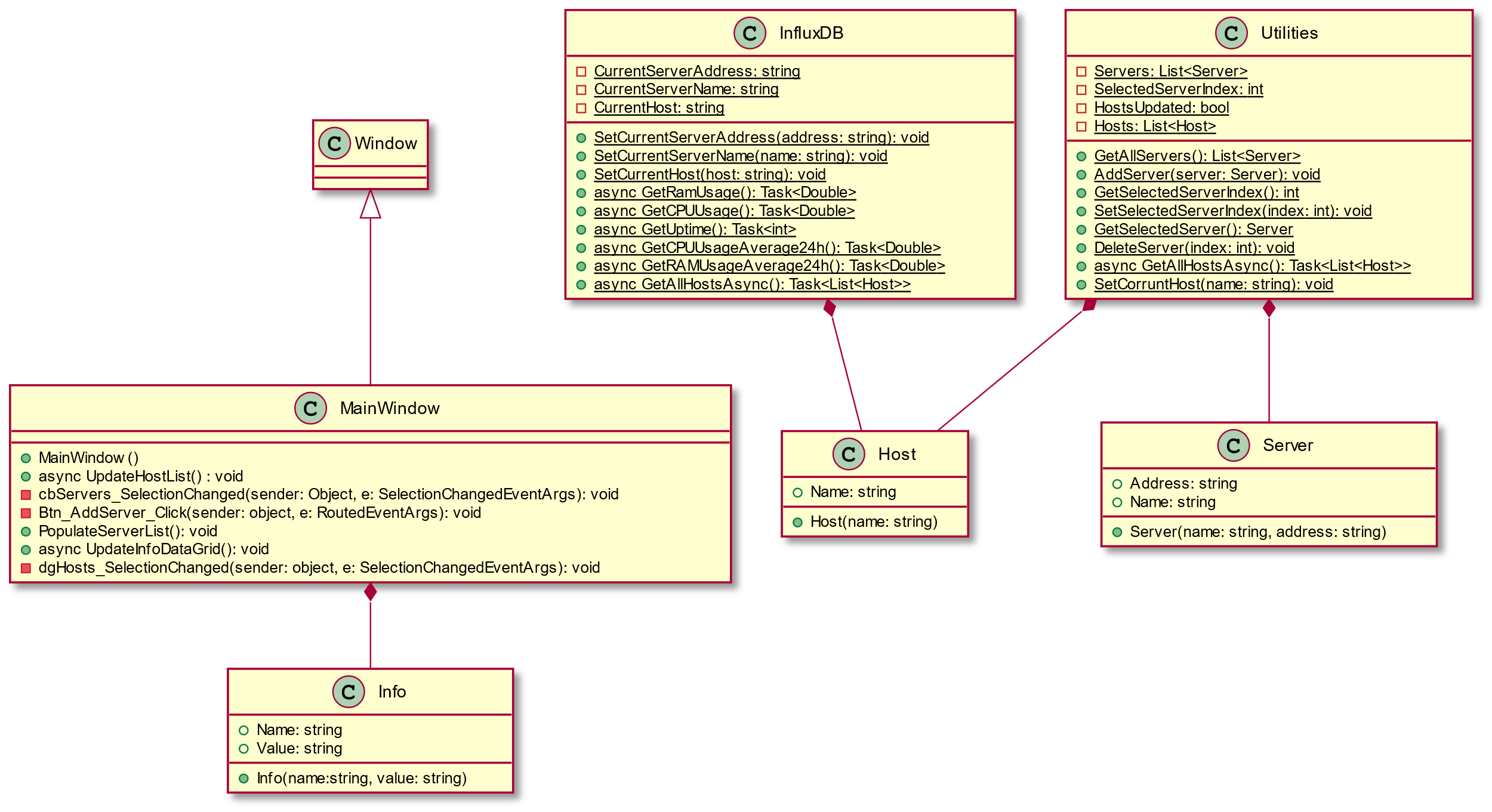
# Esquisse



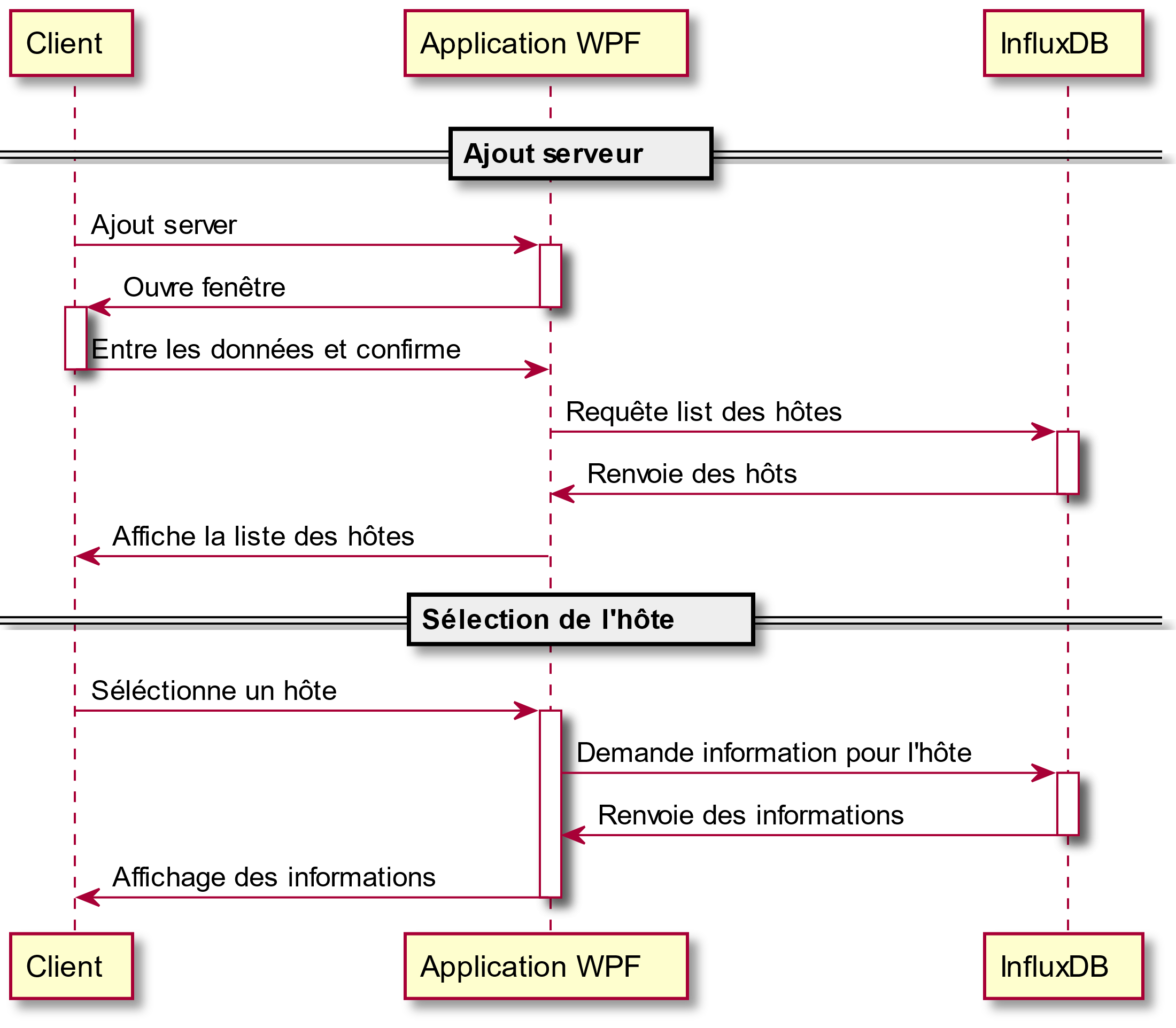
# Diagrammes

Tous les diagrammes ont été créer avec l’outil PlantUML qui permet de créer beaucoup de sortes de diagrammes différents à partir de fichier text brut simple. Ceci rend la création de diagramme rapide et uniforme, il existe des outils en ligne et des extensions pour divers editeurs de code ou de texte.

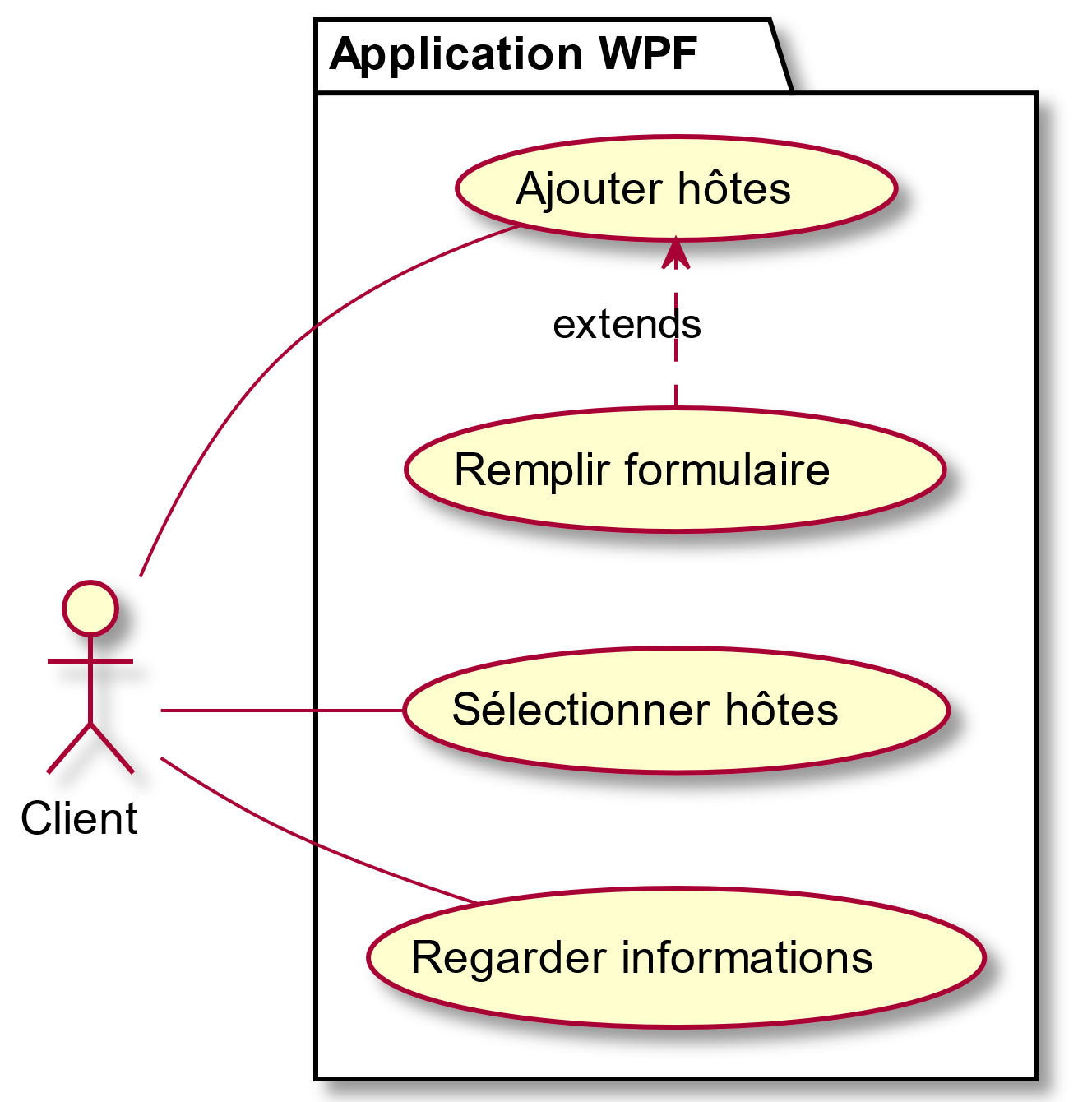
## Use case



## Diagramme de séquence



## Diagramme de classe



# Classes et méthodes

## Utilities

Utilities est une classe static qui met à disposition à tout le programmes des fonctions utilitaires qui servent principalement à gérer globalement les serveurs et les hôtes.

### Variables membre

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| List<Server> | Servers | Liste de tous les serveurs du programme |
| Int | SelectedServerIndex | Index du serveur sélectionné dans la liste des serveurs |
| Bool | HostsUpdated | Si la liste des hôtes est à jour |
| List<Host> | Hosts | Liste de tous les hôtes d’un serveur |

### Fonction GetAllServers

**Description :**

Retourne la liste des serveurs *Servers.*

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| List<Server> | Tous les serveurs du programme |

### Fonction AddServer

**Description :**

Ajoute un nouveau Server à la liste des serveurs

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Server | Server | Server à ajouter |

### Fonction GetSelectedServerIndex

**Description :**

Retourne l’index du serveur actuellement sélectionné

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Int | Index du Server actuellement sélectionné |

### Fonction SetSelectedServerIndex

**Decription :**

Set l’index du Server sélectionné

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Int | Index | Index du Server à sélectionner |

### Fonction GetSelectedServer

**Decription :**

Retourne le Server qui est actuellement sélectionné

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Server | Server qui est actuellement sélectionné |

### Fonction DeleteServer

**Description :**

Supprime un Server de la liste

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Server | server | Server à supprimer de la liste |

### Fonction GetAllHostsAsync

**Description :**

Recherche sur InfluxDB tous les hôtes du serveur actuel et les retourne dès que InfluxDB a répondu. Cette fonction est en asynchrone.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<List<Host>> | Liste de tous les hôtes sur le serveur actuel |

### Fonction SetCurrentHost

**Description :**

Change l’hôte qui est actuellement sélectionné

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | Name | Nom de l’hôte à choisir |

## InfluxDB

### Variables membre

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | CurrentServerAddress | Adresse du serveur actuel |
| String | CurrentServerName | Nom du serveur actuel |
| String | CurrentHost | Nom de l’hôte actuel |

### Fonction SetCurrentServerAddress

**Description :**

Change l’adresse du serveur actuel

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | address | Adresse du serveur à appliquer |

### Fonction SetCurrentServerName

**Description :**

Change le nom du serveur actuel

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | name | Nom du serveur |

### Fonction SetCurrentHost

**Description :**

Change l’hôte actuel

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | host | Nom de l’hôte |

### Fonction GetRamUsage

**Description :**

Fait une requête sur la base de données pour récupérer l’utilisation de RAM actuelle pour l’hôte qui est couramment choisi. Ceci est une fonction asynchrone.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<Double> | Utilisation de la RAM en pourcentage |

### Fonction GetCPUUsage

**Description :**

Fait une requête sur la base de données InfluxDB pour récupérer l’utilisation actuelle du CPU pour l’hôte qui est actuellement sélectionné.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<Double> | Utilisation du CPU en pourcentage |

### Fonction GetUptime

**Description :**

Fait une requête sur la base de données InfluxDB pour récupérer le temps en ligne de l’hôte actuel en seconde. Cette fonction est asynchrone.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<int> | Temps en ligne de l’hôte actuel en seconde |

### Fonction GetCPUUsageAverage24h

**Description :**

Fait une requête sur la base de données InfluxDB pour récupérer les utilisations de CPU des 24 dernières heures et en calcule une moyenne avant de la retourner. Cette fonction est asynchrone.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<double> | Moyenne de l’utilisation du CPU de l’hôte actuel sur les 24 dernières heures |

### Fonction GetRAMUsage24h

**Description :**

Fait une requête sur la base de données InfluxDB pour récupérer les utilisations de RAM des 24 dernières heures et en calcule une moyenne avant de la retourner. Cette fonction est asynchrone.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<double> | Moyenne d’utilisation de la RAM de l’hôte actuel sur les 24 dernières heures |

### Fonction GetAllHostsAsync

**Description :**

Fait une requête sur la base de données InfluxDB pour récupérer tous les hôtes qui sont sur le serveur actuel.

**Retour :**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Task<List<Host>> | Liste de tous les hôtes sur le serveur actuel |

## Server

Server est une classe qui est la pour stocker les données d’un serveur de manière plus facile.

### Variables membre

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | Address | Address du serveur |
| String | Name | Nom du serveur |

### Constructeur

**Description :**

Construit l’objet Server et remplie les variable membre.

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | name | Nom du serveur |
| String | address | Adresse du serveur |

## Host

Est une classe pour stocker de manière plus agréable un hôte.

### Variables membre

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | Name | Nom de l’hôte |

### Constructeur

**Description :**

Construit l’objet Host et lui affecte la variable membre

**Paramètre :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | name | Nom de l’hôte |

## MainWindow

Est la classe qui contient toutes les méthodes qui interagissent avec l’interface de la page principale.

### Constructeur

Construit la classe MainWindow et fais les premières initialisations nécessaires pour remplir les différents champs.

### Fonction UpdateHostList

**Description :**

Vide et ensuite rempli la liste des hôtes qui sont retourné de InfluxDB. Cette fonction est asynchrone.

### Fonction cbServers\_SelectionChanged

**Description :**

Cette fonction est appelée quand un élément est sélectionné dans la liste des serveurs. Elle met ensuite à jour le serveur actuel et demande la mise à jour de la liste des hôtes.

**Paramètres :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Object | sender | Elément qui a lancé l’événement |
| SelectionChangedEventArgs | e | Arguments propre au changement de sélection (pas utilisée et générée automatiquement) |

### Fonction Btn\_AddServer\_Click

**Description :**

Cette fonction est appelée quand le bouton pour ajouter un serveur est cliqué. Elle ouvre la fenêtre d’ajout de serveur.

**Paramètres :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Object | sender | Elément qui a lancé l’événement |
| RoutedEventArgs | e | Arguments propre au click, généré automatiquement |

### Fonction PopulateServerList

**Description :**

Récupère la liste des serveurs et l’ajoute à la liste de serveurs après l’avoir vidée. Si la liste était auparavant vide, met le premier serveur en tant que serveur actif.

### Fonction UpdateInfoDataGrid

**Description :**

Récupère toutes la valeurs qu’il faut pour le datagrid d’infos de l’hôte, les mets en forme et les ajoute au datagrid. Ceci est une fonction asynchrone parce qu’elle doit attendre les réponses de la classe InfluxDB qui elle doit attendre sur les réponses du serveur InfluxDB.

### Fonction dgHosts\_SelectionChanged

**Descirption :**

Cette fonction est appelée quand on clique sur un hôte dans la liste des hôtes. Elle change l’hôte active et lance la mise à jour du datagrid d’informations.

**Paramètres :**

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| Object | sender | Elément qui a lancé l’événement |
| SelectionChangedEventArgs | e | Arguments propre au changement de sélection (pas utilisée et générée automatiquement) |

## Info

Cette classe contient les informations d’une « Info » qui est une ligne dans le datagrid d’informations sur un hôte. Une liste de celle-là est utilisée comme source pour le datagrid d’information.

### Variables membre

|  |  |  |
| --- | --- | --- |
| **Type** | **Nom** | **Description** |
| String | Name | Nom de la ligne d’information |
| String | Value | Valeur qui sera à afficher (Avec les unités) |

# Protocol de tests

Pour s’assurer du bon fonctionnement du programme, des protocoles de tests doivent être effectuée.

## Vérification ajout d’un serveur

Pour vérifier que l’ajout d’un serveur fonctionne correctement, il faut :

* Cliquer sur le +
* Entrer des données
* Cliquer sur OK
* Recliquer sur le +
* Entrer les données
* Cliquer sur OK
* Vérifier que les deux ont correctement été ajouté

## Sélection d’hôte

Pour vérifier que la sélection de l’hôte fonctionne correctement fonctionne correctement il faut :

* Avoir sélectionné un serveur fonctionnel
* Vérifier que au moins un hôte soit affiché dans la liste
* Cliquer sur un hôte
* Vérifier que les données sont chargées correctement dans la liste des informations

## Informations affichées

Pour vérifier les informations qui sont affichées dans le datagrid il faut connaître les données d’un hôte. Dans mon cas je vais lire les statistiques sur l’interface web de mon serveur. Pour les vérifier il faut :

* Ajouter un serveur fonctionnel
* Cliquer sur un hôte dont on connait les statistiques
* Vérifier que les données affichées correspondent aux données qu’on a d’une source sûre

# Améliorations envisageables

## Affichage

### Beauté

* Rendre les datagrid plus élégants

## Gestion serveur

### Modification et suppression

Pouvoir modifier et supprimer des serveurs seraient une fonctionnalité importante pour une version finale du programme.

### Sauvegarde locale

Sauvegarder la liste des serveurs en locale serait bien pour ne pas devoir les ajouter à chaque fois qu’on démarre le programme.

## Graphiques

Avoir des graphiques pour afficher l’évolution de l’utilisation de RAM et du CPU serait intéressante pour analyser et trouver des mal fonctions d’un hôte.

## Code

### Utilisation des classes

Les classes Server et Hôte ne sont pas toujours utilisées quand ce serait utile, ceci serait à changer et demanderai du refactoring de code. Malheureusement je l’ai réalisé trop tard pour le changer.

# Versionning

Tout le versionning se fait sur mon GitHub. Les sources du programme, la documentation avec ses sources se situent dessus.

[ILikeBananas/ILBSYS\_WPF](https://github.com/ILikeBananas/ILBSYS_WPF)

# Sources et liens utiles

|  |  |
| --- | --- |
| **Description** | **Lien** |
| Telegraph | [influxdata.com/time-series-platform/telegraf/](https://www.influxdata.com/time-series-platform/telegraf/) |
| InfluxDB | [influxdata.com](https://www.influxdata.com/) |
| Client InfluxDB pour C# par MikaelGRA | [MikaelGRA/InfluxDB.Client](https://github.com/MikaelGRA/InfluxDB.Client) |
| PlantUML | [plantuml.com](https://plantuml.com/) |