



Module SIN22

Intégration et validation de la solution

Objectif

- Intégrer la classe IHM_WEB à la solution préexistante (celle que vous avez complétée dans l'activité précédente).
- déployer, tester et valider la solution.

Prérequis :

- notion de services réseaux.

Documents/ressources nécessaires :

- corrigé du projet C++ Builder du module SIN21

1 Intégration de la classe IHM_WEB

Dans l'activité précédente vous avez validé la classe. Il s'agit maintenant de l'intégrer dans le projet de base de départ (que vous avez modifié) et d'ajouter quelques lignes de code pour que le fichier HTML soit créé à chaque événement du Timer.

Travail à faire

- Reprenez le projet C++ Builder (ou utilisez la correction) que vous avez modifié.
- Copiez-collez les fichiers **ihm_web.cpp** et **ihm_web.h** de l'activité précédente dans le dossier de votre projet.
- Depuis l'environnement de développement, ajoutez le fichier **ihm_web.cpp** en cliquant sur le menu **Projet->Ajouter au projet** puis choisissez le fichier **ihm_web.cpp** (et pas **ihm_web.h**).
- Maintenant que la classe est déclarée, il faut créer un objet comme ceci :
 - `IHM_WEB pageMeteo("votre_chemin");`
 - On crée un objet avec passage d'un paramètre au constructeur (non étudié ici mais il est nécessaire de le comprendre).
 - Le chemin que vous donnez doit être celui ou vous voulez enregistrer le fichier html. Si vous utilisez un environnement comme EasyPhp, ce devrait être
 - `C:\Program Files\EasyPHP-5.3.5.0\www\exemple.html`. Sachant que le signe '\' pose problème il faut le doubler à chaque fois. Ce qui donne au final :
 - `IHM_WEB pageMeteo("C:\\Program Files\\EasyPHP-5.3.5.0\\www\\meteo.html");`

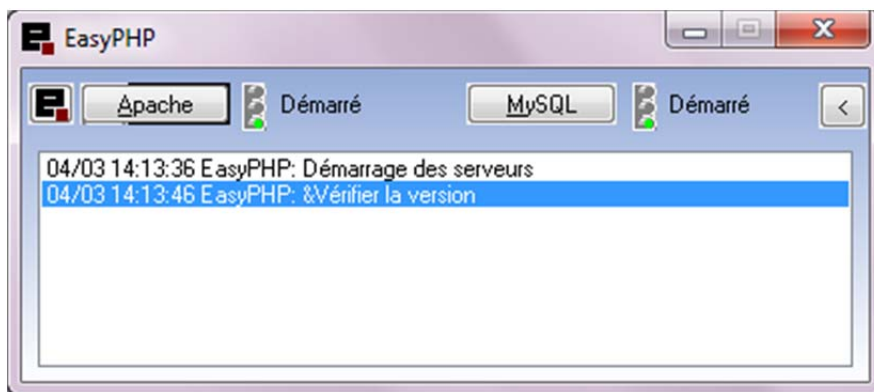
- Vous devez placer la ligne ci-dessus à la fin du traitement des informations dans votre projet. Il faut ensuite utiliser la méthode que vous avez mise au point en donnant en paramètre toutes les données dont vous disposez. Par exemple vous aurez :
 - `pageMeteo.GenererPageWeb(temperature, point_de_rose, humidite, etat_meteo, pression, etat_batterie);` (ce qui est entre parenthèse étant les variables de votre programme, cela peut donc changer).

2 Test avec un serveur web

2.1 Installation d'EasyPHP

L'objectif n'étant pas d'installer un serveur web nous prendrons un package très simple à installer : EasyPHP.

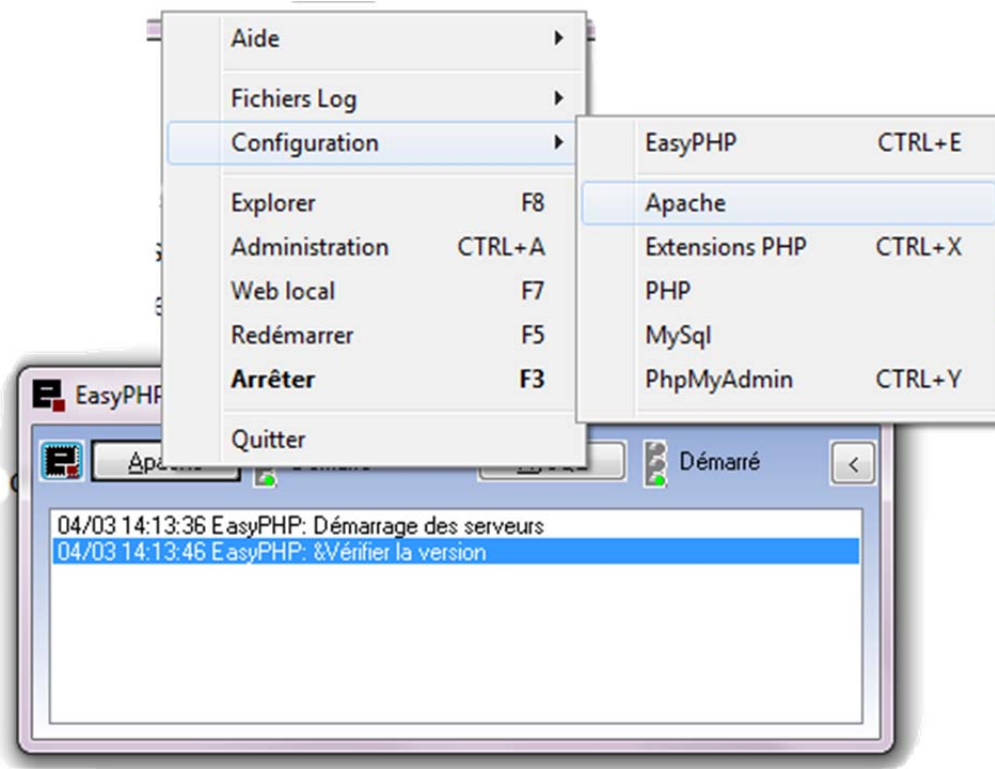
L'installation ne pose aucun problème. L'ensemble est utilisable tout de suite après l'installation. Votre serveur est lancé si vous pouvez voir cette fenêtre :



Remarque : le package d'installation installe aussi un serveur MySQL et un module PHP, ce qui est inutile ici bien sûr.

La racine du serveur se situe dans le répertoire **C:\Program Files\EasyPHP-5.3.5.0\www** (à adapter selon votre installation et votre version, mais si vous avez gardé les valeurs par défaut vous devriez avoir pratiquement la même chose). Le fichier HTML que votre application générera devra être enregistré dans ce répertoire (comme indiqué plus haut).

Remarque : par défaut, votre serveur web n'accepte que les requêtes issues du poste où il a été installé. Afin de rendre le serveur utilisable par tout le monde, il faut que vous alliez modifier une ligne du fichier de configuration en faisant un clic droit sur le bouton **Apache** de la fenêtre d'EasyPHP pour avoir ceci :



Vous sélectionnez **Apache** comme indiqué et vous aurez le fichier de configuration d'Apache dans le *bloc notes*. Repérez la ligne où il y d'écrit **Listen 127.0.0.1 :8888** (qui indique de n'écouter que les requêtes des clients sur le port 8888 provenant de la machine locale) et rajouter en dessous **Listen 80**. Cela permet à n'importe quel navigateur de n'importe quelle machine de se connecter sur le port 80 (qui est le port standard mais non obligatoire d'un serveur web).

Travail à faire

Nous allons maintenant tester la maquette complète. Pour cela vous lancerez le simulateur, *Virtual Serial Port Emulator*, le logiciel client (celui que vous avez mise au point ou le corrigé) et enfin **easyphp**. Depuis un navigateur, vous entrerez l'url suivante **<http://127.0.0.1/meteo.html>** si vous êtes en poste local (mais uniquement pour tester) ou bien l'adresse IP du poste qui héberge le serveur web (manipulation préférable pour constater l'accès à distance). Vous devez voir votre page web avec les données météo se mettre à jour toutes les secondes.