



**UNIVERSITAS INDONESIA**



**ISTIA  
UNIVERSITE D'ANGERS**

**AUTOMATION AND SUPERVISION WEB  
OF OVERHEAD CRANES**

**TESIS**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Teknik**

**FITRIADY  
0906578320**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
KEKHUSUSAN KONTROL INDUSTRI  
DEPOK  
JULI 2011**

## HALAMAN PERNYATAAN ORISINALITAS

**Tesis ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.**

**Nama : Fitriady**

**NPM : 0906578320**

**TandaTangan:**



**Tanggal : 11 Juli 2011**

## HALAMAN PENGESAHAN

Tesis ini diajukan oleh :

Nama : Fitriady  
NPM : 0906578320  
Program Studi : Teknik Elektro  
Judul : Automation And Supervision Web of Overhead Cranes

**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.**

### DEWAN PENGUJI

Pembimbing : Dr. Sebastien Lahaye

Penguji 1 : Dr. Delphine Lalane

Penguji 2 : Prof. Christian Robledo

Penguji 3 : Prof. Abdessamad Kobi

Ditetapkan di : Angers – Perancis

Tanggal : 11 Juli 2011

Mengetahui,

Ka. Departemen Teknik Elektro  
Fakultas Teknik – Universitas Indonesia



Ir. Muhammad Asvial, MSc. PhD.

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan tesis ini. Penulisan tesis ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar ganda (*Double Degree*) Magister Teknik Program Studi Teknik Mesin pada Fakultas Teknik Universitas Indonesia dan Master 2 *System Engineering and Project Management* pada ISTIA – Universite d’ Angers, Perancis.

Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan tesis ini, sangatlah sulit bagi saya untuk menyelesaikan tesis ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Dr. Sebastien Lahaye selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan tesis ini.
2. Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral.
3. Sahabat DDIP Angers yang telah banyak membantu saya dalam menyelesaikan tesis ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga tesis ini membawa manfaat bagi pengembangan ilmu.

Angers, 11 Juli 2011

Penulis



**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertandatangan di bawah ini:

Nama : Fitriady  
NPM : 0906578320  
Program Studi : Kontrol Industri  
Departemen : Teknik Elektro  
Fakultas : Teknik  
Jeniskarya : Tesis

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Non eksklusif (*Non-exclusive Royalty Free Right*)** atas karya ilmiah saya yang berjudul : **Automation and Supervision Web Of Overhead Cranes** beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non eksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Angers

Pada tanggal : 11 Juli 2011

Yang menyatakan



( Fitriady )

## ABSTRAITE

L'**automatisation** est considérée comme l'étape d'un progrès technique où apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de surveillance et de contrôle. La **supervision** est une technique industrielle de suivi et de pilotage informatique de procédés de fabrication automatisés. La supervision concerne l'acquisition de données (mesures, alarmes, retour d'état de fonctionnement) et des paramètres de commande des processus généralement confiés à des automates programmables.

**Le but du projet** sont réaliser des supervisions avec PCVUE utilisant des « WEBVUE » et aussi réaliser une supervision WEB utilisant des Google Web Toolkit (GWT) à partir de java ou .Net (Pour la communication OPC), pages de supervision accessibles depuis un navigateur Web (Internet Explorer, Firefox, ...) pour le pilotage de la maquette ainsi que la supervision du système automatisé. La maquette sera pilotée par un automate Allen Bradley SLC500.

Pour la supervision web qui utilise PcVue-WebVue, nous avons réussi à atteindre l'objectif posé. Nous avons réalisé la supervision web de ponts roulant avec succès. Pour la supervision web qui utilise Google Web Toolkit (GWT), nous avons pris de nouvelles technologies. Mais malheureusement, nous n'avons pas pu réaliser l'objectif posé. C'est parce que nous manquons notre connaissance des langages de programmation. En effet, le problème de démarrer avec de nouveaux outils représente une longue d'initiation (recherche d'information, prise en main des possibilités...)

### **Mots-clés :**

Automatisation, Supervision Web, Ponts Roulant, WebVue, Google Web Toolkit (GWT)

## ABSTRACT

**Automation** is seen as the stage of technical progress of technical devices which appear likely to assist the man, not only in his muscular efforts, but also in his intellectual work of monitoring and control. **Supervision** is an industrial technique for monitoring and computer control of automated manufacturing processes. Supervision for the acquisition of data (measurements, alarms, returning from working) and process control parameters generally outsourced to PLCs.

**The project's goal** is to make supervision PcVue using “WebVue” and also realize a supervision Web using Google Web Toolkit (GWT) from java or .Net (For OPC), pages supervision accessible from a web navigator (Internet Explorer, Firefox, ...) for controlling the layout and supervision of the automation system. The model will be driven by Allen Bradley SLC500.

For supervision web using PcVue-WebVue, we managed to achieve the objective laid. We have carried out the supervision of overhead cranes with success. For supervision web using Google Web Toolkit (GWT), we have taken on new technologies. Unfortunately, we could not achieve the objective laid. This is because we lack our knowledge of programming languages. Indeed, the problem started with new tools is along initiation (information retrieval, handling possibilities ...)

**Keywords :**

Automation, Supervision, overhead cranes, WebVue, Google Web Toolkit (GWT)

## DAFTAR ISI

HALAMAN JUDUL .....	i
LEMBAR PERNYATAAN ORISINALITAS .....	ii
LEMBAR PENGESAHAN .....	iii
KATA PENGANTAR .....	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH .....	v
ABSTARITE .....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	x
DAFTAR LAMPIRAN .....	xii
I. Introduction .....	1
1.1. Présentation du projet .....	1
1.2. L'objectif .....	4
II. Mise en œuvre de WebVue .....	5
2.1. Généralités sur les Web Services .....	5
2.1.1. Architecture .....	5
2.1.2. Principes .....	6
2.1.3. Configuration requis pour les Web Services sur Windows XP .....	7
2.2. Installation et gestion des clients WebVue .....	10
2.2.1. Installation d'un Plug-in Java .....	10
2.2.2. Configuration de l'Apparence de l'Espace de Travail HTML .....	11
2.2.3. Mise-à-jour de l'Applet .....	12
2.2.4. Test de la Connexion Client-Serveur .....	12
2.3. Utilisation de WebVue .....	13
2.3.1. Démarrage de WebVue dans un Navigateur Web .....	14
2.3.2. Connexion à un Serveur WebVue .....	14
2.3.3. Déconnexion du Serveur WebVue .....	15
III. Google Web Toolkit (GWT) .....	17
3.1. Que contient la boîte à outils? .....	17
3.1.1. SDK .....	17
3.1.2. Speed Tracer .....	18
3.1.3. Le plug-in pour Eclipse .....	19
3.1.4. Designer GWT .....	19
3.2. Développer avec Google Web Toolkit .....	19
3.2.1. Écrire .....	19
3.2.2. Débogage .....	20
3.2.3. Optimiser .....	20
3.2.4. Run .....	21



3.3.	La communication entre clients et serveurs (RPC) .....	21
IV.	Application WebVue Pour La Supervision WEB .....	22
4.1.	Méthodologie du Projet .....	22
4.2.	Exigence de Projet WebVue .....	22
4.3.	Dessiner L'Interface Graphique .....	25
4.4.	Mise en Œuvre .....	34
4.5.	Vérification .....	38
V.	Stage Projet de Développement .....	40
5.1.	Standar de Communication OPC .....	40
5.2.	Technologie Flash .....	44
5.2.1.	Développement Flash .....	44
5.2.2.	Integration avec Visual Basic .NET 2008 .....	46
5.3.	Fonctionnement de la Supervision .....	47
5.3.1.	La Vue de Visualisation du Système .....	48
5.3.2.	La Vue Historique des Défauts .....	48
5.3.3.	L'affichage de changement les valeurs de Paramètres IO .....	50
VI.	Application GWT pour la supervision Web .....	51
6.1.	Méthodologie du Projet .....	51
6.1.1.	L'examen des exigences fonctionnelles .....	51
6.1.2.	Identifier les éléments de la conception de l'interface .....	52
6.1.3.	Programmation (Codage) .....	54
6.2.	Vue sur Les Pages du Programme .....	56
6.3.	Problèmes Rencontrées .....	58
VII.	Conclusion .....	59
	Annexes WebVue Tutoriel	
	Annexes Google Web Toolkit (GWT)	
	Références	

## DAFTAR GAMBAR

Figure 1.1	Description de la maquette .....	1
Figure 1.2	Description du pupitre de commande .....	2
Figure 1.3	Maquette de traitement de surface .....	2
Figure 1.4	Architecture informatique et automatisme .....	3
Figure 2.1	Architecture de la communication Client-ServerWebvue .....	6
Figure 2.2	Topologie possible pour l'accès des clients WebVue .....	7
Figure 2.3	Internet Information Services et Classes Properties .....	10
Figure 2.4	La réponse de la connexion client-serveur .....	13
Figure 2.5	La vue de la connexion mot de passe .....	15
Figure 3.1	Description du GWT .....	17
Figure 4.1	Schématique de traitement de surface Pont Roulant .....	23
Figure 4.2	Schématique de fonctionnement de la mode automatique .....	24
Figure 4.3	Modèle le synoptique Background Template .....	25
Figure 4.4	Propriétés de l'animation libellée .....	26
Figure 4.5	Propriétés de l'animation fermeture de la fenêtre .....	26
Figure 4.6a	Événementielle de la mesure du nombre du client .....	28
Figure 4.6b	Événementielle de la mesure du nombre du client .....	28
Figure 4.7	Modèle le synoptique webvue .....	29
Figure 4.8	Propriétés d'animation texte # .....	29
Figure 4.9	Propriétés de la fenêtre du synoptique .....	30
Figure 4.10	L'interface graphique principale WebVue .....	30
Figure 4.11	L'interface graphique principale du projet Pont Roulant .....	31
Figure 4.12	Procède de traitement de surface de Pont Roulant .....	31
Figure 4.13	Procède de faire d'animation sur le bouton « suivant » .....	32
Figure 4.14	Propriétés du bouton « suivant » .....	32
Figure 4.15	Le synoptique Alarme .....	34
Figure 4.16	Propriétés du Serveur WebVue .....	35
Figure 4.17	Propriétés du Options WebVue .....	36
Figure 4.18	Propriétés de la fenêtre initiale .....	38
Figure 4.19	L'affichage sur l'ordinateur WebVue Serveur .....	39
Figure 4.20	L'affichage sur l'ordinateur WebVue Client .....	39
Figure 5.1	Vue KEPServerEx .....	43
Figure 5.2	Vue OPC Client .....	44
Figure 5.3	Environnement de développement Flash .....	44
Figure 5.4	L'animation Flash .....	46
Figure 5.5	Ajout de la référence Flash au projet .....	46
Figure 5.6	Ajout du composant .....	47
Figure 5.7	vue de visualisation du système .....	48
Figure 5.8	vue des Historique des Défauts .....	50
Figure 5.9	vue Paramètre IO .....	50
Figure 6.1	Méthodologie du projet .....	51
Figure 6.2	Ouverture de WindowsBuilder Editor .....	55
Figure 6.3	Les composant graphiques que l'on a avec GWT .....	55
Figure 6.4	Exécuter l'application .....	56
Figure 6.5	Affichage développement mode après l'exécutable .....	56

Figure 6.6	Affichage première page .....	57
Figure 6.7	Affichage deuxième page .....	57
Figure 6.8	Affichage cinquième page .....	58



## DAFTAR LAMPIRAN

Annexes WebVue Tutoriel  
Annexes GWT



# CHAPITRE 1

## INTRODUCTION

### 1.1. Présentation du projet

Le projet que nous avons réalisé consiste à superviser une automatisation de ponts roulant par l'intermédiaire d'un site internet.

Cette partie opérative comporte deux chariots motorisés, appelés respectivement « chariot gauche » et « chariot droit ». Chaque chariot porte un étrier destiné à soulever et à déplacer un « panier » où sont placés les pièces à traiter. Ces « paniers » peuvent être positionnés sur les zones de chargement (les 2 bacs aux extrémités), ou immergés dans les différents bacs de trempage (les 5 bacs centraux). Les paniers ne peuvent être déplacés latéralement qu'en position haute. Les positions des chariots mobiles sont repérées par des capteurs inductifs C1 à C10. Les positions « chariot gauche en C1 » et « chariot droit en C10 » sont les positions de repos des chariots.

Il existe une dissymétrie entre les deux chariots. Le capteur de position activé par le chariot gauche n'est pas situé au dessus de son étrier mobile contrairement au capteur de position activé par le chariot droit.

Les deux chariots sont équipés de capteurs inductifs repérant les positions hautes et basses de leur étrier et d'un capteur inductif de sécurité « anti-collision ».

La maquette de traitement de surface est schématisée ci-dessous.

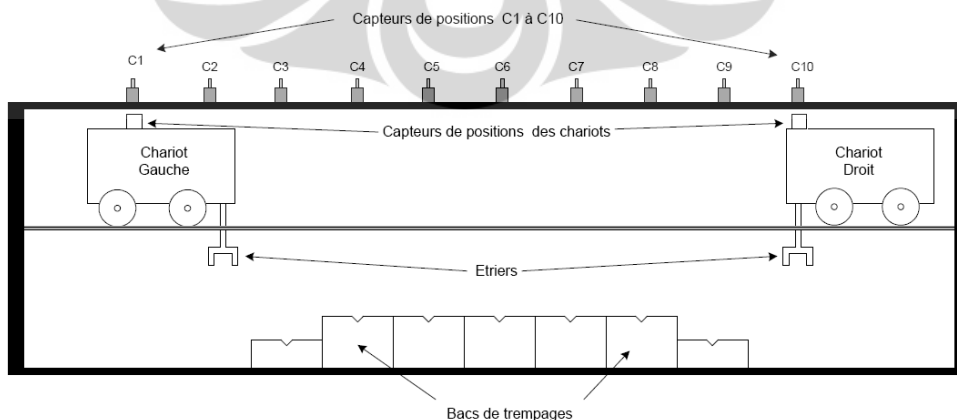


Figure 1.1 Description de la maquette

## PUPITRE DE COMMANDE

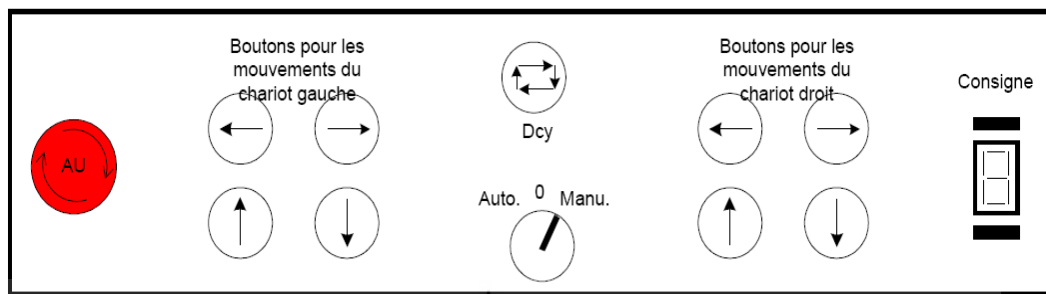


Figure 1.2 Description du pupitre de commande

La maquette sera pilotée par un automate Allen Bradley SLC500 via le logiciel constructeur RSLogic. La communication entre le logiciel de programmation automate et l'automate sera par le serveur RSLinx via la liaison Ethernet afin de laisser libre la liaison série utilisée pour le serveur OPC. Une supervision a été réalisée pour permettre à un opérateur distant de contrôler et donner des ordres de fabrication. L'application de supervision sera développée en VB.NET. La communication entre l'application de supervision et l'automate se fera en utilisant le serveur OPC KEPServer via liaison série AB-DF1 (RS 232).



Figure 1.3 Maquette de traitement de surface

Pour résumer la configuration matérielle présentée précédemment voici ce schéma :

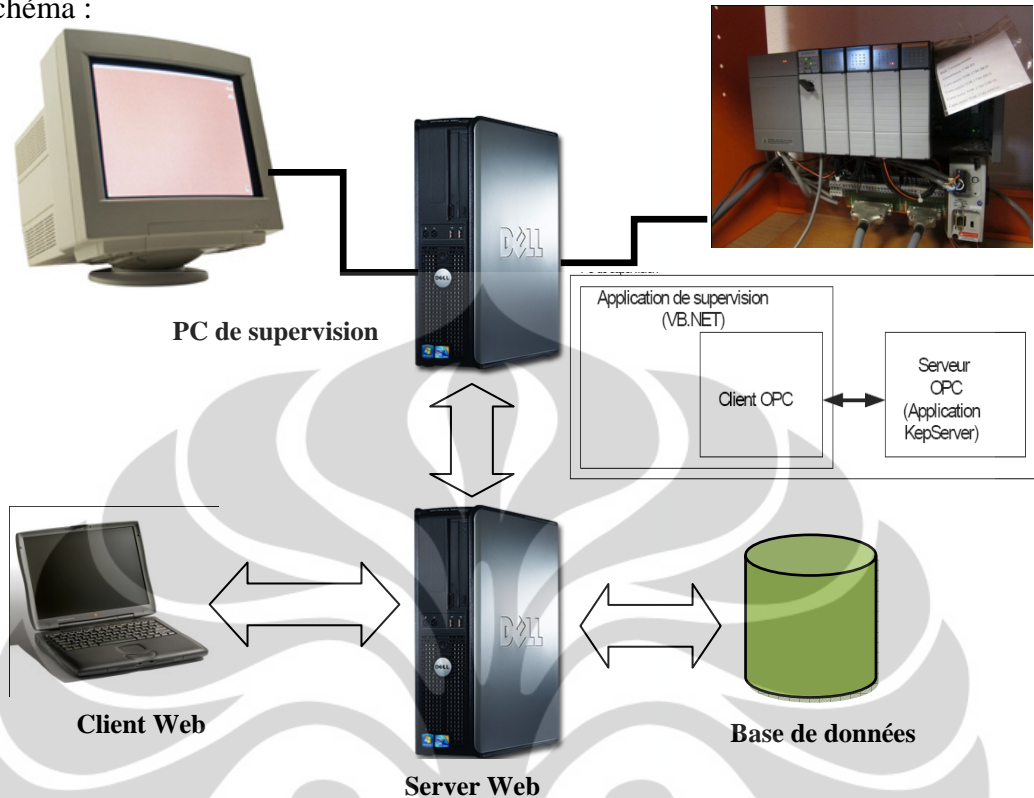


Figure 1.4 Architecture informatique et automatisée

Pour effectuer la supervision d'une automation de ponts roulant, nous utilisons WebVue et GWT (Google Web Toolkit). WebVue est composé d'une applet, qui se déroule sur les clients Web, et d'une partie serveur qui fournit les synoptiques et données à partir d'un produit PcVue dédié. GWT est un Framework permettant de créer des applications RIA (Rich Internet Application) facilement. GWT permet de créer des pages web dynamiques grâce au langage Java.

La supervision permet de :

- ✓ visualiser l'état de l'application (position des chariots et de leurs étriers)
- ✓ sélectionner un mode de fonctionnement si le sélecteur 3 positions du pupitre de commande est en position 0

- ✓ déplacer les chariots si le mode manuel est sélectionné depuis la supervision
- ✓ paramétrer et démarrer des cycles de trempage si le mode automatique est sélectionné depuis la supervision,
- ✓ visualiser plusieurs indicateurs destinés à la maintenance : durées cumulées des trempages dans chacun des bacs (pour le remplacement des solutions de trempage), ainsi que le temps total
- ✓ déplacement de chacun des chariots,
- ✓ visualiser un historique des arrêts d'urgence,
- ✓ visualiser un historique des cycles déjà effectués en mode automatique (n° d'ordre, différent temps de trempage, heure de début, heure de fin).

Tout le fonctionnement du système est enregistré dans une base de données MySQL.

## 1.2. L'objectif

Le but du projet est de réaliser des supervisions avec PCVUE utilisant des « WEBVUE » et réaliser une supervision WEB utilisant des Google Web Toolkit (GWT) à partir de java ou .Net (Pour la communication OPC), pages de supervision accessibles depuis un navigateur Internet (Internet Explorer, Firefox, ...) pour le pilotage de la maquette ainsi que la supervision du système automatisé. La maquette sera pilotée par un automate Allen Bradley SLC500.



## CHAPITRE 2

### MISE EN ŒUVRE DE WEBVUE

#### 2.1. Généralités sur les Web Services

Reposant sur l'utilisation d'un simple navigateur, **WebVue** permet de visualiser et de contrôler à distance un procédé au travers d'un réseau Internet/Intranet. Un utilisateur disposant de droits appropriés (nom d'utilisateur et mot de passe) peut accéder à une application **PcVue** depuis n'importe quel poste sur le réseau. WebVue s'exécute dans un navigateur web et est indépendant du système opératoire. Il affiche directement des synoptiques **PcVue** sans avoir à les modifier.

Sans aucune installation préalable, **WebVue** c'est :

- La visualisation de synoptiques PcVue au travers du Web.
- Un accès facile aux données du procédé (alarmes, historiques, tendances).
- Une parfaite sécurité des échanges associée à la gestion des droits d'accès de PcVue.
- Une solution idéale pour la télémaintenance et la télésurveillance.

##### 2.1.1. Architecture

WebVue est composé d'une applet, qui se déroule sur les clients Web, et d'une partie serveur qui fournit les synoptiques et données à partir d'un produit PcVue dédié.

La communication entre le serveur Web de **PcVue** et le client **WebVue** se fait via Microsoft IIS. Le client **WebVue** est rafraîchi en permanence et les données sont échangées via le protocole HTTP ou HTTPS si un cryptage s'avère nécessaire (SSLV2).

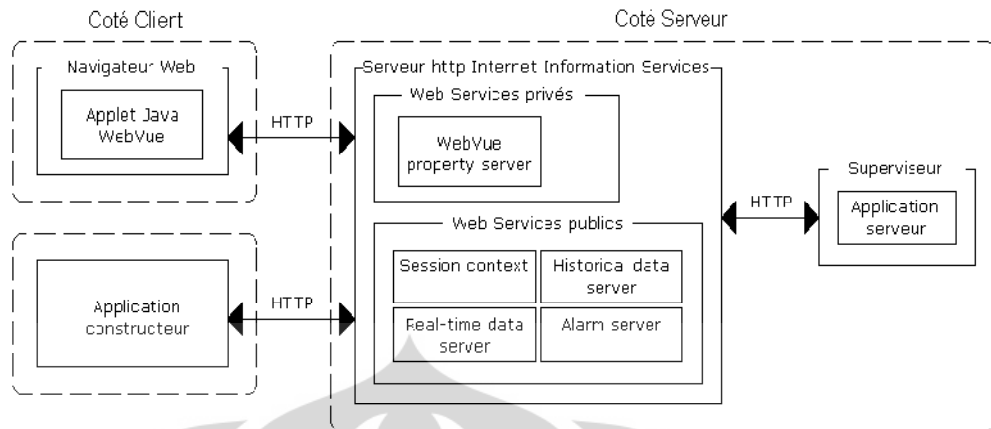


Figure 2.1 Architecture de la communication Client-Server WebVue

### 2.1.2. Principes

Le serveur HTTP de PcVue dédié fournit au navigateur Web du poste client :

- Une page HTML de démarrage.
- L'ensemble des classes JAVA de base ainsi que l'applet WebVue client.

L'applet **WebVue** une fois chargée va :

- Se connecter sur ce serveur via une liaison TCP/IP.
- Authentifier l'utilisateur.
- Télécharger le synoptique souhaité.
- Interpréter dynamiquement les changements de propriétés graphiques du synoptique que lui fournit le serveur PcVue.
- Permettre la navigation parmi les synoptiques de l'application.
- En relation avec les droits utilisateurs, autoriser l'acquiescement d'alarmes, le passage de télécommandes et de télé consignes des historiques d'un ensemble de variables.

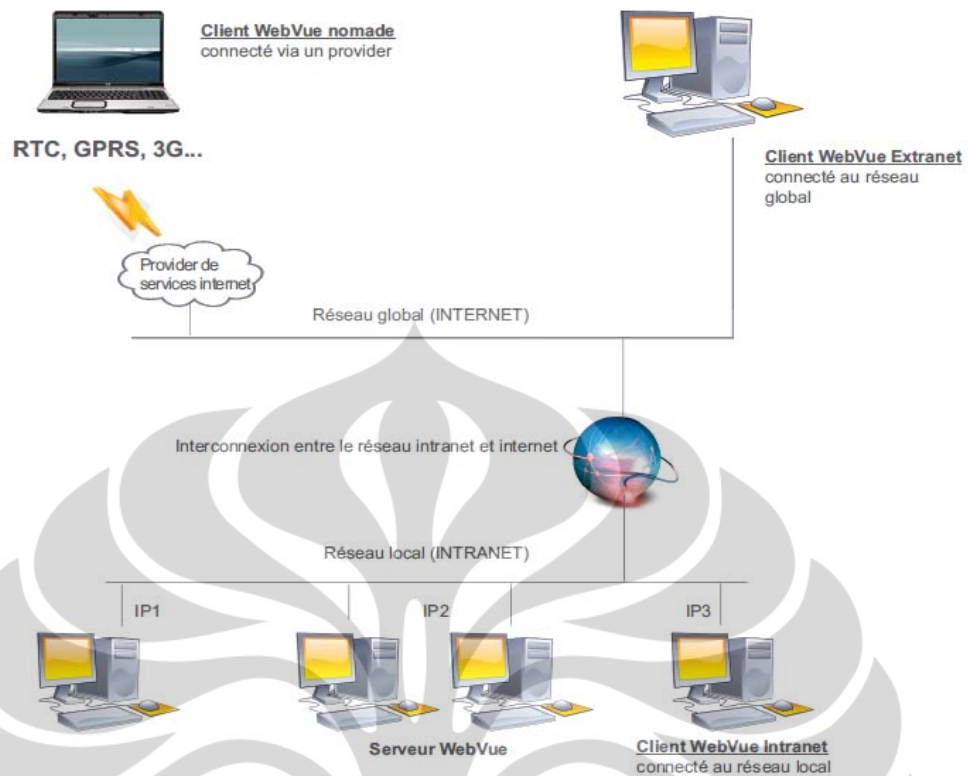


Figure 2.2 Topologie possible pour l'accès des clients WebVue

### 2.1.3. Configuration requis pour les Web Services sur Windows XP

Les Web Services du Superviseur utilisent un certain nombre de composants fournis par Microsoft qui doivent être installés avant la configuration et l'utilisation des Web Services. Les composants doivent être installés dans l'ordre suivant :

- L'étape 1 doit être exécutée avant la procédure d'installation du Superviseur. Si vous devez installer des composants Windows, vous aurez besoin du CD de distribution Windows.
- Les étapes 2 et 3 sont exécutées automatiquement par la procédure d'installation du Superviseur.
- L'étape 4 sera exécutée après la création d'un projet.

IIS doit être installé avant le Framework.Net. Si le Framework.NET est déjà installé mais pas IIS, le Framework.NET doit d'abord être désinstallé.

### **Etape 1 - IIS et extensions FrontPage**

1. A partir du menu Démarrer de Windows, sélectionner le Panneau de configuration.
2. Sélectionner l'option Ajout/Suppression de programmes.
3. Sélectionner l'option Ajouter ou supprimer des composants Windows.
4. Sélectionner le composant Services Internet (IIS), puis cliquer sur le bouton Détails et vérifier que l'option Extensions Serveur FrontPage 2000 est cochée. Fermer la boîte de dialogue en cliquant sur le bouton OK.
5. Sélectionner le bouton Suivant pour installer IIS. Durant le processus d'installation vous pouvez être invités à insérer le CD de distribution Windows.

### **Etape 2 - Installation de .NET Framework 1.1**

Le .NET Framework est installé pendant le processus d'installation du Superviseur. Si un problème survient, vous pouvez l'installer manuellement à partir du CD de distribution en utilisant le chemin d'accès suivant : Start\Net.Framework.1.1\dotnetfx.exe

### **Etape 3 - Installation de WSE 2.0**

WSE 2.0 est installé pendant le processus d'installation du Superviseur. Si un problème survient, vous pouvez l'installer manuellement à partir du CD de distribution en utilisant le chemin d'accès suivant : Sv\Wse\Microsoft WSE 2.0 SP3 Runtime.msi

WSE doit être installé après .NET Framework.

### **Etape 4 - Répertoires virtuels**

Voir l'aide principale du Superviseur pour plus d'informations sur la configuration des répertoires virtuels de WebVue et du Web Services Toolkit en utilisant la boîte de dialogue de configuration de WebVue.

Le terme répertoire virtuel est utilisé pour décrire le mécanisme permettant à un client Web d'accéder à des fichiers localisés sur un serveur en utilisant un alias plutôt que sa localisation réelle.

L'utilisation d'un alias pour accéder aux fichiers présente plusieurs avantages :

- Parce qu'un alias est habituellement plus court que le chemin d'accès du répertoire, il est plus facile d'utilisation.
- Un alias est plus sécurisant car les utilisateurs ne connaissent pas la localisation physique des fichiers sur le serveur et ne peuvent pas utiliser ces informations pour modifier vos fichiers.
- Les alias vous permettent de modifier plus facilement la localisation de votre application. Plutôt que de modifier l'URL, vous modifiez le lien entre l'alias et la localisation physique du répertoire.

#### ***Répertoires virtuels utilisés par WebVue***

Les répertoires virtuels suivants sont utilisés par WebVue :

- Classes - Utilisé par le client WebVue pour charger l'applet Java.
- WebPropertyServer - Utilisé par le client WebVue pour échanger les données temps réel avec le serveur.
- <Project Name> - Utilisé par le client WebVue pour accéder à la configuration du projet. <Project Name> est le nom du projet courant.

#### ***Répertoires virtuels utilisés par le Web Services Toolkit***

Les répertoires virtuels suivants sont utilisés par le Web Services Toolkit :

- SessionContext - Utilisé par les applications tierces pour la gestion des sessions et des utilisateurs.
- RealTimeData - Utilisé par les applications tierces pour accéder aux valeurs temps-réel des variables du Superviseur.
- RealTimeAlarm - Utilisé par les applications tierces pour accéder aux alarmes du Superviseur.
- HistoricalData - Utilisé par les applications tierces pour accéder aux données historiques du Superviseur.

### *Répertoires virtuels utilisés par les portails de démonstration*

Le Superviseur est livré avec de portail Web tirant partie du Web Services Toolkit et qui utilise également des répertoires virtuels :

- SVWEB\_STAGE

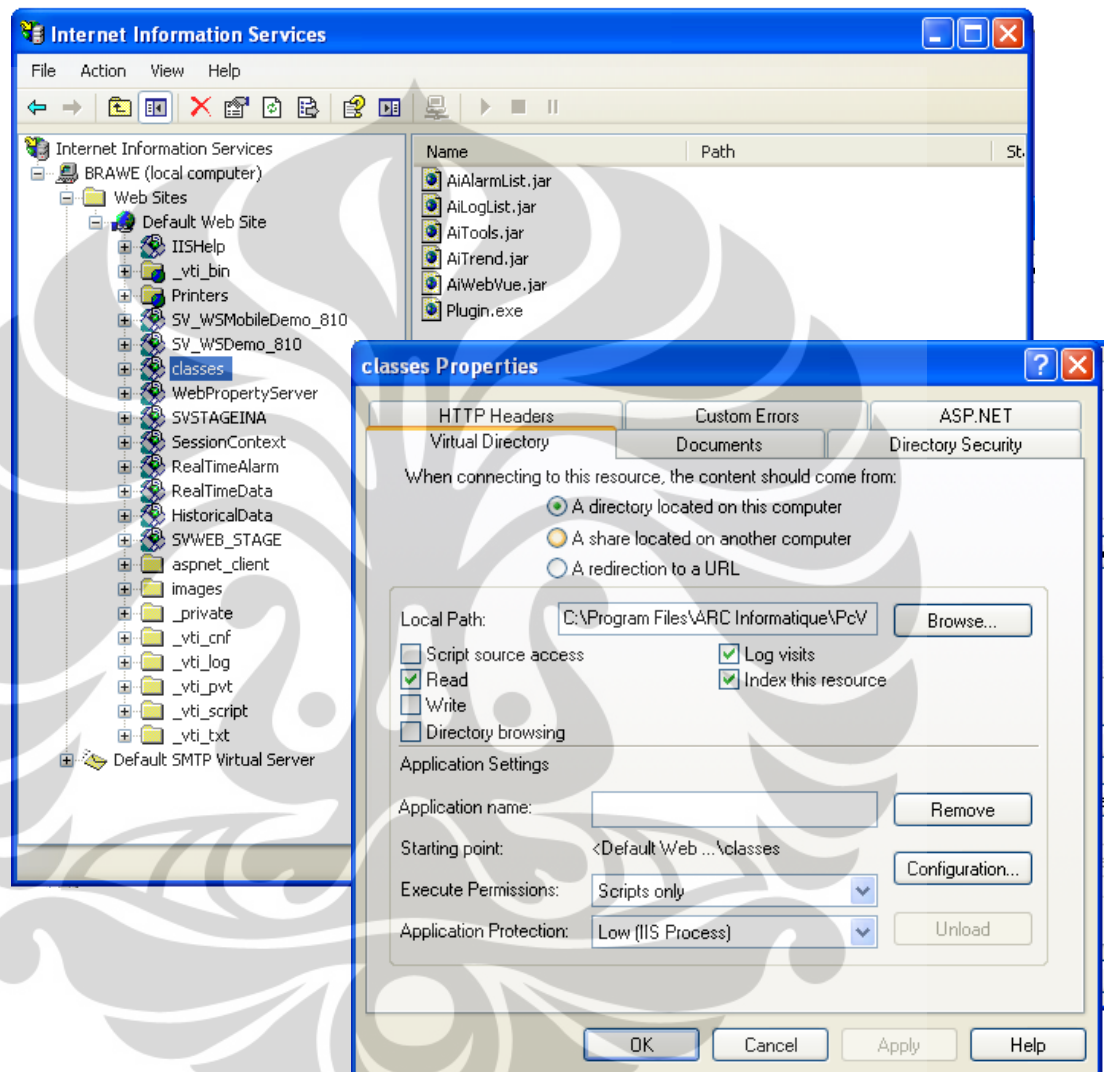


Figure 2.3 Internet Information Services et Classes Properties

## 2.2. Installation et gestion des clients WebVue

### 2.2.1. Installation d'un Plug-in Java

Il n'y a pas de configuration requise pour le Client, excepté que le navigateur Web doit avoir le Plug-in Java installé.

Les environnements Run-Time Java 2 et Plug-in Java doivent être les versions fournies avec le Superviseur que nous utilisons.

Une version Win32 auto-installable de l'environnement Run-Time Java 2 (incluant le Plug-in Java) est copiée sur votre disque dur lors de l'installation du Superviseur.

Pour installer le plug-in Java :

1. Sélectionner le fichier exécutable dans le sous-répertoire bin\WebServerExtensions\WebVue\Plugin sous le répertoire racine du Superviseur (son nom est du type 'j2re-...-windows-i586-i.exe'.)
2. Copier le sur le PC où le Client WebVue sera exécuté.
3. Exécuter le (double-clic) et suivre les instructions.

### **2.2.2. Configuration de l'Apparence de l'Espace de Travail HTML**

Les développeurs maîtrisant le langage HTML peuvent personnaliser la page hypertexte qui détermine comment le navigateur Web affiche l'Applet. Par exemple, un style "personnalisé" peut être incorporé ou l'espace de travail peut être réduit de 100 % sur la longueur et la largeur disponibles dans la fenêtre de navigation.

#### *Personnalisation de l'espace de travail HTML*

Vous pouvez modifier l'espace de travail HTML de la façon suivante :

1. Déconnecter le Client WebVue.
2. Vider le cache des pages du navigateur afin d'éviter de recharger la page existante.
3. Dans la structure de fichier du Superviseur, sélectionner le fichier INDEX.HTML dans le sous-répertoire USR\<<Project name>\WEB.
4. Ouvrir une copie de celui-ci avec un éditeur de texte ou avec un éditeur Web adapté.
5. Personnaliser le suivant vos besoins.
6. Sauvegarder le avec le nom USER.HTML dans le même répertoire (noter le 'L' dans le suffixe).
7. Redémarrer à la fois le Superviseur et le Client WebVue.

Votre fichier utilisera l'apparence personnalisée de l'espace de travail HTML dans le Client WebVue.

### 2.2.3. Mise-à-jour de l'Applet

Une fois que le navigateur Web d'un PC a chargé une copie de l'applet WebVue dans son cache, il n'a pas besoin de le faire de nouveau. Cependant, vous pouvez avoir besoin de mettre à jour le logiciel WebVue, en particulier sur chaque PC sur lequel fonctionne le Client WebVue.

Si vous lancez le procédé d'installation du Superviseur, une nouvelle version d'Applet remplacera automatiquement la précédente. Cela consiste en un jeu de fichiers JAR (Modules Run-Time 'Java ARchive'). Ceux-ci résident dans le répertoire Bin\WebServerExtensions\Webvue\Classes dans le répertoire principal (racine) du Superviseur. Pour forcer un navigateur à charger la nouvelle applet, vous devez vider le cache du plug-in Java. La prochaine fois que vous exécutez WebVue sur le navigateur, la nouvelle applet sera chargée à partir du Superviseur.

Le Cache du navigateur Web et le cache d'image du Serveur WebVue ne sont pas reliés. Les navigateurs conservent également d'autres caches et des listes dynamiques, comme celles pour la navigation des pages précédemment visualisées.

#### Vidage du cache plug-in Java

1. Ouvrir le panneau de configuration de Windows.
2. Ouvrir le panneau de configuration du plug-in Java en double-cliquant sur l'icône Java Plug-in.
3. Sélectionner l'onglet Cache.
4. Cliquer sur le bouton Effacer pour vider le cache. Cliquer sur Oui pour confirmer.
5. Fermer le plug-in Java et le panneau de configuration Windows.

### 2.2.4. Test de la Connexion Client-Serveur

Vous pouvez tester la connexion entre chaque PC en utilisant un utilitaire connu sous le nom de Ping ('PacketInterNetGroper'). Dans un réseau IP,



Ping envoie un paquet simple et attend une réponse, afin de tester la fonction la plus basique d'un réseau IP (livraison d'un paquet simple).

#### Utilisation de Ping

1. Ouvrir une boîte de commande DOS sur votre PC.
2. Taper ping <Computer Name>ou ping <IP Address>, par exemple 'ping poste1' ou 'ping 90.0.0.50'.

Si la connexion se fait, la réponse sera telle que le montre l'image ci-dessus.

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\usrlocal>ping 172.20.82.106

Envoi d'une requête 'ping' sur 172.20.82.106 avec 32 octets de données :

Réponse de 172.20.82.106 : octets=32 temps=1 ms TTL=128
Réponse de 172.20.82.106 : octets=32 temps<1ms TTL=128
Réponse de 172.20.82.106 : octets=32 temps<1ms TTL=128
Réponse de 172.20.82.106 : octets=32 temps<1ms TTL=128

Statistiques Ping pour 172.20.82.106:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

C:\Documents and Settings\usrlocal>

```

Figure 2.4 La réponse de la connexion client-serveur

Si vous avez des difficultés de connexion avec un Serveur WebVue, assurez-vous auprès de votre administrateur réseau qu'il n'y a pas de serveur de Proxy entre le Client WebVue et le Serveur WebVue.

### 2.3. Utilisation de WebVue

WebVue est constitué de deux composants :

- Un client léger (appelé Client WebVue) qui s'exécute sous une machine virtuelle Java dans un navigateur Web.
- Un serveur (appelé Serveur WebVue) qui est un composant du Superviseur.

Le Client WebVue et le Serveur WebVue communiquent en utilisant HTTP (HyperText Transfer Protocol) au travers d'un Intranet, de l'Internet ou d'une connexion RAS.

Le Client WebVue et le Serveur WebVue sont indépendants des notions client-serveur d'une application de supervision multiposte.

### **2.3.1. Démarrage de WebVue dans un Navigateur Web**

Pour démarrer WebVue dans un navigateur Web, vous entrez l'URL (Universal Resource Locator) du projet sur lequel le Superviseur est lancé, dans le champ adresse. L'URL prend la syntaxe suivante :

`http://computer id/<projectname>`

- `<Computer id>` - Le nom ou l'adresse IP du PC sur lequel le Superviseur exécutant le serveur WebVue est localisé. Par exemple, Station1 ou 90.0.0.50.
- `<projectname>` - Le nom du projet que le Superviseur exécute.

Vous pouvez tester WebVue en l'exécutant sur le même PC que le Superviseur. Dans ce cas, vous utilisez localhost comme le `<Computer id>`.

Une fois que le navigateur Web a trouvé le PC sur lequel le Superviseur est lancé, le message Loading Java Applet s'affiche à l'écran. Le navigateur Web charge alors l'application Java WebVue. La première fois que vous lancez WebVue, ceci peut prendre plusieurs minutes suivant le type de connexion (intranet, internet ou RAS). Par la suite, cela se fera beaucoup plus rapidement car l'application Java est sauvegardée localement dans le cache du navigateur Web.

Lorsque l'application Java est chargée, un écran de démarrage WebVue s'affiche suivi par une boîte de dialogue demandant la connexion (login).

### **2.3.2. Connexion à un Serveur WebVue**

Pour se connecter à un Serveur WebVue, un utilisateur doit avoir les droits d'accès (Configuration.Administrer.Droits.Profils et Utilisateurs) comme

décrit dans le chapitre de l'aide principal Création de Profils et d'Utilisateurs. Si vous modifiez les droits d'accès, les modifications seront immédiatement prises en compte dans le Superviseur et les Clients WebVue.

Pour établir la connexion au serveur, vous devez fournir les détails de l'utilisateur :

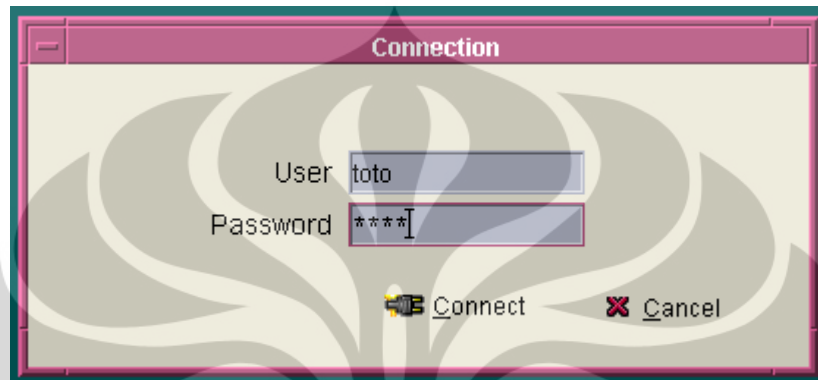


Figure 2.5 La vue de la connexion mot de passe

Nom et mot de passe - Entrer le nom et le mot de passe de l'utilisateur et cliquer sur l'icône Connexion.

WebVue termine le processus de démarrage et affiche le synoptique de démarrage de l'utilisateur.

S'il vous est demandé "Do you want to allow software such as ActiveX controls and plug-ins to run?", sélectionner Oui.

Si un message d'erreur apparaît "Accès au Serveur WebVue refusé, Client déconnecté", sélectionner OK, fermer le navigateur, vérifier que le Serveur WebVue est lancé avec les paramètres corrects, puis répéter la procédure de connexion.

### 2.3.3. Déconnexion du Serveur WebVue

Pour déconnecter un client WebVue à partir du serveur WebVue, saisir un URL différent ou fermer le navigateur Web.

## CHAPITRE 3

### GOOGLE WEB TOOLKIT (GWT)

Google Web Toolkit (GWT) est un kit de développement pour la construction et l'optimisation des applications complexes basée sur un navigateur. Son objectif est de permettre le développement d'applications web de haute performance, sans que le développeur n'ait à être un expert des navigateurs, XML http Request, et JavaScript. GWT est utilisé par de nombreux produits chez Google, dont Google Wave et la nouvelle version d'AdWords. Il est open source, complètement libre, et utilisé par des milliers de développeurs dans le monde.

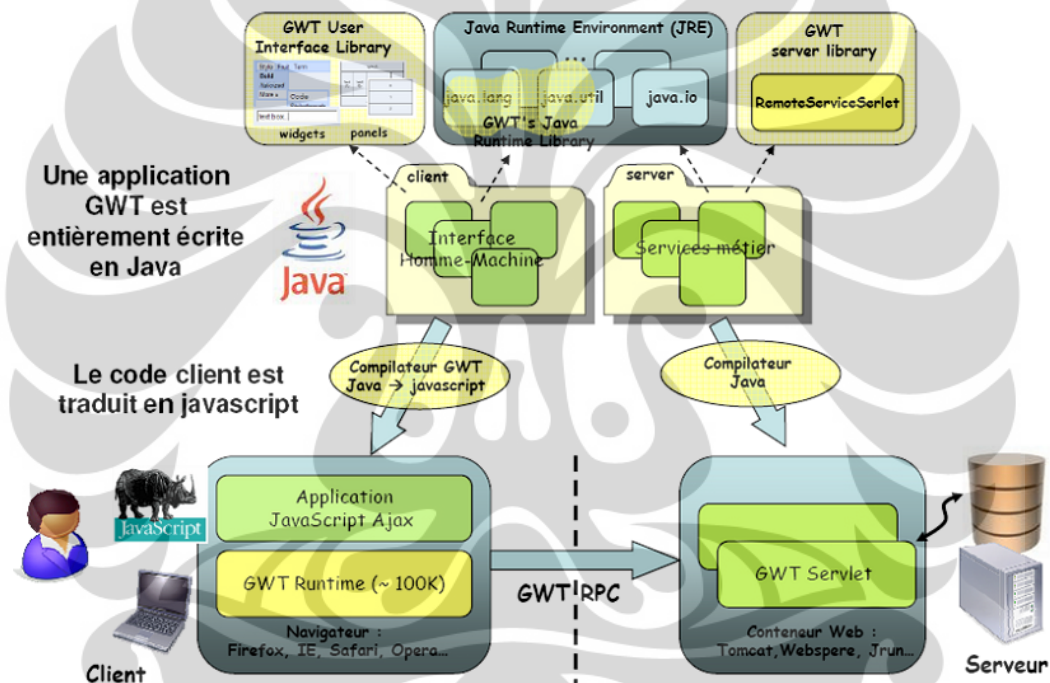


Figure 3.1 Description du GWT

### 3.1. Que contient la boîte à outils?

#### 3.1.1. SDK

Le SDK GWT contient les bibliothèques Java API, le compilateur, et le serveur de développement. Il vous permet d'écrire des applications côté client en Java et de les déployer comme JavaScript.

L'écriture d'applications web peut être un processus fastidieux et source d'erreurs. Vous pouvez passer de 90% de votre temps de travail autour de bizarreries liées au navigateur. En outre, la construction, la réutilisation et

le maintien de grandes bases de code JavaScript et composants AJAX peuvent être difficile et fragile. Google Web Toolkit (GWT) facilite ce travail en permettant aux développeurs de construire rapidement et maintenir des applications JavaScript front-end dans le langage de programmation Java.

### **Comment fonctionne le SDK**

Avec le SDK GWT, vous écrivez votre front AJAX dans le langage de programmation Java qui GWT le compile en JavaScript optimisé et qui fonctionne automatiquement dans tous les principaux navigateurs. Au cours du développement, vous pouvez parcourir rapidement dans le même "edit - rafraîchissement - vue" cycle vous êtes habitué à l'aide de JavaScript, avec l'avantage supplémentaire d'être en mesure de déboguer et de mettre en place votre ligne de code Java en ligne. Quand vous serez prêt à déployer, le compilateur GWT compilation de votre code en source Java optimisé, les fichiers JavaScript autonome.

### **Écrire des applications AJAX en langage Java, puis compiler à l'optimisation de JavaScript**

Contrairement à JavaScript qui travaille seulement au un niveau textuel, le compilateur GWT effectue une analyse complète des statiques et optimise votre code GWT, produisant souvent un code JavaScript qui se charge et s'exécute plus rapidement que l'équivalent JavaScript manuscrit. Par exemple, le compilateur GWT élimine en toute sécurité le code mort pour à faire en sorte que votre script compilé soit le plus petit possible.

#### **3.1.2. Speed Tracer**

Speed Tracer est une extension de Chrome qui vous permet de repérer les problèmes de performances dans vos applications web.

Speed Tracer est un outil pour vous aider à identifier et résoudre les problèmes de performances dans vos applications web. Il permet de visualiser des mesures qui sont prises à partir de points d'instrumentation à l'intérieur du navigateur et les analyse votre application quand elle

s'exécute. Speed Tracer est disponible en tant qu'extension Chrome et fonctionne sur toutes les plates-formes où les extensions sont actuellement prises en charge (Windows et Linux). Utilisation Speed Tracer vous êtes capable de faire une meilleure idée de l'endroit où le temps est passé dans votre application. Ceci inclut les problèmes causés par l'analyse et l'exécution JavaScript, mise en page, l'appariement de sélection, la gestion des événements DOM, le chargement des ressources réseau, les feux de minuterie, les rappels XMLHttpRequest, et plus encore.

### **3.1.3. Le plug-in pour Eclipse**

Le plug-in pour Eclipse IDE fournit un soutien pour Google Web Toolkit et les projets de cette appli web.

Google Plugin pour Eclipse est un ensemble d'outils de développement logiciel qui permet aux développeurs Java des rapidement concevoir, construire, optimiser et déployer des applications de cloudcomputing. GPE aide les développeurs à créer un client riche en utilisant le code Ajax Google Web Toolkit, à optimiser les performances avec Speed Tracer, et à déployer sans effort des applications grâce à l'App Engine. Ces outils puissants suppriment l'ennui et les développeurs peuvent se concentrer sur la création de l'application. GPE est la première suite d'outils de développement intégrés, conçus spécialement pour les développeurs Java Eclipse pour créer des applications de qualité rapide et fiable.

### **3.1.4. Designer GWT**

Designer GWT vous permet de créer des interfaces utilisateur en quelques minutes avec des outils de conception intelligente. Glisser-déposer, et la génération automatique de code.

## **3.2. Développer avec Google Web Toolkit**

### **3.2.1. Écrire**

Le SDK GWT fournit un ensemble de base des API Java et Widgets. Il vous permet d'écrire des applications AJAX en Java, puis compiler les

sources hautement optimisées pour JavaScript qui s'exécute sur tous les navigateurs, y compris les navigateurs mobiles pour Android et l'iPhone. Construire des applications AJAX de cette manière est plus productive grâce à un niveau supérieur d'abstraction comme la manipulation du DOM et de la communication XHR.

### 3.2.2. Débogage

Vous pouvez déboguer des applications AJAX dans votre IDE favori comme vous le feriez pour une application de bureau, et dans votre navigateur favori comme vous le feriez si vous codiez en JavaScript. Le développeur GWT plugin enjambe le fossé entre bytecode Java dans le débogueur JavaScript et le navigateur. Grâce à développeur GWT plugin, il n'y a pas de compilation de code JavaScript pour le voir dans le navigateur. Vous pouvez utiliser le même cycle de rafraîchissement pour modifier la carte page à l'aide de JavaScript, et pour dans le même temps, inspecter les variables, points d'arrêt définis, et d'utiliser tous les outils de débogage qui s'offrent à vous avec Java. Et parce que le mode de développement GWT est maintenant dans le navigateur lui-même, vous pouvez utiliser des outils comme Firebug et l'inspecteur que vous codez en Java.

### 3.2.3. Optimiser

Google Web Toolkit contient deux outils puissants pour créer des applications web optimisées. Le compilateur GWT effectue les optimisations globales au sein de votre code de base - méthodes in-lining, en supprimant le code mort, l'optimisation des chaînes, et plus encore. En fixant des split-points dans le code, il peut également segmenter votre téléchargement en fragments multiples JavaScript, le fractionnement des applications à grande accélérer le temps de démarrage. Goulets d'étranglement de performance ne sont pas limités à JavaScript. Disposition du navigateur et CSS se comportent souvent de façon étrange qui est difficiles à diagnostiquer. Speed Tracer est une nouvelle extension

Chrome de Google Web Toolkit qui vous permet de diagnostiquer les problèmes de performance dans le navigateur.

#### 3.2.4. Run

Lorsque vous êtes prêt à déployer, GWT compile votre code source Java en optimisé, autonome JavaScript fichiers qui s'exécutent automatiquement sur tous les principaux navigateurs, ainsi que les navigateurs mobiles pour Android et l'iPhone.

### 3.3. La communication entre clients et serveurs (RPC)

**RPC** (*RemoteProcedureCall*) est un protocole réseau permettant de faire des appels de procédures sur un ordinateur distant à l'aide d'un serveur d'applications. Ce protocole est utilisé dans le modèle client-serveur et permet de gérer les différents messages entre ces entités.

GWT fournit ses propres appels de procédure distante (RPC) qui permettent au client de GWT pour appeler les méthodes côté serveur. L'amies en œuvre de GWTRPC est basée sur la technologie des servlets. GWT permet aux objets Java pour être envoyés directement entre le client et le serveur qui est automatiquement sérialisés parle cadre. Avec GWTRPC la communication est presque transparente pour le client GWT et toujours asynchrone afin que le client ne bloque pas pendant la communication.

Le GWTRPC cadre, il est facile pour les composants client et serveur de votre application web pour échanger des objets Java sur HTTP. Le code côté serveur qui est invoquée par le client est souvent désigné comme un service. La mise en œuvre d'un service GWTRPC est basée sur l'architecture bien connue des servlets Java. Dans le code du client, vous utilisez une classe proxy généré automatiquement pour passer des appels vers le service. GWT va gérer la sérialisation d'objets Java en passant en arrière-arguments dans les appels de méthode et la valeur de retour.

Le servlet côté serveur est généralement considéré comme un «service» et l'appel d'une procédure à distance est appelée «l'invocation d'un service ». Ces objets peuvent ensuite être utilisées sur le client (IU) côté.



## CHAPITRE 4

### APPLICATION WEBVUE POUR LA SUPERVISION WEB

#### 4.1. Méthodologie du Projet

Comme l'un des applications du web service, WebVue sera utilisé pour refaire le TP supervision industrielle Pont Roulant alors peut l'afficher au travers d'un navigateur web (Internet Explorer, Firefox, ...). A cet effet, les méthodes qui seront utilisées pour le réaliser sont la suivante:

- **Exigences** – pour être en mesure utilisant WebVue comme un web supervision, il doit alors être besoin de pré-requis avant de commencer à l'utiliser. Ensuite, on va procéder de traitement de surface de projet Pont Roulant d'après TD-TP Supervision Industrielle.
- **Dessin** – ce sera dessiner les interfaces graphiques qui sont affichées au travers d'un navigateur web.
- **Mise en œuvre** – les interfaces graphiques qui ont été dessinées seront stockées dans un répertoire virtuel à la façon de la configuration de serveur WebVue. Donc, il peut s'afficher sur le navigateur web. (implementasi bagaimana cara mimic2 yang sudah dibuat di webvue dapat ditampilkan pada web browser)
- **Vérification** – ce sera faire de vérification du projet à l'aide de deux ordinateurs qui se connectent dans un réseau local. Un ordinateur est comme un serveur et l'autre comme un client.

#### 4.2. Exigences de Projet WebVue

Comme expliqué dans le chapitre 2, nous devons remplir les exigences qui devaient exécuter l'application WebVue. Si les conditions ne sont pas remplies, il y aura provoquer des défaillances au moment de l'affichage des synoptiques au travers d'un navigateur web.

IIS doit être installé afin de créer un répertoire virtuel qui contient des projets PCVue ont été faites. Ce répertoire virtuel peut aussi être interprété comme un répertoire Serveur qui être accessibles par les clients de Web services.

En outre, le plug-in Java doit également être installé pour exécuter et afficher les synoptiques qui a été faite sur l'application PCVue au travers d'un navigateur web en temps réel.

### *Description du système automatisé à superviser*

D'après TD-TP Supervision Industrielle, Allen Bradley PCVue, on s'intéresse au procédé de traitement de surface schématisé ci-dessous.

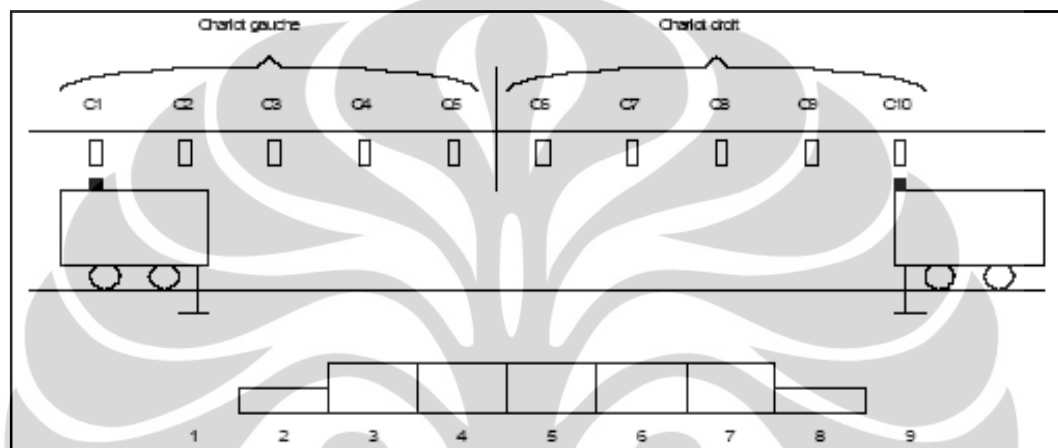


Figure 4.1 Schématisation de traitement de surface Pont Roulant

Celui-ci comporte deux chariots motorisés, appelés "chariot gauche" et "chariot droit". Chaque chariot dispose d'un étrier destiné à soulever et à déplacer un "panier" dans lequel sont placées les pièces à traiter. Les emplacements 1 et 9 constituent des zones de repos pour les chariots. Les bacs 2 et 8 représentent respectivement une zone de chargement et une zone de déchargement. Les bacs 3 à 7 sont des zones de trempage. Les positions des chariots mobiles sont repérées par des capteurs inductifs C1 à C10.

Ce procédé est piloté par un automate Allen-Bradley SLC500. L'application automate a précédemment été développée. Un bouton trois positions du pupitre opérateur permet de choisir entre trois modes de marche :

### Mode automatique

En mode automatique, le fonctionnement du procédé peut être schématisé comme suit :

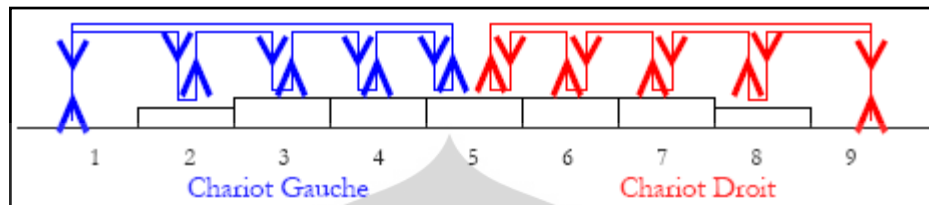


Figure 4.2 Schématique de fonctionnement de la mode automatique

Toutes les pièces passent dans les bacs selon l'ordre 2,3,4,.....,8. L'emplacement 5 est une zone de transition, le chariot gauche y dépose les paniers traités dans les bacs 3 et 4, le chariot droit vient ensuite récupérer le panier dans ce bac pour qu'il soit traité dans les zones 6 et 7 puis déposé en 8.

Ce mode se lance uniquement dans la position initiale (les chariots aux deux extrémités et leurs bras en positions basses), et après un appui sur le bouton de départ cycle.

### Mode manuel

Le mode manuel permet au chariot gauche de se déplacer des bacs 1 à 6 et des bacs 4 à 9 pour le chariot droit. Pour cela, quatre boutons par chariot sont disponibles sur le pupitre de commande. Les déplacements des chariots ne sont autorisés que si les bras sont en position haute ou basse.

### Mode neutre

Dans ce mode, l'état de la maquette est figé : aucun actionneur ne peut être activé (mais la maquette reste sous tension). L'arrêt d'urgence arrête et initialise le programme. Après avoir libéré le bouton d'arrêt d'urgence, il est impératif de repasser par le mode manuel pour reprendre la main sur la maquette. La maquette est équipée d'un capteur anti-collision, conditionnant le déplacement du chariot gauche vers la droite et du chariot droit vers la gauche.

### 4.3. Dessiner l'Interface Graphique

Ce sera dessiner 4 synoptiques dans ce projet et il y a deux synoptiques qu'ils seront afficher au travers de navigateur web en savoir l'interface graphique de Pont Roulant et d'Alarme.

Avant de faire dessiner l'interface graphique de Pont Roulant, tout d'abord on va dessiner deux modelés de synoptique qui seront utiliser quand dessiner l'interface graphique.

#### *Création d'un Modèle de Synoptiques*

Un modèle de synoptique est créé de la même façon que n'importe quelle autre fenêtre excepté que la sauvegarde se fait dans le dossier des modèles de synoptiques. Les éléments de dessin et les animations de ce modèle de synoptique sont entièrement fonctionnels dans tous les synoptiques utilisant ce modèle, mais ils peuvent seulement être édités et modifiés dans le modèle de synoptique.

Le premier modèle de synoptique est « **Background template** ». Il sera utilisé pour dessiner les interfaces graphiques principales du projet.

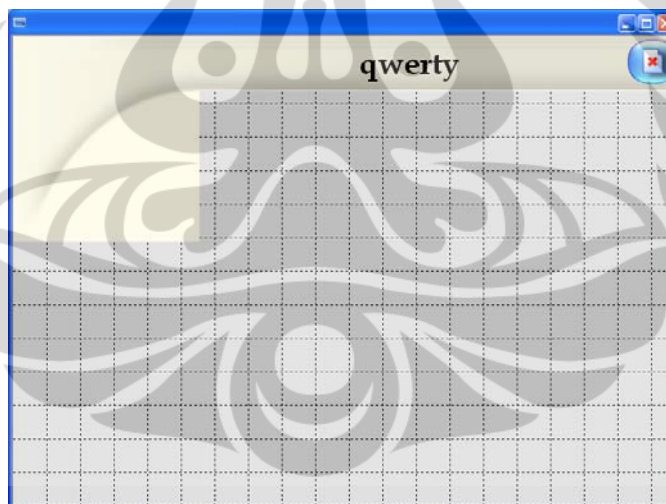


Figure 4.3 Modèle le synoptique Background Template

On va faire deux animations dans cette modèle synoptique, ils sont une l'animation libellée et une l'animation fermeture de la fenêtre. Les propriétés des deux animations sont la suivant :

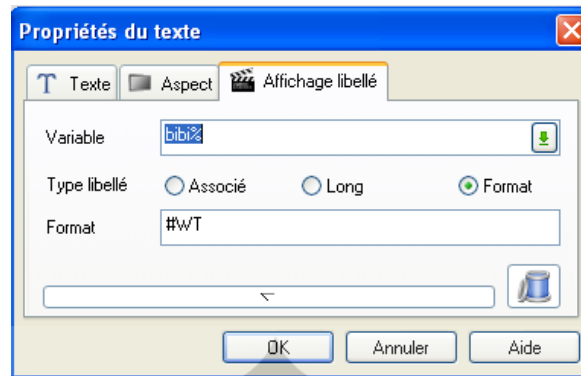


Figure 4.4 Propriétés de l'animation libellée

Un libellé de variables est bilingue et si on change la langue du projet vous devez vous assurer qu'il y a un libellé dans la langue appropriée. Afin d'afficher des informations contextuelles, il faut choisir le bouton d'option Contexte et entrer une chaîne de substitution dans le champ. La chaîne de substitution détermine ce qui est affiché dans le libellé. #WT dans le format de la propriété du texte c'est-à-dire on va afficher le titre de la fenêtre.

L'animation du bouton fermeture de la fenêtre va forçage le programme « EndWebVue ».



Figure 4.5 Propriétés de l'animation fermeture de la fenêtre

Le programme WEBVUE est utilise pendant supervision sur le navigateur web.

```

Sub Main ( )
  @WEBVUE.IEPATH      =      "C:\\Program      Files\\Internet
Explorer\\IEXPLORE.EXE";
End Sub

Sub StartWebVue ( )
  Application          ( "LOAD",          @WEBVUE.IEPATH,
"http://localhost/svhmiweb_stage/");
End Sub

Sub EndWebVue ( )
  Dim cIEpath As Str, cMimic As Str;
  Dim iUser As Integer;

  iUser = GetArg("WEB");
  cMimic = GetArg ("WINDOW");

  If (iUser != 0) Then                                '----- Opened from IE
    cIEpath = @WEBVUE.IEPATH;
    Application ("UNLOAD", cIEpath);
  Else                                                '----- Opened from the
Supervisor
    Window ("OPEN", "Home", "");
    Window ("CLOSE", cMimic, "");
  End If

End Sub

Sub Stats()

  Dim lHandle as Long;
  Dim cWebdata As Str;
  Dim iResult As Integer;

  lHandle = Alloc_Buffer ( 256 );
  iResult = WebVue ( "LIST", lHandle);
  cWebdata = Cget_Buffer ( lHandle, 0, 255);

  @WEBVUE.NUMUSERS = Tos(iResult );

Free_Buffer (lHandle);

End Sub

```

Il y a 4 sub programme dans le programme WEBVUE. Chacun sub programme sera utiliser comme fonction dans l'onglet de forçage programme quand faire d'animation dans le synoptique.

Il existe sub main programme qu'il s'appelle « StartWebVue » et « EndWebVue ». Les deux sub main programme vont utiliser pour commencer et se termine le projet qui a démarré dans le navigateur web.

Le sub main programme « Stats » est nécessaires pour mesure le nombre de client WebVue qui a se connectée. Pour cela, il nécessaire quand déclencher le programme WebVue sur un événementielle du projet.

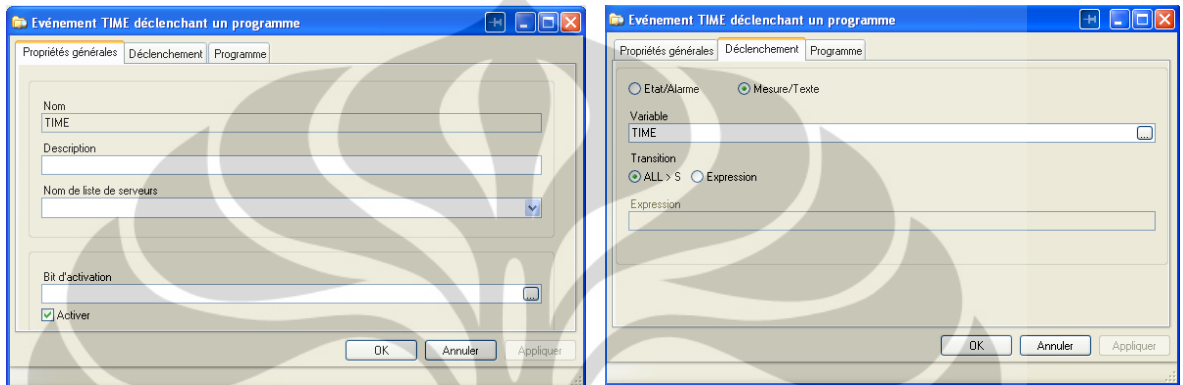


Figure 4.6a Événementielle de la mesure du nombre du client

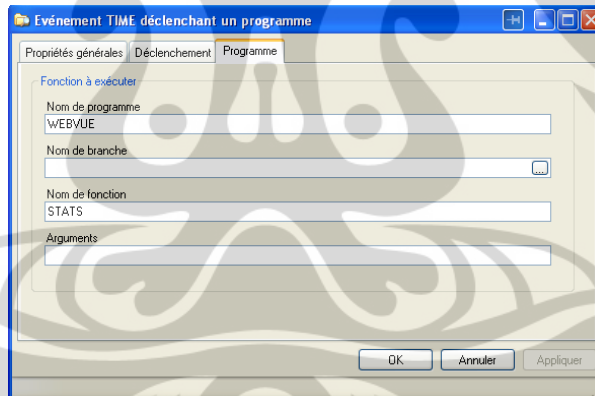


Figure 4.6b Événementielle de la mesure du nombre du client

Le deuxième modèle de synoptique est « **webvue** ». Il sera utilisé pour dessiner les interfaces graphiques qui seront affichés au travers de navigateur web.

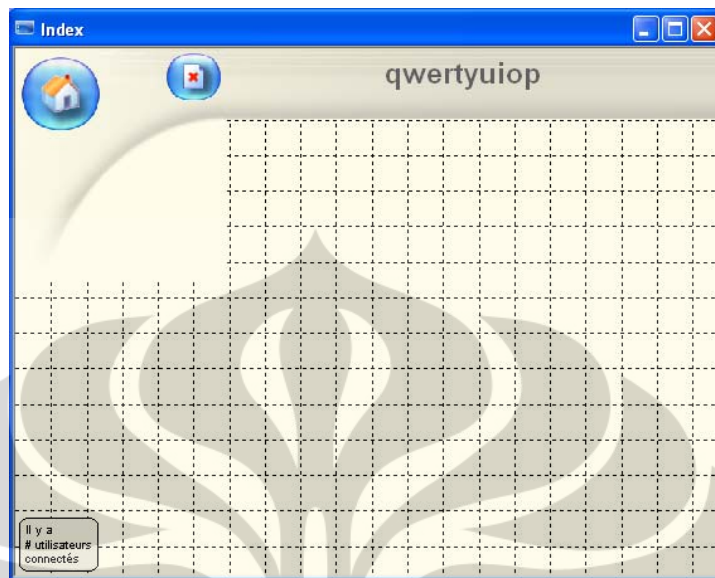


Figure 4.7 Modèle le synoptique webvue

Dans cette modèle, il y a la mesure de nombre de l'utilisateur qui est accès de serveur WebVue.



Figure 4.8 Propriétés d'animation texte #

Le variable mesure NUMUSERS est limité à 10 utilisateurs de WebVue.

### *Création des interfaces graphiques*

Dans la boîte des propriétés de chaque synoptique basé sur un modèle, nous pouvons sélectionner quelle propriété est héritée de ce modèle de synoptique. Nous pouvons également changer le modèle sur lequel le synoptique est basé ou bien tout supprimer.



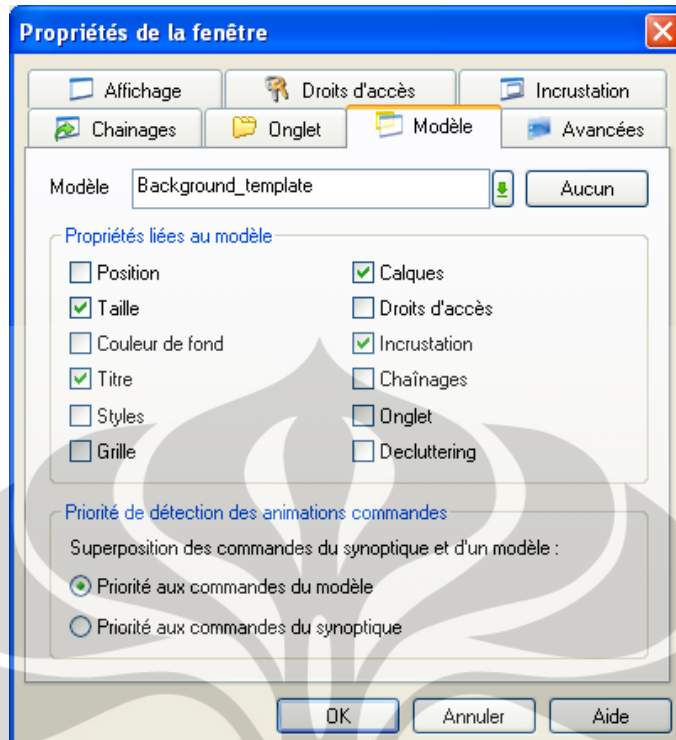



Figure 4.9 Propriétés de la fenêtre du synoptique

Dans cette modèle de synoptique PcVue, on va dessine les interfaces graphiques qui seront utiliser en web service application. Les interfaces graphiques que nous avons dessinées pour web supervision sont la suivant :



Figure 4.10 L'interface graphique principale WebVue

On va faire d'animation d'ouverture l'autre fenêtre en appuyant sur le bouton . Le synoptique qu'il va afficher est le synoptique « **Connect IE** ».

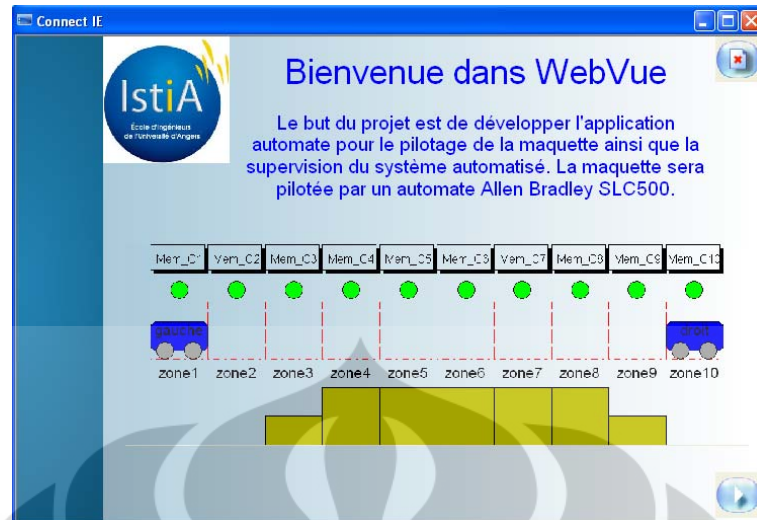



Figure 4.11 L'interface graphique principale du projet Pont Roulant

Ensuite, en appuyant sur le bouton  dans ce synoptique, on va se connecter le projet avec le navigateur web « Internet Explorer » et afficher le synoptique de traitement de surface de Pont Roulant dans ce navigateur.

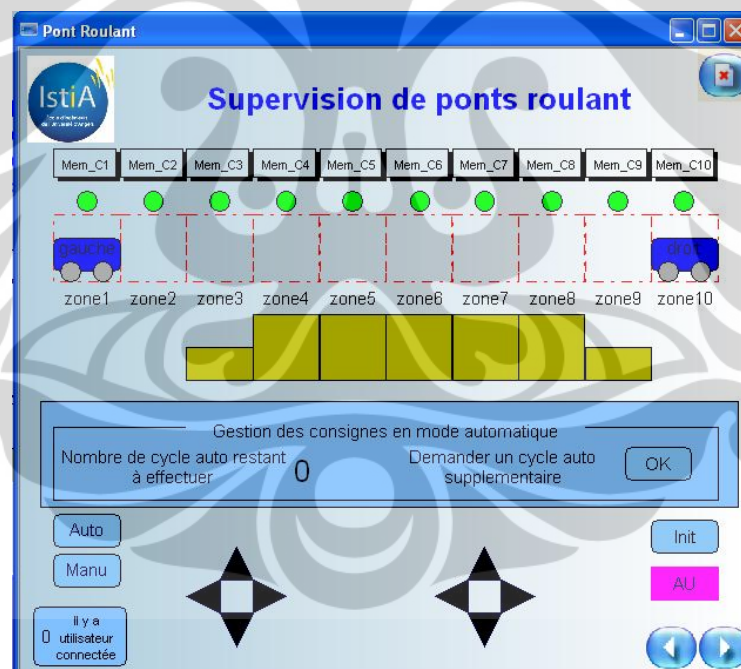



Figure 4.12 Procède de traitement de surface de Pont Roulant

Dans le but d'afficher ce synoptique au travers de navigateur web, on va faire d'animation sur le bouton  dans l'interface principale du projet Pont Roulant.

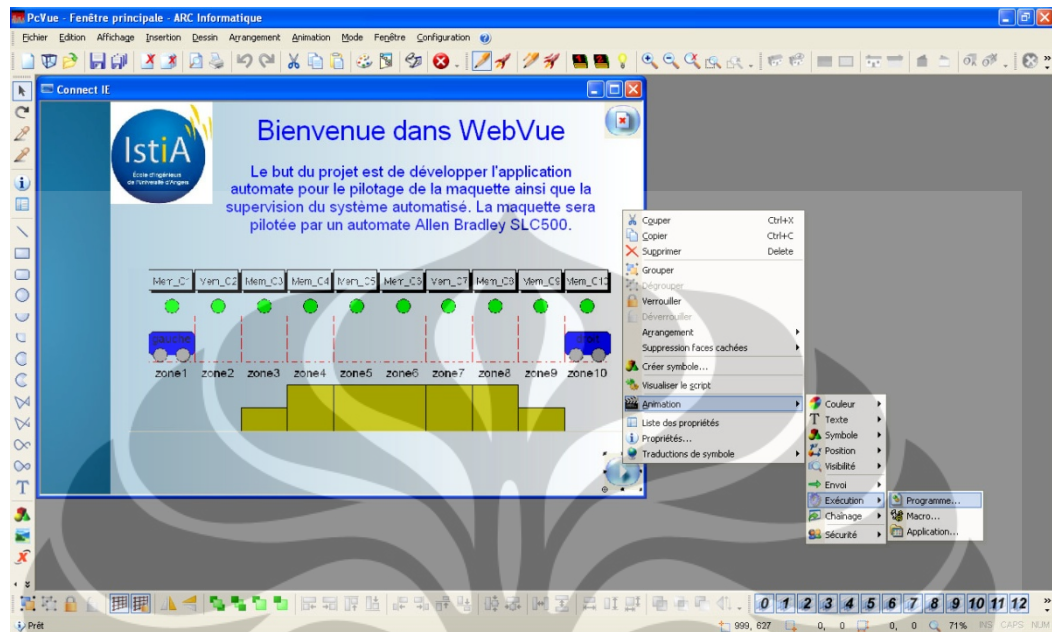


Figure 4.13 Procède de faire d'animation sur le bouton « suivant »

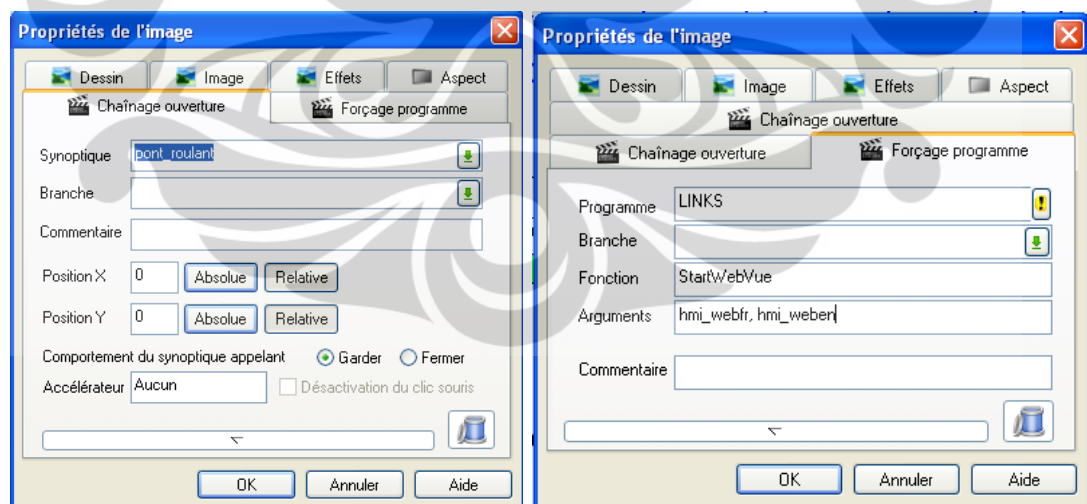



Figure 4.14 Propriétés du bouton « suivant »

Dans l'onglet forçage programme, Cliquer sur l'icône  pour préciser le programme à associer. En suit, préciser la fonction dans ce programme à exécute et les arguments qui sont utilises dans ce programme.

Le programme LINKS est utilisé pour relier le projet de web superviseur avec le Serveur WebVue et après on peut afficher le dans le navigateur web. Données stockées sur le Serveur Webvue peut être appelé via localhost en raison d'Internet Information Services (IIS).

```

Sub StartWebVue()
    Dim cIEpath As Str;
    Dim cProject As Str;
    Dim cURL As Str;

    if (LANGUAGE("GET")==1) then
        cProject=GetArg("ARG1");
    else
        cProject=GetArg("ARG2");
    end if

    cIEpath = @WEBVUE.IEPATH;
    cURL=ADDSTRING("http://localhost/svweb_stage/index.html?project=",cProject);

    Application ("LOAD", cIEpath, cURL);
End sub

Sub StartIISPortal()
    Dim cIEpath As Str;
    cIEpath = @WEBVUE.IEPATH;
    Application ("LOAD", cIEpath, "http://localhost/classes/");
End sub

```

Les variables propres au programme (dite *de travail*) sont déclarées grâce au mot réservé dim:Dim cIEpath As Str;

Ce programme va générer une adresse URL « http://localhost/svweb\_stage/index.html?project= » qui est déclaré dans ce programme et puis suivi d'argument du nom du projet qui est écrit sous l'onglet forçage programme. Ensuite, le programme LINK va activer IIS portail pour accès le répertoire virtuel qui a été créé. classes est un répertoire virtuel qui est utilisé par le client WebVue pour charger l'applet Java. Le dernier synoptique que nous avons fait est Alarme. Le synoptique affiche l'alarme arrêt urgence et anti collision.

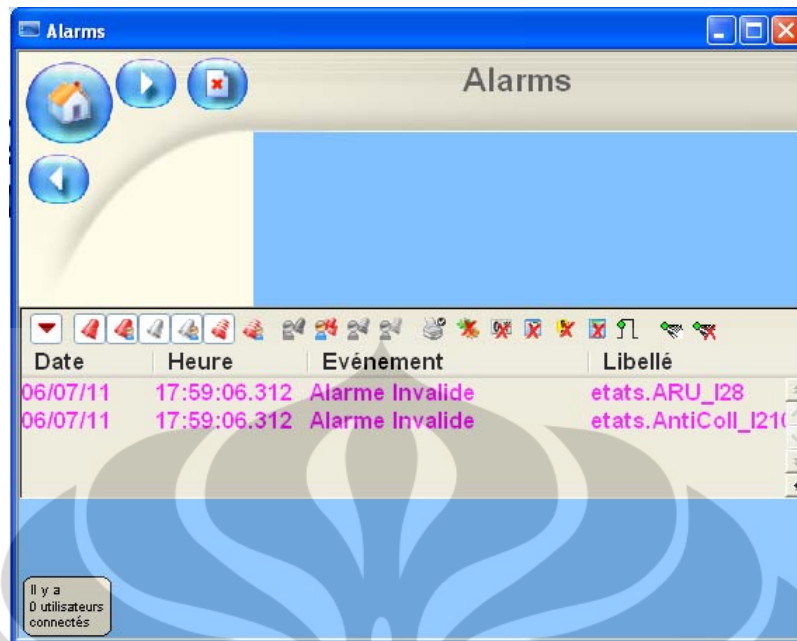


Figure 4.15 Le synoptique Alarme

#### 4.4. Mise en œuvre

A manière d'affiché des synoptiques qui ont été dessiné au travers de navigateur web, on doit mise en œuvre de serveur WebVue.

Pour configurer les propriétés du Serveur WebVue, on sélectionne la commande Configuration.Communication.Serveur WebVue. Les propriétés principales se trouvent dans l'onglet Serveur WebVue.

##### *Paramétrage des Répertoires Virtuels IIS*

La première étape pour tout nouveau projet est de configurer les répertoires virtuels IIS. Sans cela, toutes les autres propriétés de la boîte de dialogue sont indisponibles.

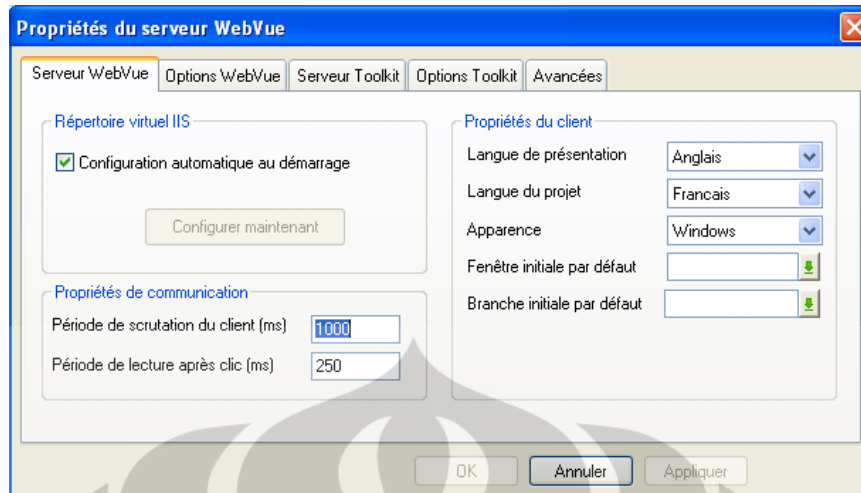


Figure 4.16 Propriétés du Serveur WebVue

Pour paramétrer les répertoires virtuels IIS, cliquer sur le bouton **Configurer Maintenant**. Ceci crée trois répertoires virtuels, ils sont classés, WebPropertyServer et le nom du projet « SVWEB\_STAGE ». Lorsque ceci est fait, les autres propriétés de la boîte de dialogue deviennent accessibles.

Lorsqu'on utilise le bouton Configurer Maintenant, un alias est créé entre l'URL que nous utilisons pour démarrer WebVue et la localisation physique du projet de Supervision. On coche la propriété Configuration automatique au démarrage à la but de chaque fois que le Superviseur est démarré, l'alias est régénéré.

Les propriétés du client déterminent les langues par défaut, l'apparence du Client WebVue ainsi que le synoptique affiché lors de la connexion d'un WebVue au Superviseur.

- Langue de présentation - Spécifie la langue d'affichage du menu WebVue et des outils.
- Langue de projet - Spécifie la langue d'affichage des synoptiques WebVue.
- Apparence - Sélectionne l'apparence du menu système client WebVue, des menus pop-up et l'espace de travail HTML.
- Fenêtre et branche initiales par défaut - Sélectionne la fenêtre, et la branche, affichée dans le client WebVue lors de la première connexion au

Superviseur seulement si la fenêtre et la branche ne sont pas définies dans la configuration utilisateur.

### *Options du Serveur WebVue*

Les options pour le serveur WebVue sont configurées dans l'onglet Options WebVue de la boîte de dialogue Propriétés du serveur WebVue.

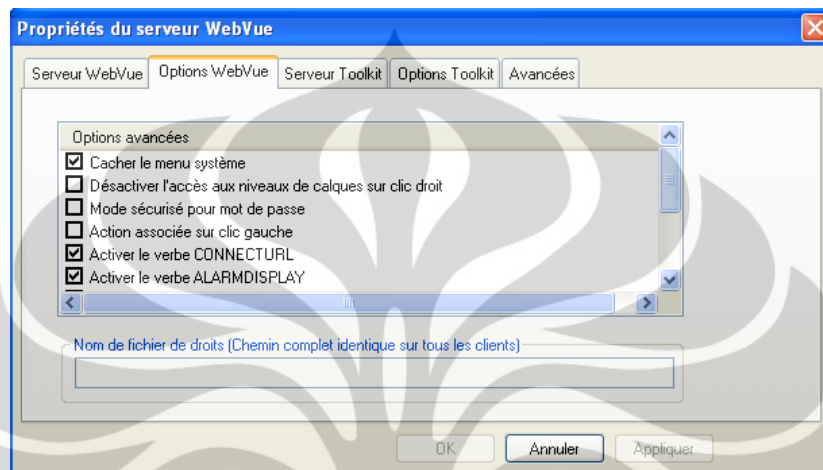


Figure 4.17 Propriétés du Options WebVue

### *Propriétés générales*

- Cacher le menu système - Si coché, le menu WebVue normalement affiché dans les navigateurs est caché.
- Désactiver l'accès aux niveaux de calques sur clic droit - Si coché, la possibilité de sélectionner des calques, à partir du menu pop-up affiché par un clic droit dans le synoptique WebVue, est désactivée.
- Mode sécurisé pour mot de passe - Voir le chapitre Utilisation du Mode Sécurisé pour la Propriété Mot de Passe.
- Action associée sur clic gauche - Si coché, dans une fenêtre d'alarme, l'action associée à une alarme est directement exécutée lors d'un clic gauche sur l'alarme.
- Activer les synoptiques contextuels - Si cochée, permet l'utilisation des fenêtres contextuelles dans WebVue.
- Autoriser le cryptage des données - Si coché, les données transférées entre le client WebVue et le Serveur sont cryptées. Lorsque les données transférées sont cryptées, elles sont moins vulnérables aux accès pirates.

Un symbole cadenas fermé apparaît dans le coin en bas à gauche de la fenêtre de navigation pour confirmer que le lien est protégé.

- Autoriser la compression des données - Si coché, les données transférées entre le serveur WebVue et le client sont compressées en utilisant un algorithme de type ZIP.

*Les propriétés relatives au SCADA Basic*

- Activer le verbe CONNECTURL - Si coché, permet l'utilisation du mode CONNECTURL de l'instruction WEBVUE dans le SCADA BASIC.
- Activer le verbe ALARMDISPLAY - Si coché, permet l'utilisation des modes DOMAIN, NATURE et FILTER de l'instruction ALARMDISPLAY dans une fenêtre d'alarmes d'un synoptique WebVue (les autres modes ne sont pas disponibles).
- Activer le verbe LOGDISPLAY - Si coché, permet l'utilisation des modes DOMAIN, NATURE, FILTER et EXECUTE de l'instruction LOGDISPLAY dans une fenêtre de consignations d'un synoptique WebVue (les autres modes ne sont pas disponibles).
- Activer le verbe TREND - Si coché, permet l'utilisation du mode SETVAR de l'instruction TREND avec une fenêtre de tendance dans un synoptique WebVue.

*Propriété de communication*

- Période de scrutation du client - Période à laquelle les données temps réel du client WebVue sont rafraîchies.
- Lecture après clic - Après un clic de commande dans un synoptique, le client WebVue effectue une requête de lecture pour rafraîchir la page. Ce paramètre permet de régler le temps entre le clic et la lecture.

En outre, on va configurer aussi la configuration utilisateur WebVue. Le synoptique (fenêtre) sélectionné ne sera affiché que si la configuration utilisateur ne spécifie pas de fenêtre initiale. Pour cela, dans l'onglet WebVue de la boîte de dialogue Propriétés du Profil (Configuration.Administrer.Droits), cliquer avec le bouton droit sur le nom du profil.



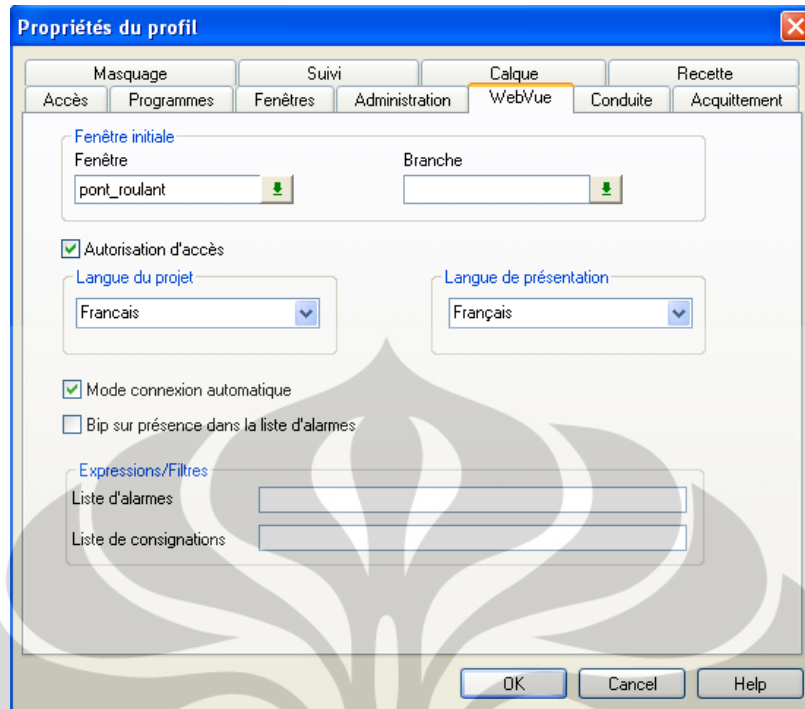


Figure 4.18 Propriétés de la fenêtre initiale

En cochant la propriété de la Mode connexion automatique dans le profil d'un utilisateur, on peut autoriser un utilisateur à démarrer WebVue et à se connecter au Superviseur sans avoir à saisir un nom d'utilisateur et un mot de passe. Pour cela, on doit ajouter les lignes suivantes à la page INDEX.HTML chargée par le navigateur Web lors du démarrage de WebVue.

```
+ "<PARAM name=username value=\"toto\">\n"
+ "<PARAM name=password value=\"toto\">\n"
```

Où « toto » est un nom de compte et un mot de passe d'Administrateur.

#### 4.5. Vérification

Enfin, on va vérifier le Projet à l'aide de deux ordinateurs. Le premier comme WebVue Serveur et l'autre comme un WebVue Client.

Après nous avons le démarrage du projet dans le premier ordinateur, nous allons tester pour accéder à l'URL du projet dans le navigateur web sur le deuxième ordinateur.

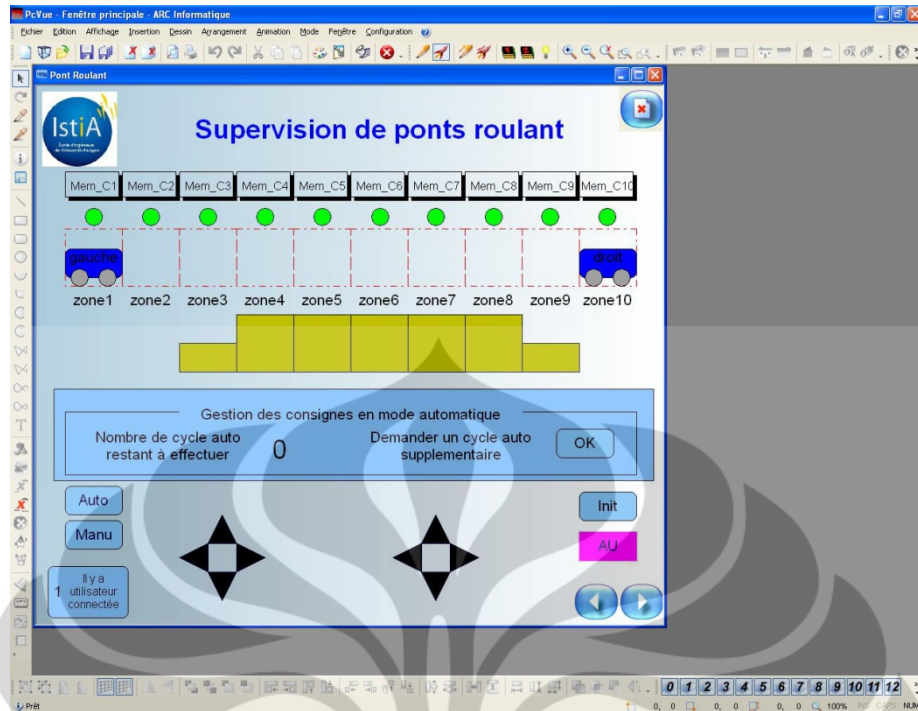


Figure 4.19 L'affichage sur l'ordinateur WebVue Serveur



Figure 4.20 L'affichage sur l'ordinateur WebVue Client

## CHAPITRE 5

### STAGE PROJET DE DEVELOPEMENT

#### 5.1. Standar de Communication OPC

**OPC** pour «OLE for Process Control» est «*un standard d'interface ouvert, basé sur les spécifications fonctionnelles des technologies OLE/COM et DCOM*». Plus précisément, OPC vise à simplifier l'interfaçage entre les composants d'automatisme et les applications informatiques (comme les systèmes de supervision), provenant de différents fournisseurs. Ce standard facilite et normalise l'échange des données entre les applications de contrôle et d'automatisation.

Un serveur OPC peut intégrer différentes fonctionnalités pour la communication avec les applications clientes. Ces fonctionnalités sont réparties dans différentes *interfaces*. Chaque interface regroupe des méthodes et des événements utilisables pour le dialogue entre clients et serveur OPC. Pour notre application nous avons utilisé la spécification OPC Data Access qui est une interface pour l'accès aux données en temps réel. De plus nous utilisons l'interface automation puisque notre supervision a comme langage de développement VB.NET.

Le serveur OPC utilisé est KepServeurEx de KepWare Products.

#### **Ecriture d'un client OPC en VB.Net**

- ✓ Création d'un objet de type OPCServer

```
Dim WithEvents OPCserv As OPCAutomation.OPCServer  
OPCserv = New OPCAutomation.OPCServer
```

- ✓ Connexion au serveur OPC

```
OPCserv.Connect("KEPware.KEPServerEx.V4")
```

- ✓ Création d'un objet de type OPCGroups

```
Dim OPCgrs As OPCAutomation.OPCGroups  
OPCgrs = OPCserv.OPCGroups
```

✓ Création d'un objet de type OPCGroup

```
Dim WithEvents OPCgr As OPCAutomation.OPCGroup
OPCgr = OPCgrs.Add("Groupe")
```

✓ On fixe souscrit l'objet de type OPC Group

```
OPCgr.IsSubscribed = True
```

✓ Création d'un objet de type OPCItems

```
Dim OPCits As OPCAutomation.OPCItems
OPCits = OPCgr.OPCItems
```

✓ On remplit la collection d'items. On crée un tableau d'objets OPC items à l'aide d'ajout items.

```
Dim tabitems(28) As String
Dim cltHandles(28) As Integer

Public Const nomChannel As String = "Channel1"
Public Const nomDevice As String = "Device1"
Public Const nomGroupeCapteurs As String = "Capteurs"
Public Const nomGroupeCommande As String = "Commande"
Public Const nomGroupeEntree As String = "Entree"
```

```
tabitems(1) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." +
"ModeAuto"
tabitems(2) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." +
"ModeManu"
tabitems(3) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." + "ARU"
tabitems(4) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." +
"AntiColl"
tabitems(5) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." +
"BT_DCY"
tabitems(6) = nomChannel + "." + nomDevice + "." + nomGroupeEntree + "." +
"NbCycle"

tabitems(7) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_DépD_CD"
tabitems(8) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_DépG_CD"
tabitems(9) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_Montée_CD"
tabitems(10) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_Desc_CD"
tabitems(11) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_DépD_CG"
tabitems(12) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_DépG_CG"
tabitems(13) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_Montée_CG"
tabitems(14) = nomChannel + "." + nomDevice + "." + nomGroupeCommande + "." +
"BT_Desc_CG"

tabitems(15) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C1"
tabitems(16) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C2"
tabitems(17) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C3"
tabitems(18) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C4"
tabitems(19) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C5"
tabitems(20) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C6"
tabitems(21) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C7"
```

```

tabitems(22) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C8"
tabitems(23) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." + "C9"
tabitems(24) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." +
"C10"
tabitems(25) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." +
"FC_Desc_CD"
tabitems(26) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." +
"FC_Desc_CG"
tabitems(27) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." +
"FC_Montée_CD"
tabitems(28) = nomChannel + "." + nomDevice + "." + nomGroupeCapteurs + "." +
"FC_Montée_CG"

```

- ✓ on crée un tableau des clienthandles associés

```

Dim nbelementOPC As Integer = 28
For i = 1 To nbelementOPC
    cltHandles(i) = i
Next

```

- ✓ Et puis, on fait appel à additems

```

OPCcltItems.AddItems(nbelementOPC, tabitems, cltHandles,
srvHandles, Erreurs)

```

Tableaux clienthandle sont nécessaires d'associer des valeurs des items OPC en cas de changement de leur valeur. On les référencera à l'aide de leurs Handles dans les méthodes d'écriture, lecture, etc.

**Voici le code associé à l'événement Datachange de l'objet de type OPCGroup (cet événement est généré dès qu'un des items change de valeur) :**

```

PrivateSub OPCgr_DataChange(ByVal TransactionID AsInteger, ByVal
NumItems AsInteger, ByRef ClientHandles As System.Array, ByRef
ItemValues As System.Array, ByRef Qualities As System.Array, ByRef
TimeStamps As System.Array) Handles OPCgr.DataChange

'Pour tous les items changés
For i = 1 To NumItems

'si l'item numéro 1 est changé
If ClientHandles(i) = 1 Then
    'code
'si l'item est devenu vrai/false (pour un boolean)
    If (ItemValues(i) = True) Then
        'code
    Else
        'code
    End If
End If
End If

```

On peut écrire une valeur à l'item OPC qu'il a une nouvelle valeur.

```
Private Sub Montée_CG_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventHandler) Handles Montée_CG.MouseDown
    OPCcltItems.Item(13).Write(1)
End Sub

Private Sub Montée_CG_MouseUp(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventHandler) Handles Montée_CG.MouseUp
    OPCcltItems.Item(13).Write(0)
End Sub
```

### Configuration du serveur OPC KepServer :

Channel1 correspond aux propriétés de la communication avec l'automate.

Device1 correspond aux propriétés de l'automate.

Les dossiers sur la partie gauche de l'écran (Capteur, Commande et Entree) correspondant aux collections d'items.

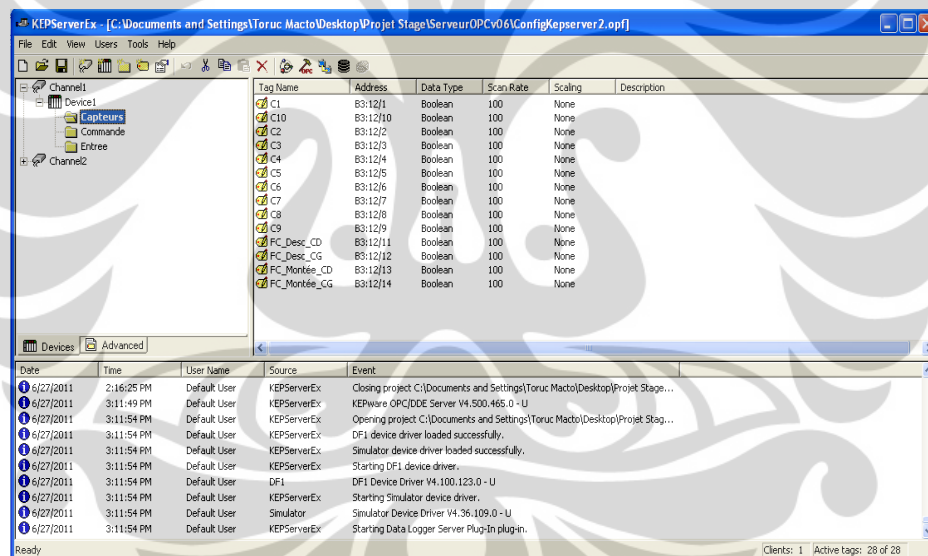


Figure 5.1 Vue KepServerEx

La partie droite de l'écran décrit tous les items de la collection sélectionnée dans la partie gauche. TagName est le nom de l'item. Address est l'adresse de la variable dans l'automate. Data Type est le type de l'item (Boolean, Word..). Scan rate est le taux de rafraichissement. Pour chaque item, on précise également si l'on veut simplement lire la variable de l'automate ou bien également écrire.

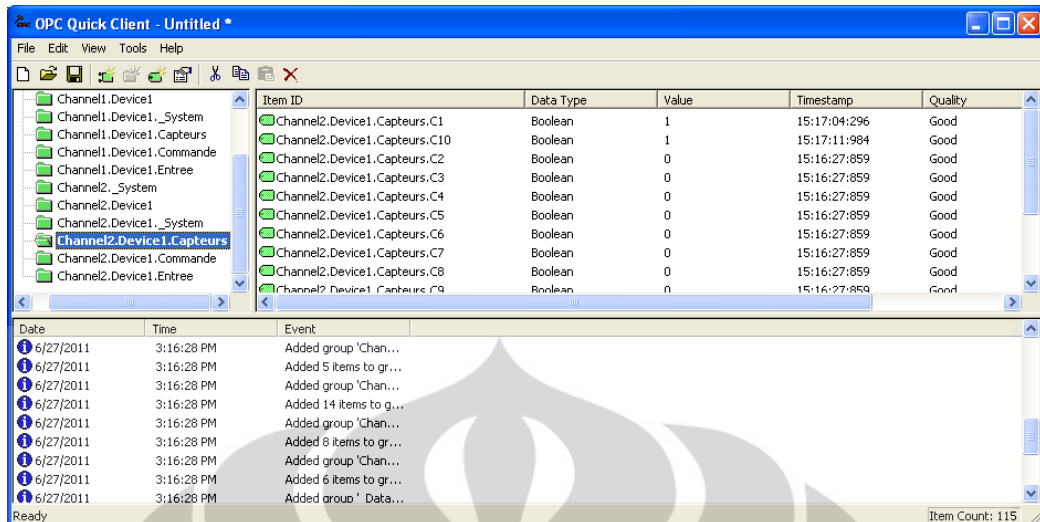


Figure 5.2 Vue OPC Client

## 5.2. Technologie Flash

### 5.2.1. Développement Flash

La supervision Flash a été développée de manière à être intégrée dans l'application VB.Net. Des variables communes sont remontées depuis l'automate. Les animations de supervision ont été réalisées avec le logiciel Macromedia Flash 8.

Dans un premier temps, il est nécessaire de créer des objets graphiques sur des images du scénario à l'aide de la boîte à outils du logiciel. Ces objets graphiques, sont convertis en symboles identifiés chacun par un nom unique.

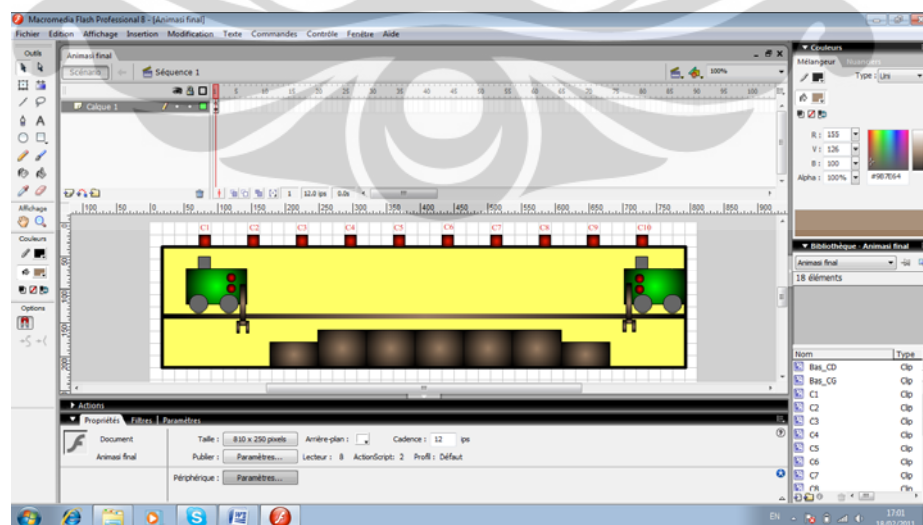


Figure 5.3 Environnement de développement Flash

ActionScript est langage de programmation de Flash. Il permet l'utilisation de fonctions poussées pour ajouter des fonctionnalités supplémentaires au développement d'animation.

#### Exemple : Initialisation des symboles

```
1. //Initialisation des symboles
2. var C1;
3. var C2;
4. var C3;
5. var C4;
6. var C5;
```

#### Exemple : Initialisation des positions

```
18. //initialisation des positions
    var CHG_C1 = 98.3;
    var CHG_C2 = 170.3;
    var CHG_C3 = 243.3;
    var CHG_C4 = 315.3;
    var CHG_C5 = 387.3;
    var CHG_C6 = 459.3;
    var CHG_C7 = 531.3;
    var CHG_C8 = 603.3;
```

#### Exemple : Mouvement des chariots

```
53. if (c6 == 1)
54. {
55.     C7._visible = false;
56.     C6._visible = true;
57.     C5._visible = false;
58.     if (chx == CHG_C5 or chx == CHG_C7)
59.     {
60.         chx = CHG_C6;
61.     }
62.     if (chx2 == CHD_C5 or chx2 == CHD_C7)
63.     {
64.         chx2 = CHD_C6;
65.     }
66. }
```

Une fois l'animation Flash terminée, aussi bien au niveau du graphisme que de la programmation, on l'exporte au format swf que l'on pourra ensuite utiliser dans Visual Studio.



Voici le résultat final de l'animation.

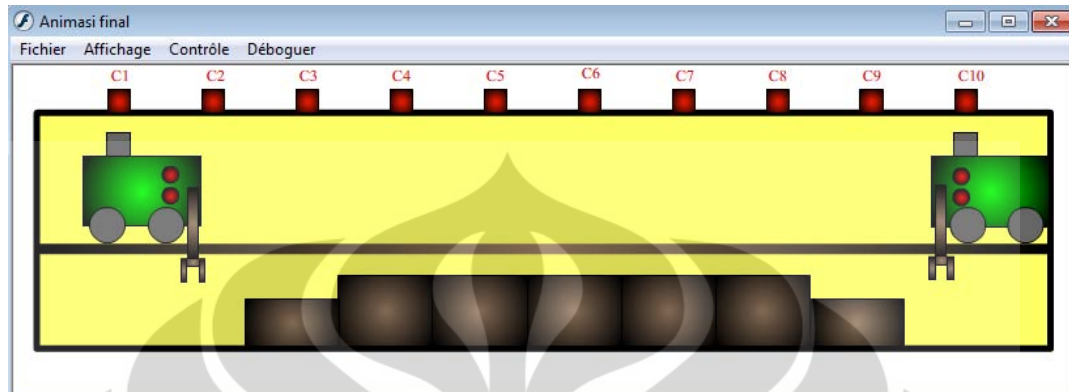


Figure 5.4 L'animation Flash

### 5.2.2. Intégration avec Visual Basic.Net 2008

Voici le code pour l'utilisation de l'animation Flash :

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    AxShockwaveFlash1.AllowScriptAccess = "always"
    AxShockwaveFlash1.Movie = Application.StartupPath &
        "\Supervision.swf"
    AxShockwaveFlash1.Play()
End Sub
```

Il est nécessaire d'ajouter une référence au projet Visual Basic. Il s'agit d'une référence sur un objet COM créé lors de l'installation de Flash.

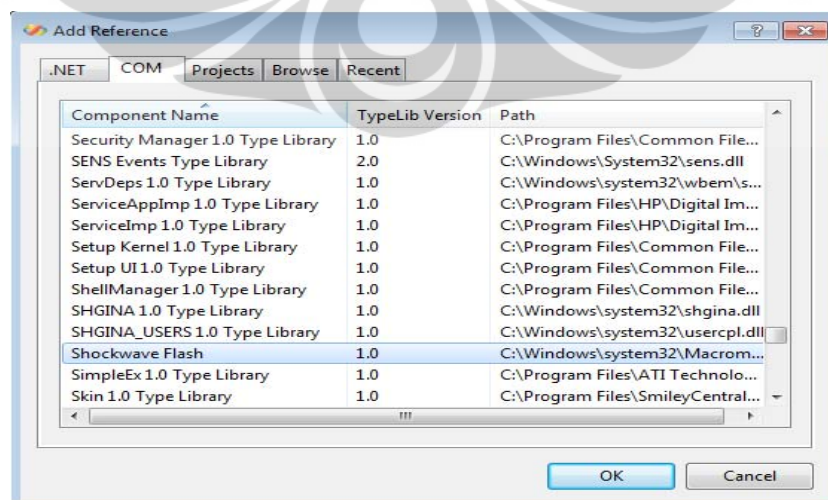


Figure 5.5 Ajout de la référence Flash au projet

Il faut alors ajouter ce composant dans la boîte à outils de Visual Basic.NET. On effectue un clic droit sur la boîte à outils et cliquer sur Choisir les éléments.

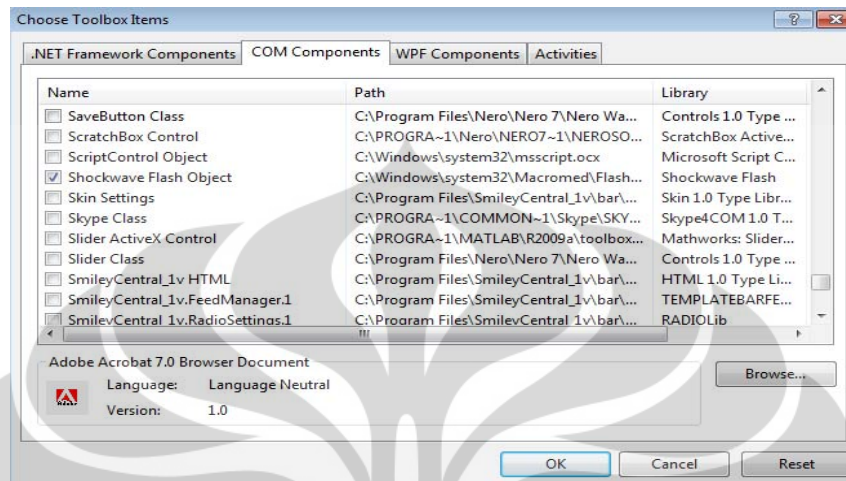


Figure 5.6 Ajout du composant

Ainsi, l'objet Shockwave Flash est ensuite disponible dans la boîte à outils de Visual Studio. Il suffit alors d'ajouter un objet de ce type sur la fenêtre principale de l'application.

### 5.3. Fonctionnement de la supervision

La supervision réalisée en .NET devait répondre à un cahier des charges et doit donc permettre :

- ✓ De visualiser l'état de l'application (position des chariots et de leurs étriers)
- ✓ De sélectionner un mode de fonctionnement si le sélecteur 3 positions du pupitre de commande est en position 0
- ✓ De déplacer les chariots si le mode manuel est sélectionné depuis la supervision
- ✓ De paramétrer et démarrer des cycles de trempage si le mode automatique est sélectionné depuis la supervision
- ✓ De visualiser plusieurs indicateurs destinés à la maintenance : durées cumulées des trempages

- ✓ dans chacun des bacs (pour le remplacement des solutions de trempage), ainsi que le temps total de déplacement de chacun des chariots
- ✓ De visualiser un historique des arrêts d'urgence
- ✓ De visualiser un historique des cycles déjà effectués en mode automatique (n° d'ordre, différentiels de trempage, heure de début, heure de fin).

### 5.3.1. La vue de visualisation du système

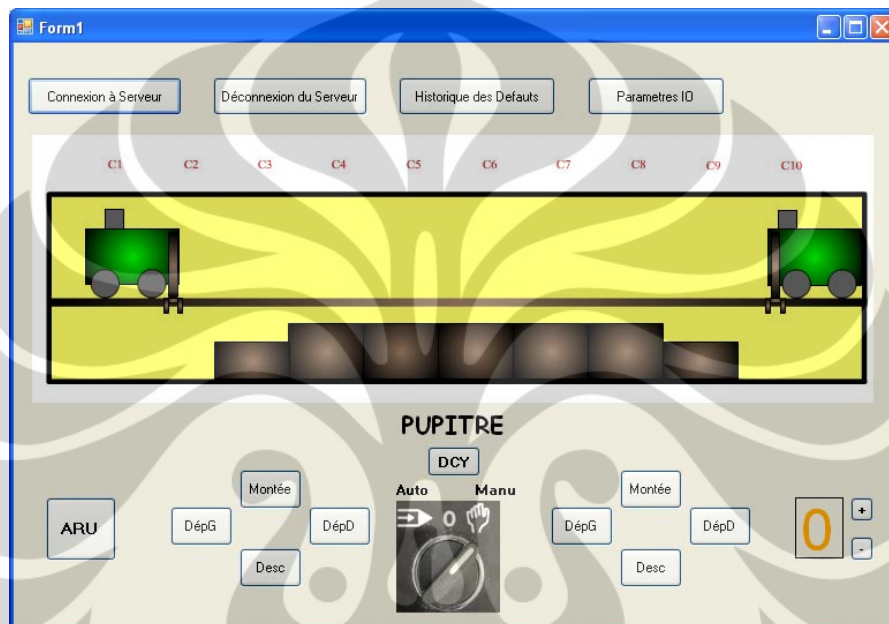


Figure 5.7 Vue de visualisation du système

### 5.3.2. La vue historique des défauts

On va créer une base de données qui est connectée avec le client OPC. La base de données contient deux historiques de défauts, à savoir l'arrêt d'urgence et l'anti collision. L'historique des défauts est constitué du nom des défauts, la date et l'heure et la location des défauts.

Tout d'abord on va ouvrir la connexion entre VB Net et le fournisseur de base de données.

```
Public Sub Connexion_BDD()
    cnxstr = "provider = microsoft.jet.oledb.4.0 ; data source = " & Application.StartupPath & "\..\..\BDD\dbOPC.mdb;"
    cnx = New OleDbConnection
    cnx.ConnectionString = cnxstr
    cnx.Open()
End Sub
```

En suit, on crée la fonction de la requête la base de données.

```
Public Function Requete_BDD(ByVal requete As String) As Integer
    'Création de la requête sql
    sql = requete
    'Création de la commande et on l'instancie (sql)
    cmd = New OleDbCommand(sql)
    'Création du dataadapter (dta) et on l'instancie (cmd)
    dta = New OleDbDataAdapter(cmd)
    'On instancie la commande (cmd) à la connection (cnx)
    cmd.Connection() = cnx
    'On charge le dataset (dts) grace à la propriété fill du
    dataadapter
    (dta)
    dta.Fill(dts, "DEFAULT")
    'On charge la datatable (dtt) grace à la propriété tables du
    dataset
    (dts)
    dtt = dts.Tables("DEFAULT")

    Requete_BDD = dtt.Rows.Count

    Form_Histo.Histo_Grid.DataSource = dtt

    Form_Histo.Histo_Grid.Refresh()

    Form_Histo.Histo_Grid.Columns(0).HeaderText = "ID"
    Form_Histo.Histo_Grid.Columns(0).Width = 35
    Form_Histo.Histo_Grid.Columns(1).HeaderText = "Nom du Defaut"
    Form_Histo.Histo_Grid.Columns(1).Width = 140
    Form_Histo.Histo_Grid.Columns(2).HeaderText = "Date/Heure du
    Defaut"
    Form_Histo.Histo_Grid.Columns(2).Width = 172
    Form_Histo.Histo_Grid.Columns(3).HeaderText = "Position du
    Defaut"
    Form_Histo.Histo_Grid.Columns(3).Width = 180

End Function
```

Pour écrire dans une base de données, on va crée :

```
Public Sub Ecrire_In_BDD(ByVal Nom_Def As String, ByVal Date_Def
As Date, ByVal Position_Def As String)
    Dim requete As String
    Connexion_BDD()
    requete = "INSERT INTO DEFAULT(NOM_DEF,DATE_DEF,POSITION_DEF)
VALUES ('" + Nom_Def + "','" + Date_Def.ToString + "','" +
    Position_Def + "')"
    Requete_BDD(requete)
    Deconnexion_BDD()
    Refresh_Grid(query)
End Sub
```

Dans le cas de Datachange de la valeur des items OPC (ARU et AntiColl), cette méthode va écrire l'événement dans la base de données.

ID	Nom du Defaut	Date/Heure du Defaut	Position du Defaut
15	ARU	6/22/2011 10:45 PM	Maquette
14	ARU	6/22/2011 10:45 PM	Maquette
13	ARU	6/22/2011 10:45 PM	Maquette
12	ARU	6/22/2011 10:33 PM	Maquette
11	ARU	6/22/2011 10:31 PM	Maquette
10	ARU	6/22/2011 10:29 PM	Maquette
9	ARU	6/22/2011 10:26 PM	Maquette
8	ARU	6/22/2011 10:20 PM	Maquette
7	ARU	6/21/2011 2:59 PM	Maquette
6	ARU	6/16/2011 1:38 PM	Maquette
5	ARU	6/16/2011 1:33 PM	Maquette
4	ARU	6/16/2011 1:30 PM	Maquette
3	ARU	6/16/2011 1:28 PM	Maquette
2	Anti-Collision	6/9/2011 12:44 PM	Maquette
1	ARU	6/9/2011 12:44 PM	Maquette

Figure 5.8 Vue des Historique des Défauts

### 5.3.3. L'affichage de changement les valeurs de Paramètre IO

Tout changement aux items de client OPC sera affiché dans un tableau de Paramètres IO. Pour écrire de changement des items OPC dans le tableau de Paramètres IO, on va crée :

```
Dim ItemOPC As ListViewItem

If ClientHandles(i) = 1 Then
    ItemOPC =
    Form_ParamIO.ListView_Entrees.Items.Add("ModeAuto", 0)
    ItemOPC.SubItems.Add("I:2/6")
    ItemOPC.SubItems.Add(ItemValues(i).ToString)
End if

If ClientHandles(i) = 2 Then
    ItemOPC =
    Form_ParamIO.ListView_Entrees.Items.Add("ModeManu", 1)
    ItemOPC.SubItems.Add("I:2/7")
    ItemOPC.SubItems.Add(ItemValues(i).ToString)
End if
```

Entrees			Commande			Capteurs		
Nom	Adresse	Etat	Nom	Adresse	Etat	Nom	Adresse	Etat
AntiColl	I:2/10	False	BT_DepD_CD	B3:10/0	False	C1	B3:12/1	True
ARU	I:2/8	True	BT_DepD_CG	B3:10/4	False	C10	B3:12/10	True
ModeAuto	I:2/6	False	BT_DepG_CD	B3:10/1	False	C2	B3:12/2	False
ModeManu	I:2/7	True	BT_DepG_CG	B3:10/5	False	C3	B3:12/3	False
Nb_Cycle	N:7/0	0	BT_Desc_CD	B3:10/3	False	C4	B3:12/4	False
			BT_Desc_CG	B3:10/7	False	C5	B3:12/5	False
			BT_Montée_CD	B3:10/2	False	C6	B3:12/6	False
			BT_Montée_CG	B3:10/6	False	C7	B3:12/7	False
						C8	B3:12/8	False
						C9	B3:12/9	False
						FC_Desc_CD	B3:12/11	False
						FC_Desc_CG	B3:12/12	False
						FC_Montée_CD	B3:12/13	False
						FC_Montée_CG	B3:12/14	False

Figure 5.9 Vue Paramètre IO

## CHAPITRE 6

### APPLICATION GWT POUR LA SUPERVISION WEB

#### 6.1. Méthodologie du Projet

Le processus de développement implique la suivante

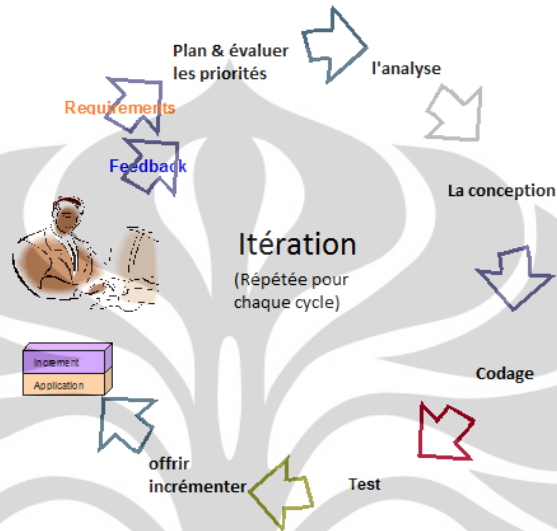


Figure 6.1 Méthodologie du projet

##### 6.1.1. L'examen des exigences fonctionnelles

Initialement, nous souhaitons que l'application de supervision WEB à faire quatre choses :

- Fournir aux utilisateurs la capacité de surveillance de la maquette.
- Afficher l'animation de la maquette
- Fournir aux utilisateurs la possibilité de contrôler la maquette
- Afficher un historique de données d'arrêt urgence.

Notre projet s'occupe de la présentation des différentes pages. Nous avons décidé de créer 5 pages:

##### 1. La page d'identification

Identifier les utilisateurs qui voudraient accéder aux différentes pages de supervision.

##### 2. La page d'accueil

Cette page fournit des informations pour chacune des autres pages et des onglets qui peuvent accéder aux différentes pages de notre projet.

3. La page de supervision

Cette page affiche la supervision du système et les boutons qui peuvent contrôler le système.

4. La page d'historique d'arrêt urgence

Cette page affiche les données d'arrêt urgence.

5. La page de déconnexion

Cette page fournit des informations à quitter le site et les boutons qui peuvent accéder à la première page de notre projet.

### 6.1.2. Identifier les éléments de la conception de l'interface

Nous décidons que nous besoin de ces éléments d'interface utilisateur:

- **la page d'identification**

Une boîte d'entrée pour entrer dans l'identification de l'utilisateur

Un bouton pour faire le lien avec d'autres pages.

Addition :

Coup de tête et image d'architecture du système

Sélection des widgets GWT pour mettre en œuvre les éléments d'interface

élément de l'interface	GWT mise en œuvre
Une boîte d'entrée	<a href="#">TextBox</a> widget
Un bouton	<a href="#">Button</a> widget
Coup de tête	HTML statique dans la page d'accueil HTML
Image d'architecture du système	fichier d'image référencée à partir de la page d'accueil HTML

- **la page d'accueil**

3 onglets pour faire le lien avec d'autres pages.

Addition :

Coup de tête et image de maquette

Sélection des widgets GWT pour mettre en œuvre les éléments d'interface

élément de l'interface	GWT mise en œuvre
Trois onglets affichent une rangée d'onglets cliquables	<a href="#">TabLayoutPanel</a> widget
Coup de tête	HTML statique dans la page d'accueil HTML
Tête supplémentaires	HTML statique dans la page d'accueil HTML
Image de maquette	fichier d'image référencée à partir de la page d'accueil HTML

- **la page de supervision**

Mise en animation

Une boîte pour afficher le nombre de cycles

5 boutons carrés (Auto, Manu, Init, AU, OK Nombre du cycle)

8 boutons image triangle (4 chariot gauche, 4 chariot droit : monte, descende, gauche, droit)

Addition :

Coup de tête

Sélection des widgets GWT pour mettre en œuvre les éléments d'interface

élément de l'interface	GWT mise en œuvre
Mise en animation	<a href="#">AnimatedLayout</a> widget
Une boîte	<a href="#">TextBox</a> widget
5 boutons carrés	<a href="#">Button</a> widget
8 boutons image triangle	<a href="#">Button Image</a> widget
Coup de tête	HTML statique dans la page d'accueil HTML
Image d'architecture du système	fichier d'image référencée à partir de la page d'accueil HTML



- **la page d'historique d'arrêt urgence**

Table

Addition :

Coup de tête

Sélection des widgets GWT pour mettre en œuvre les éléments d'interface

élément de l'interface	GWT mise en œuvre
Une table pour contenir les données	<a href="#">FlexTable</a> widget
Coup de tête	HTML statique dans la page d'accueil HTML

- **La page de déconnexion**

Un bouton pour faire le lien avec de première page.

Addition :

Coup de tête et tête supplémentaires pour les informations déconnectées

Sélection des widgets GWT pour mettre en œuvre les éléments d'interface

élément de l'interface	GWT mise en œuvre
Un bouton	<a href="#">Button</a> widget
Coup de tête	HTML statique dans la page d'accueil HTML
Tête supplémentaires	HTML statique dans la page d'accueil HTML

### 6.1.3. Programmation (Codage)

Un projet GWT est composé de deux sections. Une section client et une section serveur. GWT fournit ses propres appels de procédure distante (RPC) qui permettent au client de GWT pour appeler les méthodes côté serveur.

Pour faciliter la conception, nous avons intégré l'outil WindowsBuilder à Eclipse. On peut donc ouvrir le fichier *.java* qui contient les éléments graphiques du page web, avec WindowsBuilder en effectuant un click droit sur le fichier, puis ouvrir le fichier avec WindowsBuilder.

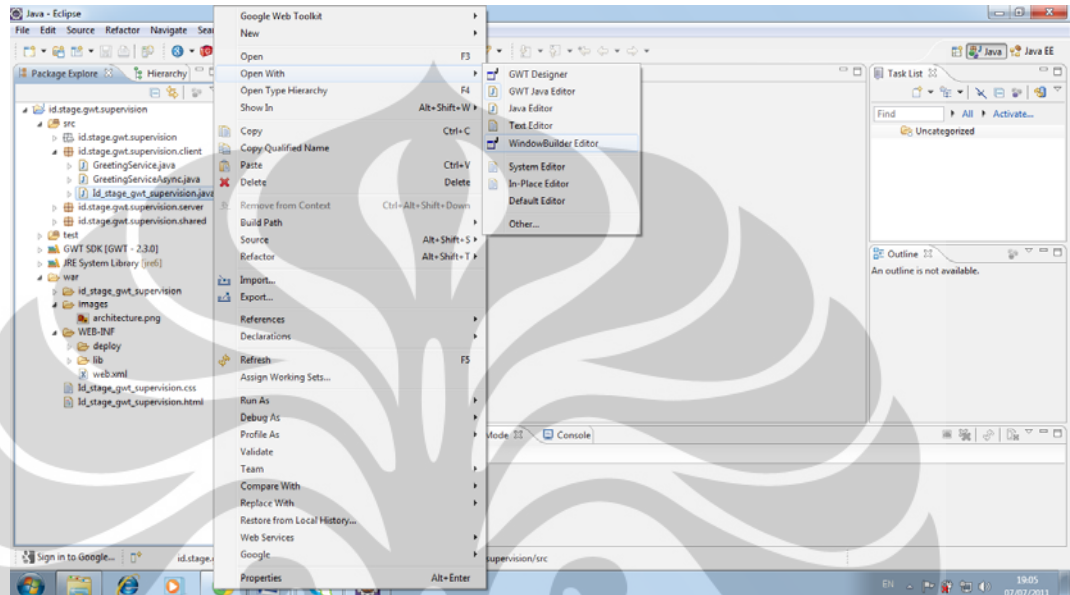


Figure 6.2 Ouverture de WindowsBuilder Editor

Cela nous ouvre un concepteur graphique qui intègre une barre d'outils. Elle permet l'utilisation d'objets graphique comme les boutons, barre de menus, etc.

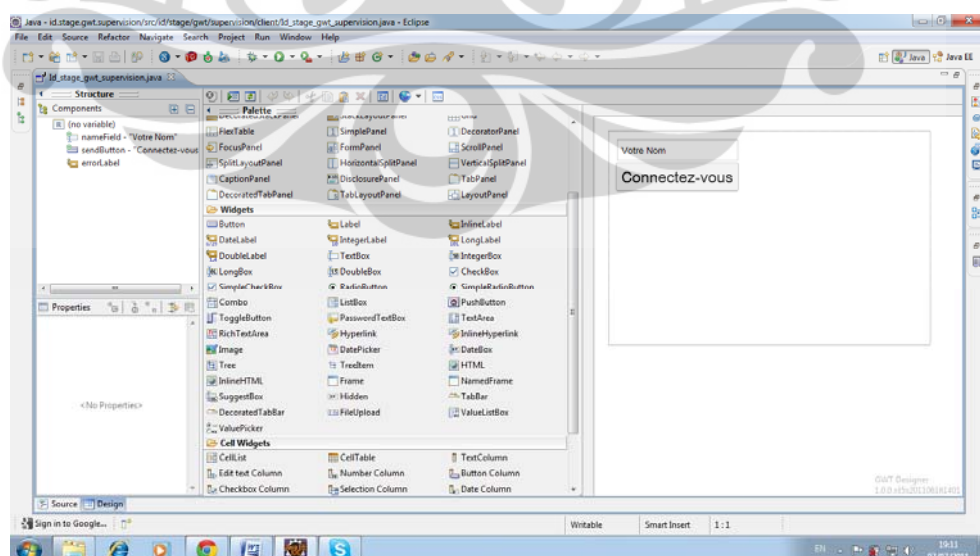


Figure 6.3 Les composant graphique que l'on a avec GWT

Dans les premières étapes de la programmation, chaque page est composé une par une dans un fichier différent. Nous espérons après l'apparition de chaque page sera formé, il est ensuite réalisé la connexion entre une page à l'autre ainsi que l'activation de la fonction de supervision.

## 6.2. Vue sur Les Pages du Programme

Pour exécuter l'application, cliquez droit sur le projet et sélectionnez Run as-> "Web application".

Cela ouvre une nouvelle vue "Development Mode". Copiez l'URL de ce point de vue.

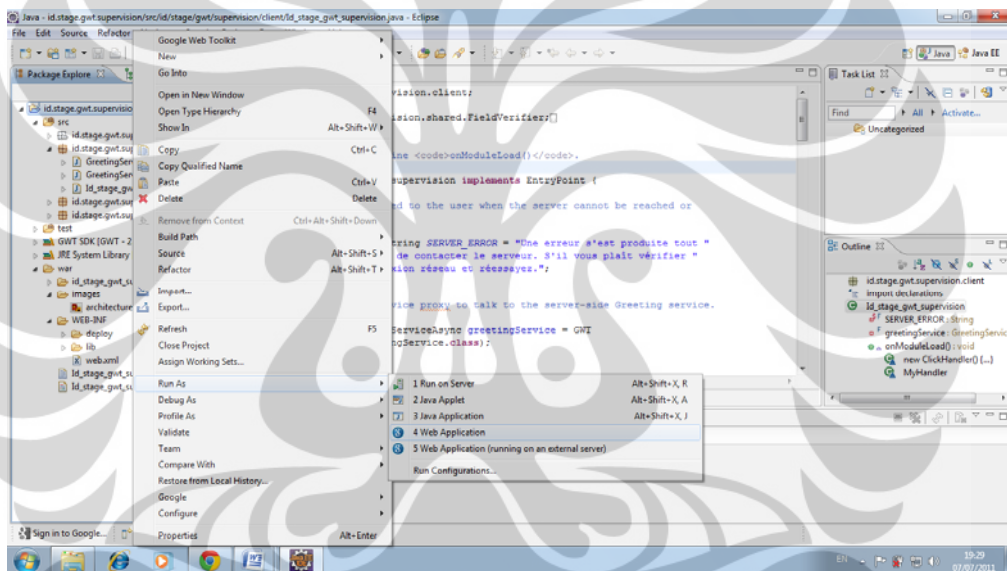


Figure 6.4 Exécuter l'application

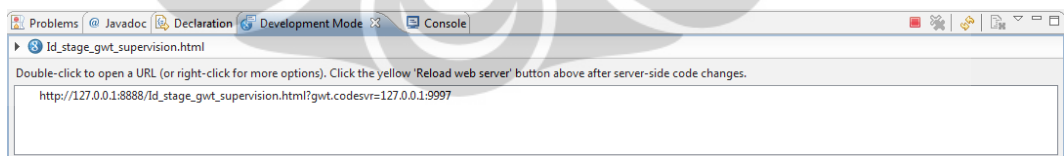


Figure 6.5 Affichage développement mode après l'exécutable

La vue est générée pour la première page (voir ci-dessous).



Figure 5.6 Affichage première page

La vue est générée pour la deuxième page (voir ci-dessous).

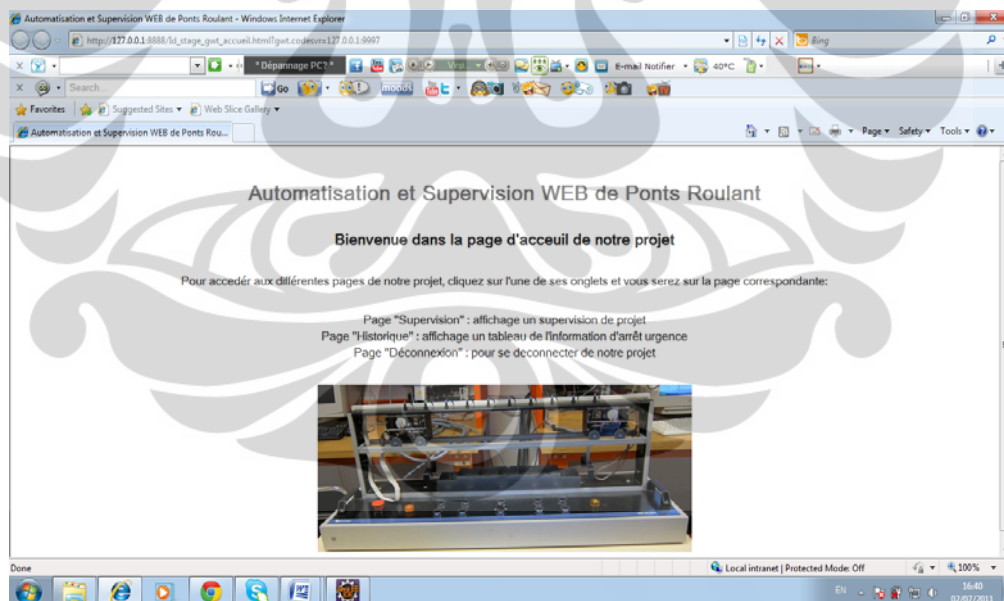


Figure 5.7 Affichage deuxième page

Pour la page supervision et la page arrêt d'urgence, nous n'avons pas fini de créer les affichages à cause de difficultés avec l'animation et connexion à la base de données.

La vue est générée pour la dernière page (voir ci-dessous).

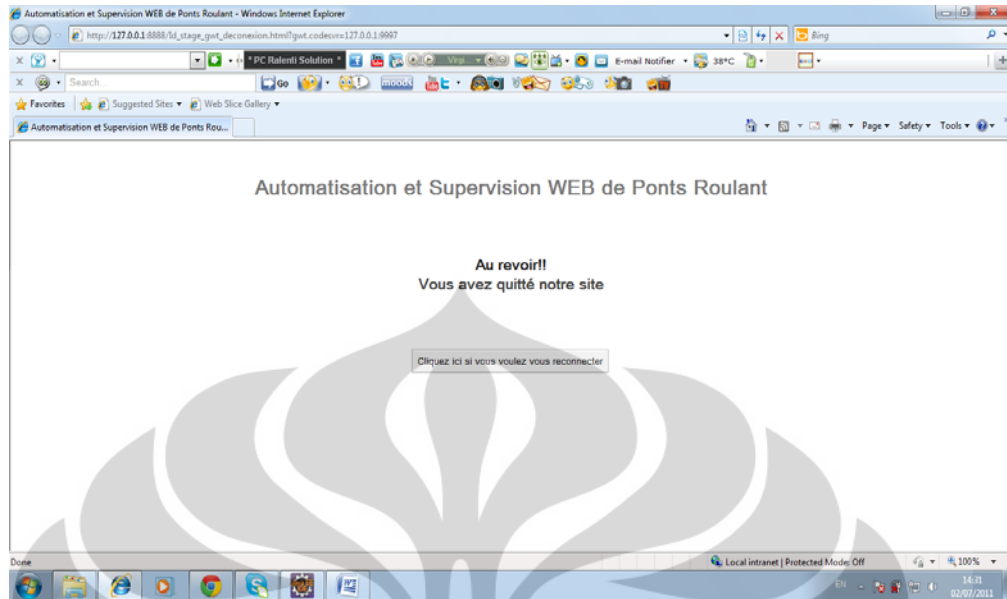


Figure 5.8 Affichage cinquième page

### 6.3. Problèmes Rencontrés

Une des difficultés majeures rencontrées a tout au long du stage utilise GWT à démarrer avec de nouveaux outils représente une longue d'initiation (recherche d'information, prise en main des possibilités...)

Habitué avec le logiciel nécessitait plus de temps à cause de manque connaissance de programmation

## CHAPITRE 7

### CONCLUSION

Pour conclure le projet du stage que nous avons réalisé fut très enrichissant. Ce projet nous a permis de mettre en œuvre de nombreux aspects techniques. Pour la supervision web qui utilise PcVue-WebVue, nous avons réussi à atteindre l'objectif posé. Nous avons réalisé la supervision web de ponts roulant avec succès. Dans le WebVue utilisation, dont l'un doit être considéré est l'étape d'installation du logiciel.

Pour la supervision web qui utilise Google Web Toolkit (GWT), nous avons pris de nouvelles technologies. Mais malheureusement, nous n'avons pas pu réaliser l'objectif posé. C'est parce que nous manquons notre connaissance des langages de programmation. Dans l'utilisation de google-plugin dans Eclipse, nous avons eu la chance de travailler sur un projet comportant une grande richesse en matière d'étude (java/J2EE, .html, .xml, .css). En effet, le problème de démarrer avec de nouveaux outils représente une longue d'initiation (recherche d'information, prise en main des possibilités...)



# **LAMP IRAN**

## **WebVue Tutoriel**

## Le But du projet :

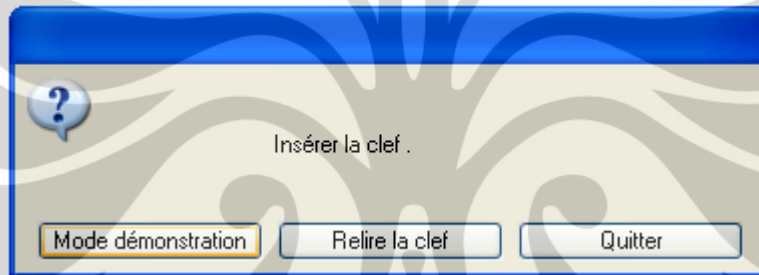
Réaliser une supervision avec PcVue utilisant « WebVue ».

## Étape pratique :

1. Créer un nouveau projet PcVue.

La première fois que le superviseur est lancé après son installation, il apparaît de la fenêtre de chargement (l'Observateur d'événements) que les messages d'état affichent et l'erreur alors que le logiciel charge les gestionnaires différents.

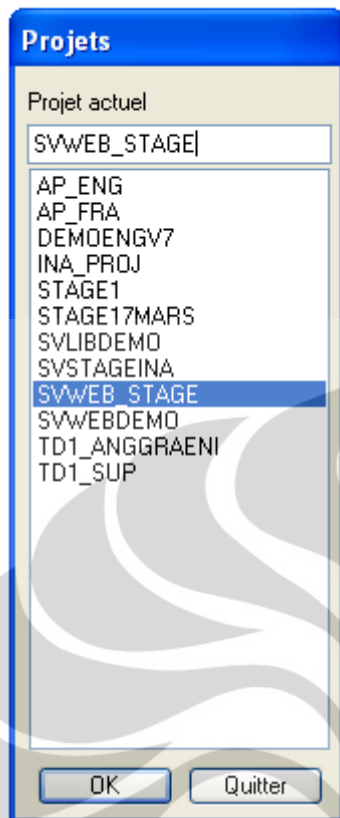
Si vous essayez de démarrer le logiciel sans une clé de protection, une boîte de dialogue vous demandera de l'insérer.



- Pour exécuter le superviseur en version de démonstration, sélectionnez la mode démonstration.
- Pour exécuter le mode approprié pour la clé de protection, insérer la clé et sélectionnez Relire la clef.
- Pour annuler le démarrage, sélectionnez Quitter.

Après son installation, il crée automatiquement deux projets par défaut pour chacune des deux langues choisies. Par exemple, si les langues choisies étaient français et anglais, les projets AP\_FRA et AP\_ENG serait créé. Vous pouvez créer un nouveau projet ou bien sélectionner un projet existant en utilisant une boîte de dialogue affichée par la commande Projets du sous-menu Projet.





#### Création d'un nouveau projet

1. A partir du sous-menu Projets afficher la boîte de dialogue Projets.
2. Saisir le nom du nouveau projet dans le champ texte situé en haut de la boîte. Les noms de projets sont limités à 12 caractères.
3. Sélectionner le bouton de commande OK. Le superviseur crée le nouveau projet et s'arrête.
4. Relancer et redémarrer le Superviseur. Le nouveau projet doit être sélectionné.

#### Choix d'un projet

1. A partir du sous-menu Projets afficher la boîte de dialogue Projets.
2. Choisir le projet désiré en double cliquant sur son nom dans la liste.
3. Sélectionner le bouton de commande OK.
4. Arrêter et redémarrer le Superviseur.

Créer un nouveau nom du projet PcVue par exemple **>SVWEB\_STAGE**

2. Copier les images et les données de SVWEBDEMO dans SVWEB\_STAGE

#### Copier les images

C:\Program files\ARC Informatique\PcVue 8.10\Usr\SvWebDemo\B  
 Copie à C:\Program files\ARC Informatique\PcVue  
 8.10\Usr\SVWEB\_STAGE\B

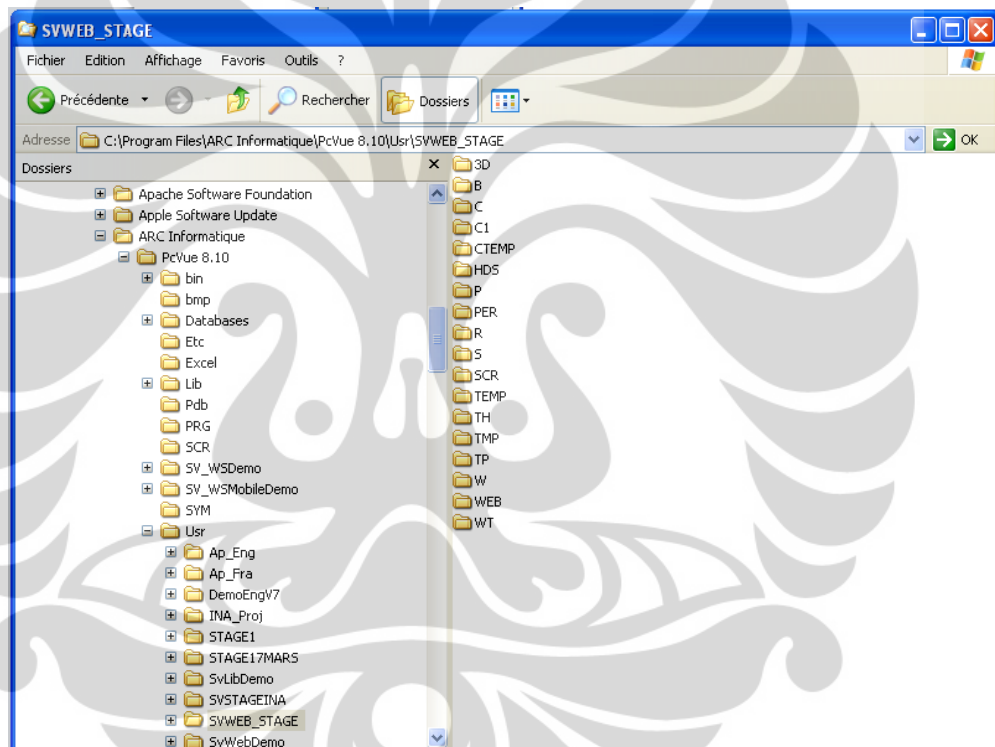
Copier les modèle

C:\Program files\ARC Informatique\PcVue 8.10\Usr\SvWebDemo\WT  
Copie à C:\Program files\ARC Informatique\PcVue  
8.10\Usr\SVWEB\_STAGE\WT

Copier les images

C:\Program files\ARC Informatique\PcVue 8.10\Usr\SvWebDemo\S  
Copie à C:\Program files\ARC Informatique\PcVue  
8.10\Usr\SVWEB\_STAGE\S

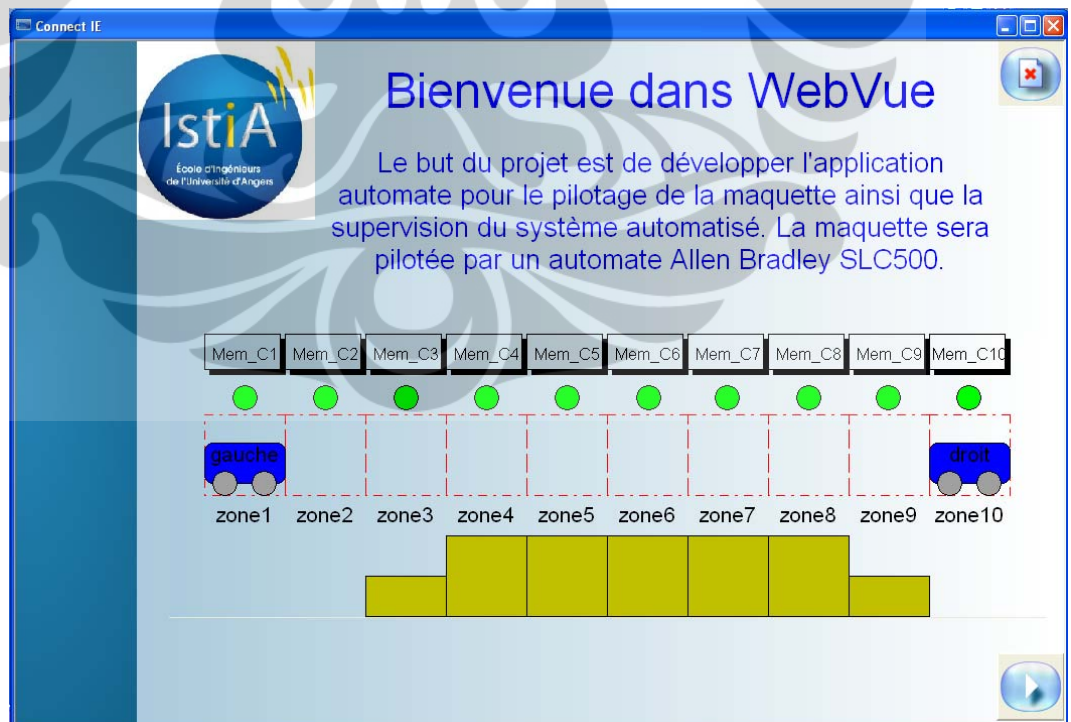
L'objectif des façons sont réutiliser les images et les modèles qui sont utilisé dans le projet web demo de Web Superviseur. Ces seront utiliser pour dessiner les interfaces graphique dans ce projet.



Fermer le programme et recommencer l'ouverture du nouveau Projet créé précédemment.

### 3. Créer 4 nouvelles pages graphiques

Dans cette synoptique PcVue, on peut créer une page graphique.



**Pont Roulant**

**Supervision de ponts roulant**

Mem\_C1 Mem\_C2 Mem\_C3 Mem\_C4 Mem\_C5 Mem\_C6 Mem\_C7 Mem\_C8 Mem\_C9 Mem\_C10

zone1 zone2 zone3 zone4 zone5 zone6 zone7 zone8 zone9 zone10

Gestion des consignes en mode automatique

Nombre de cycle auto restant à effectuer: 0

Demander un cycle auto supplémentaire

Auto

Manu

Il y a 0 utilisateurs connectés

Init

AJ

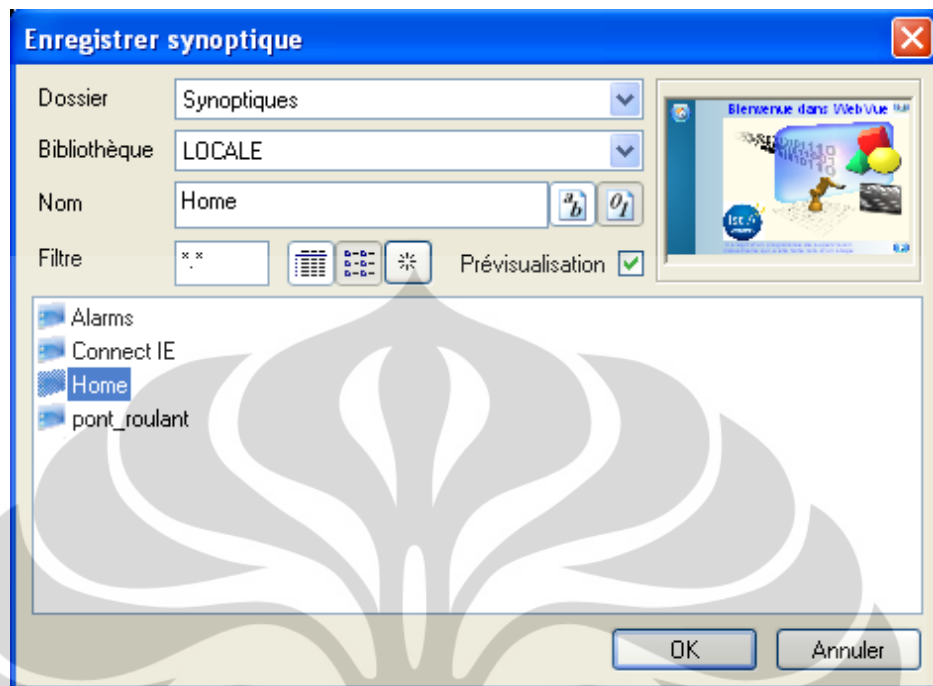
**Alarms**

Alarms

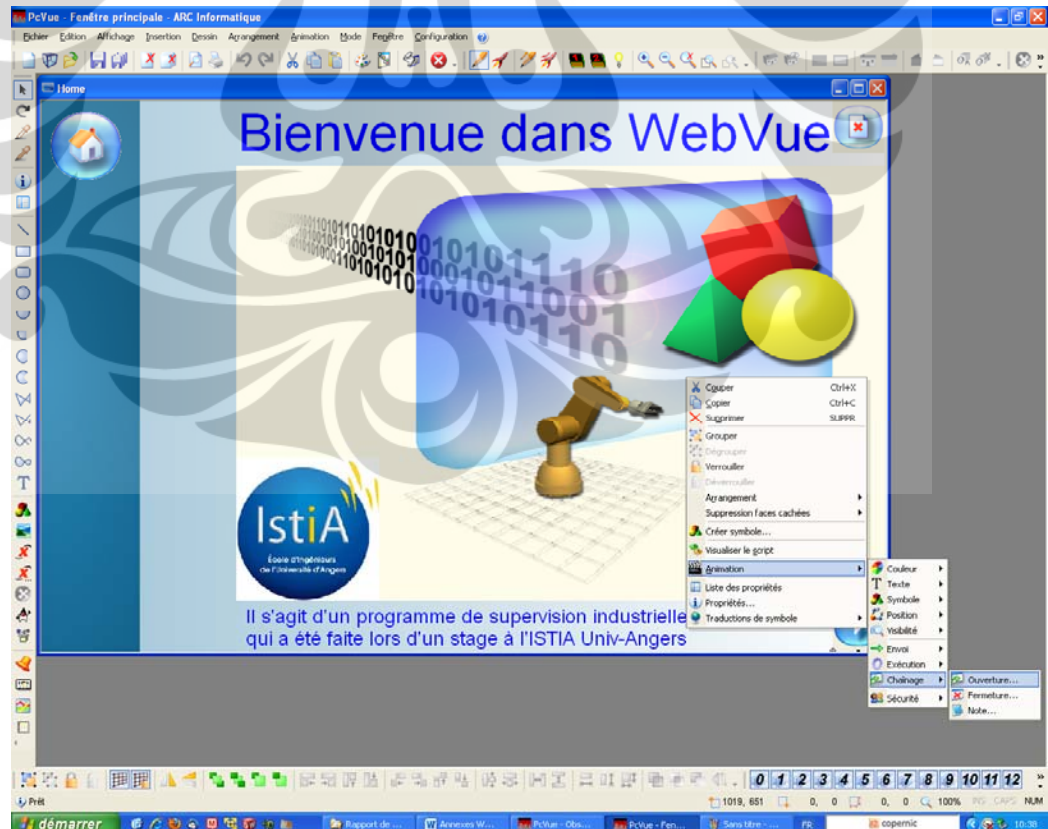
Date	Heure	Évènement	Libellé
03/07/11	22:09:19.109	Alarme Invalide	etats.ARU_I28
03/07/11	22:09:19.109	Alarme Invalide	etats.AntiColl_I210

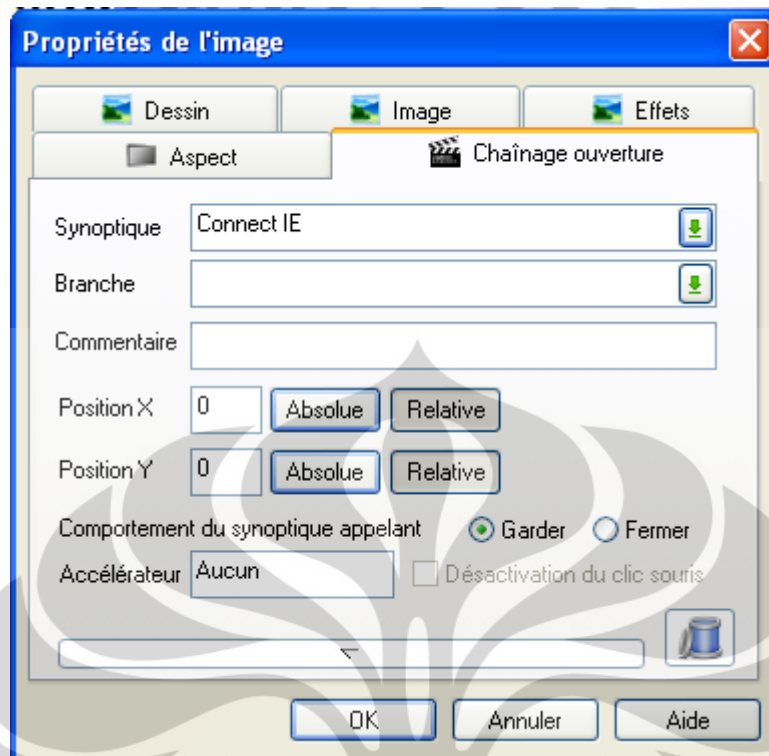
Il y a 0 utilisateurs connectés

4. Enregistrer les synoptiques dans la bibliothèque LOCALE

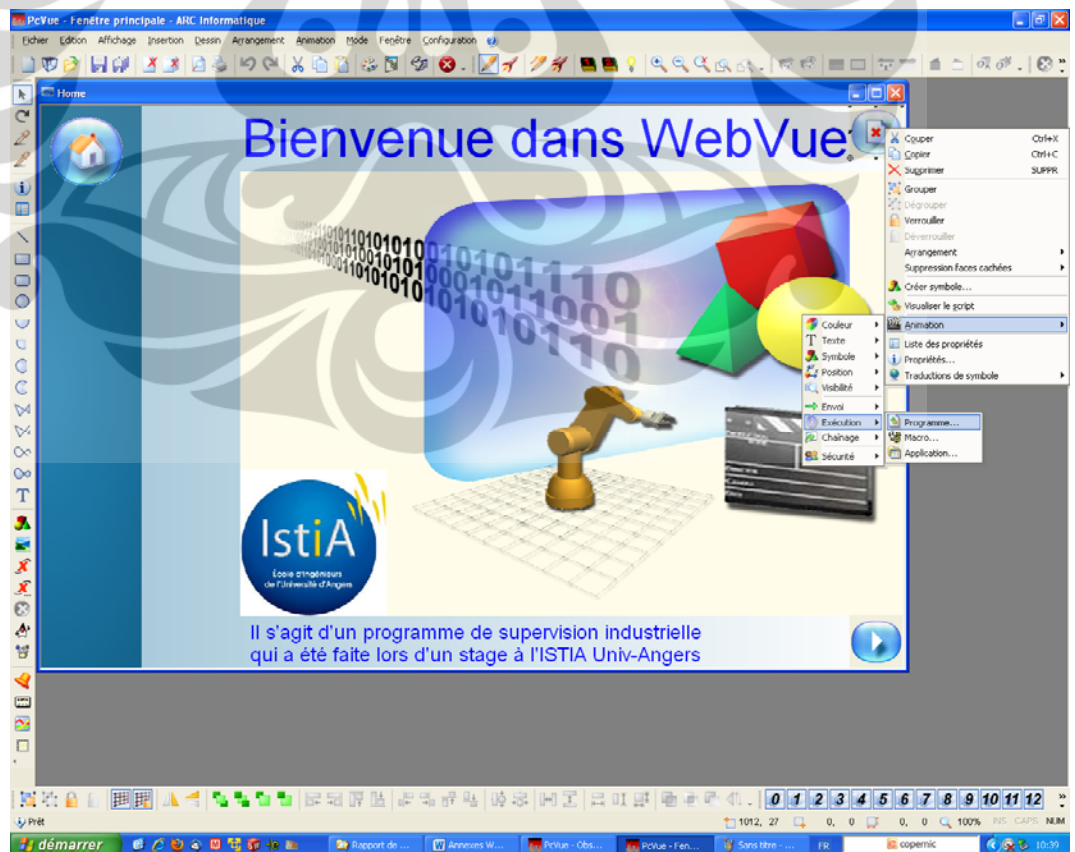


5. Pour lier le deuxième synoptique en premier cliquer droit sur l'icône , puis **>Animation> Chainage > Ouverture**

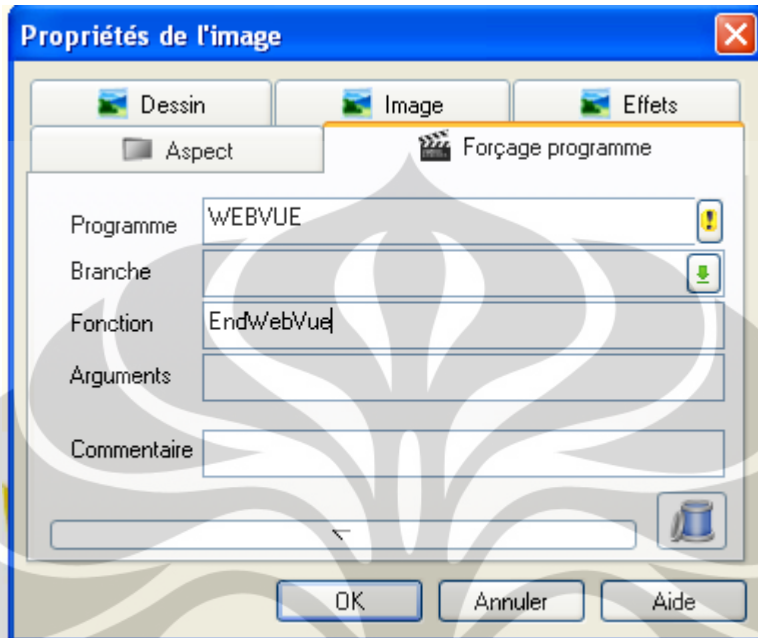




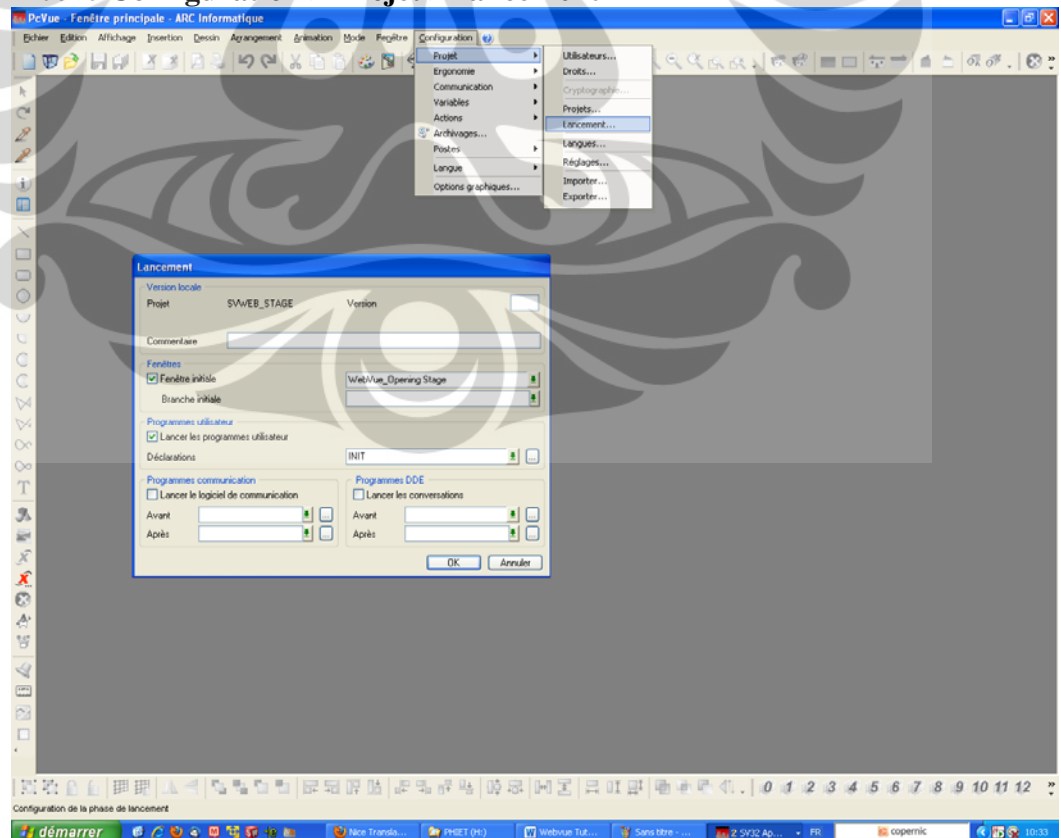
6. Pour provoquer fermeture du synoptique home, cliquer sur le bouton droit sur l'icône bouton , puis **>Animation>Exécution > Programme**



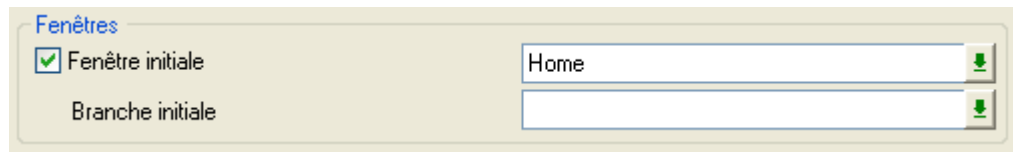
Forçage programme  
Programme : WEBVUE  
Fonction : EndWebVue



7. pour démarrer un affichage synoptique, cliquer sur le bouton de droit sur l'icône **Configuration** > **Projet** > **Lancement**



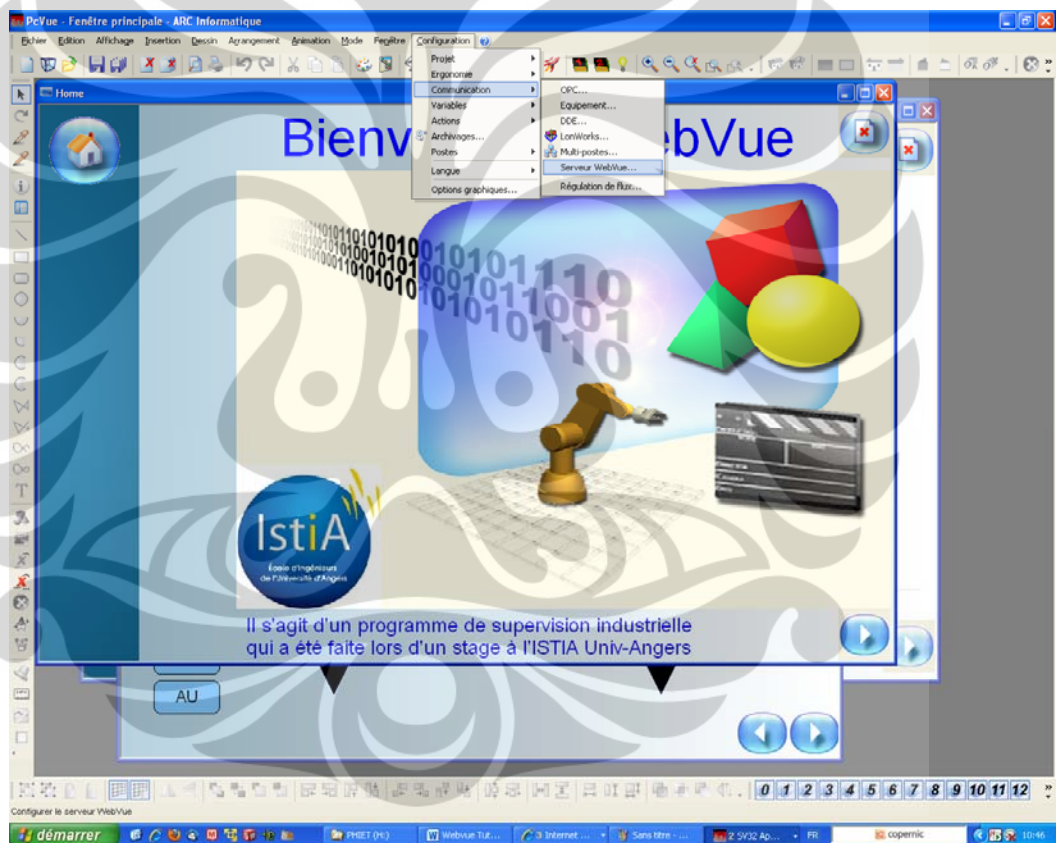
Dans la fenêtre, fenêtre initiale Home



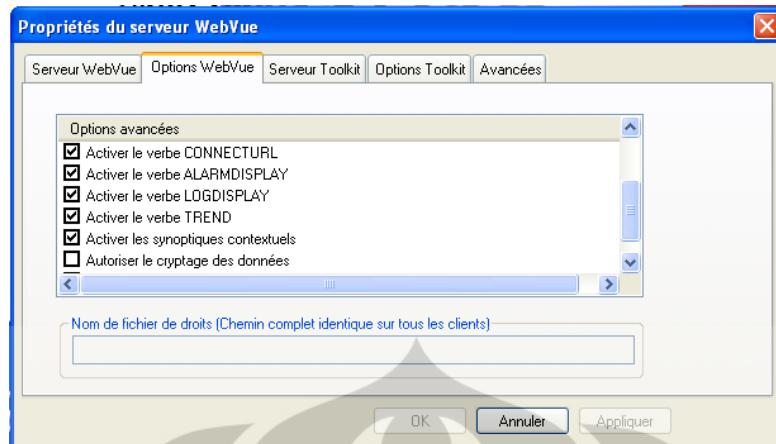
Dans le programmes utilisateur



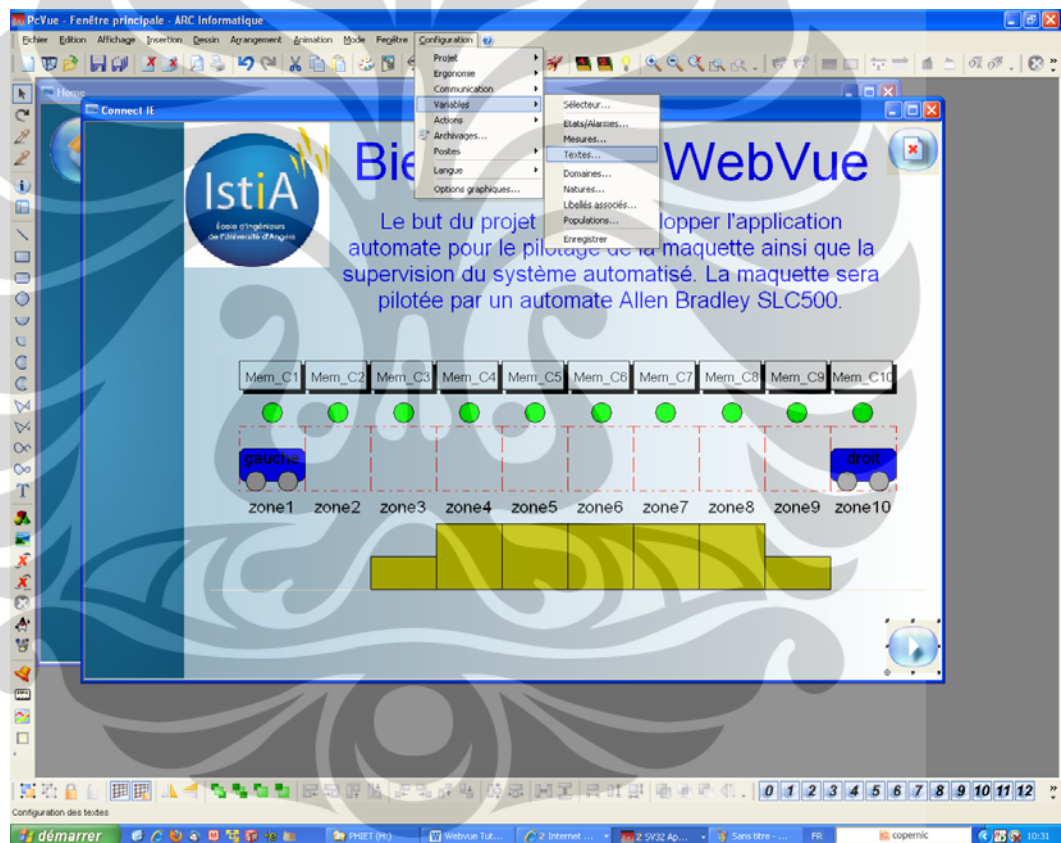
8. Pour la connexion à l'internet, on doit fait :
  - a. Configurer server webvue







b. Déclarer des variables le texte



**Fiche de description texte**

Nom du texte: WEBVUE.IEPATH Chercher

Libellé: Supprimer...

Options: ...

Commande: Niveau de conduite: 0

Domaine: xxxxxxxx

Nature: xxxxxxxx

Format:

Taille: 132

Type de texte: Interne

Commande

Diffusion

<< >> Valider Annuler Initialiser

Valider > Enregistre

c. Déclarer des variables la mesure

The screenshot shows the 'PcVue - Fenêtre principale' window with the 'Configuration' menu open. The 'Mesures' option is selected, and a sub-menu is visible with the following items: Sélecteur..., Etablir/Alarmer..., Mesures..., Textes..., Domaines..., Natures..., Libellés associés..., Populations..., and Enregistrer. The main workspace displays a 'WebVue' diagram with 10 measurement points (Mem\_C1 to Mem\_C10) and 10 zones (zone1 to zone10). The diagram includes a 'gauche' button on the left and a 'droit' button on the right, with a dashed red box highlighting the measurement points and zones. The background text reads: 'Le but du projet est de développer l'application automate pour le pilotage de la maquette ainsi que la supervision du système automatisé. La maquette sera pilotée par un automate Allen Bradley SLC500.'

**Fiche de description mesure**

Nom de la mesure: WEBVUE.NUMUSERS | Chercher

Libellé: | Supprimer...

Options: ...

Domaine: xxxxxxxx

Nature: xxxxxxxx

Format: |

Unité: |

Bande morte: 0 | Type: 0

Min: 0 | Max: 10

**Système de seuil**

phaut / haut / bas / pbas

Type	Valeur	Hystérésis	Etat
<input type="checkbox"/> phaut	0	0	...
<input type="checkbox"/> haut	0	0	...
<input type="checkbox"/> bas	0	0	...
<input type="checkbox"/> pbas	0	0	...

**Type de mesure**

Interne

Consigne

Chronomètre | ...  Compteur | ...

**Interne**

Diffusion

**Consigne**

Niveau de conduite: 0

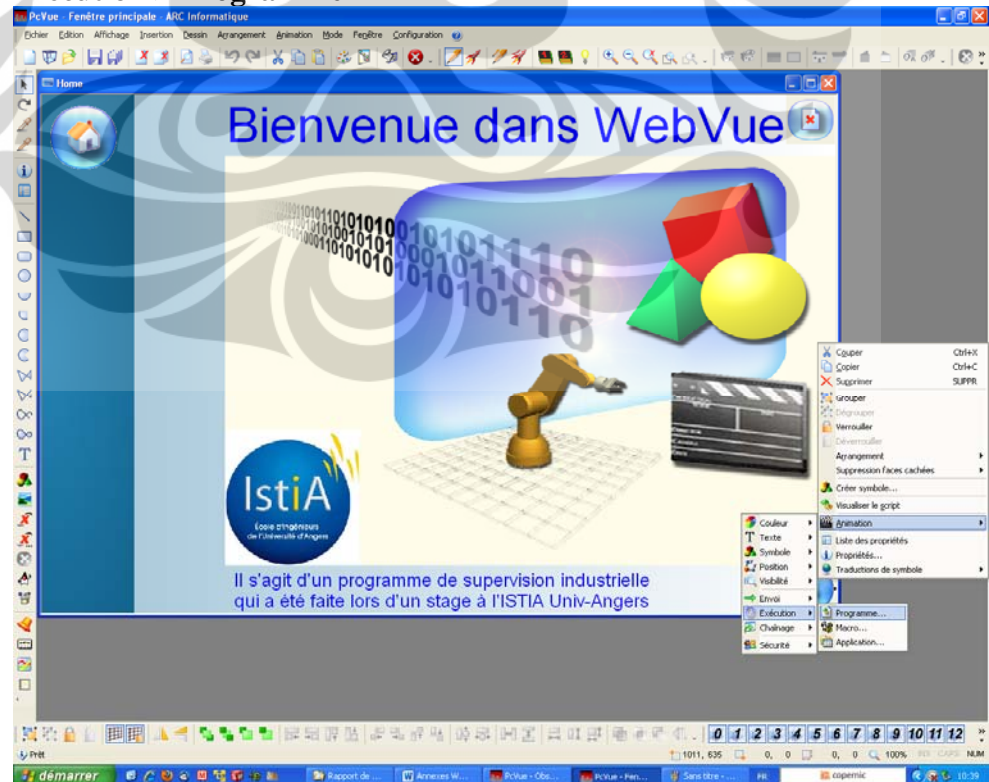
Minimum: 0


Maximum: 10

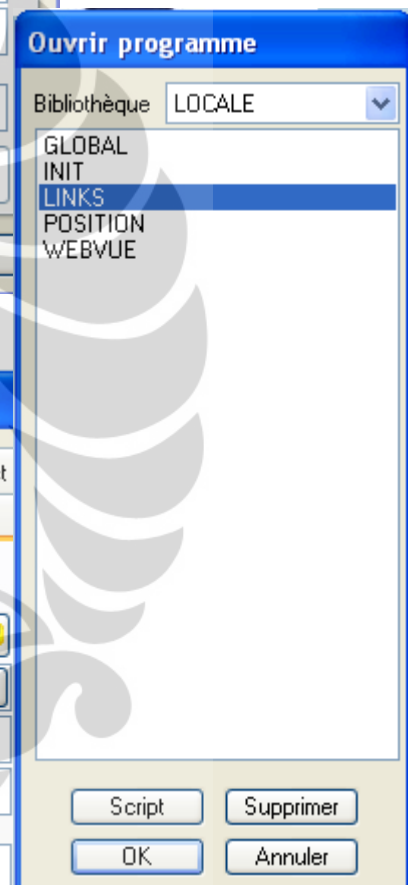
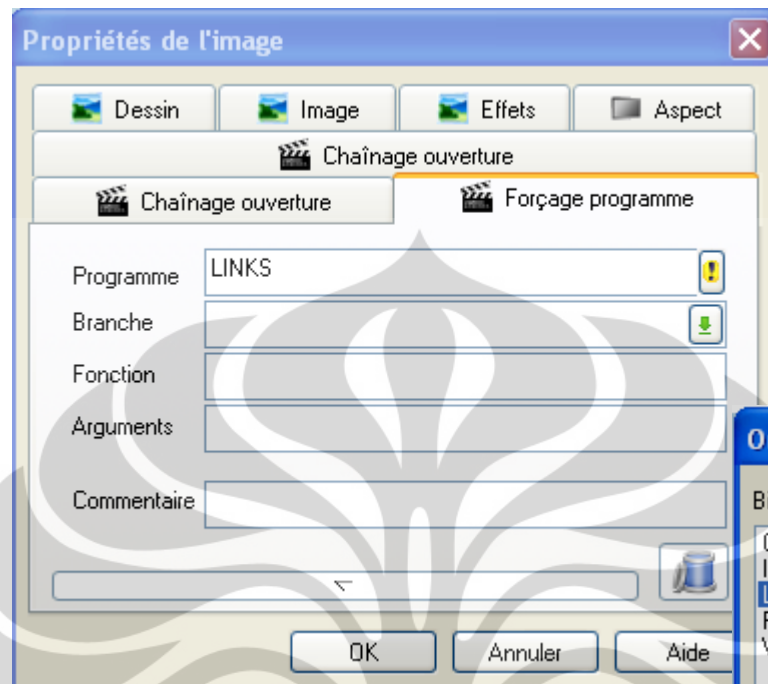
<< >> Valider Annuler Initialiser

Valider > Enregister

- d. Réglage de la programmation  
 Cliquer sur le bouton de droit sur l'icône  > **Animation** > **Execution** > **Programme**



Cliquant sur l'icône 



## Programmation en WebVue

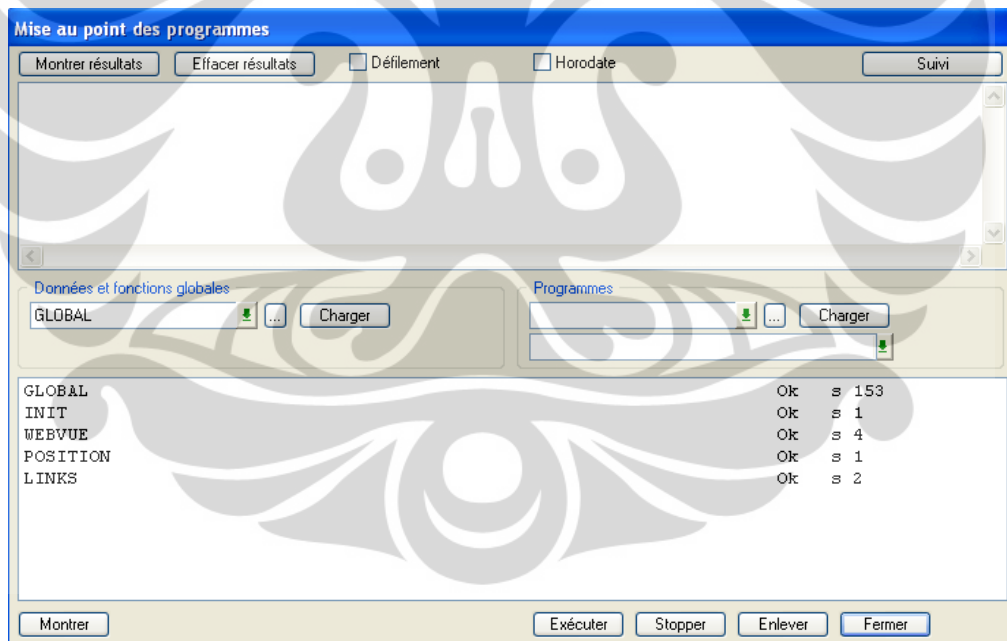
Des traitements numériques et/ou logiques peuvent nécessiter la création de programmes inclus dans la supervision. Avec PCVUE les programmes peuvent être écrits en SCADA Basic ou en VBScript. SCADA BASIC est un langage interprété structuré en bloc avec une syntaxe proche du standard industriel BASIC.

### Réalisation d'un programme

Les fichiers de programmes peuvent être créés, débogués et testés en ligne en utilisant les possibilités de la boîte de dialogue Mise au point des Programmes. Cette dernière est affichée

- par la commande **Configuration>Actions>Programmes**,
- ou bien en cliquant sur l'icône .

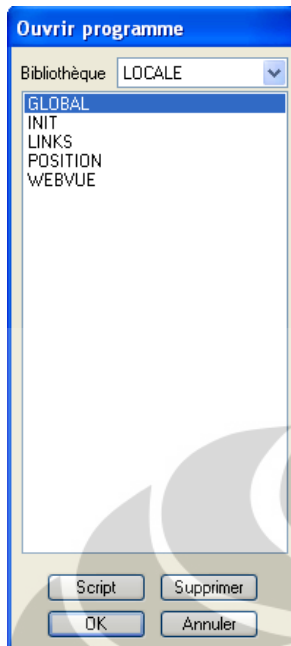
Deux zones sont similaires en apparence et en fonctionnement : l'une, appelée '**Données et fonctions globales**', traite de la déclaration du programme global (programme obligatoires d'initialisation et de déclaration) et l'autre, appelée '**Programmes**', de tous les autres programmes (programmes écrits pour des tâches particulières).



Ouvre la fenêtre d'édition des programmes. Pour un nouveau programme, il faut avoir précédemment saisi son nom dans la zone de texte à gauche.



Affiche la boîte de dialogue suivante pour pouvoir choisir un programme déjà existant



On peut choisir directement la bibliothèque que l'on souhaite. Avec la bibliothèque locale, tous les programmes présents sous le répertoire c:\pcvue32\usr\nom\_analyse\p sont affichés.

- Affiche le script du programme choisi
- Supprime le programme choisi
- Sélectionne le programme

Pour un programme sélectionné :

- cliquer sur **charger** provoque sa précompilation (les références aux variables et aux fonctions ne sont toutefois pas vérifiées),
- cliquer sur **exécuter** provoque l'exécution d'un programme chargé,
- cliquer sur **Montrer les résultats** permet de visualiser dans la zone supérieure les messages des programmes et les erreurs de syntaxe.

### ***Le programme GLOBAL***

Le programme GLOBAL (chargé à partir de la zone '**Données et fonctions globales**') est utilisé pour uniquement déclarer les variables de travail et les fonctions qui sont utilisées dans tout le projet. Il doit exister, même s'il est vide, sinon les autres programmes du projet ne peuvent pas fonctionner. Le programme GLOBAL peut avoir n'importe quel nom et son fichier source est sauvegardé dans le même répertoire que les autres programmes (répertoire nommé "P").

```

\----- Program GLOBAL
'commentaire
\----- End

```

### ***Le programme INIT***

Le programme INIT est un program SCADA Basic qui est utilisé pour uniquement initialise la première condition de travail quand démarrer ce projet. Cette condition est une condition où la position des chariots, les capteurs et le nombre de cycles dans la position ou la valeur par défaut. En outre, il y a trois programs qui est charger et l'exécute quand démarrer ce projet.

```
Sub main()
```

```

@WEBVUE.IEPATH="C:\\Program Files\\Internet
Explorer\\IEXPLORE.EXE";
@mesures.Nb_cycle=0;
@mesures.posCG=206;
@mesures.posCD=844;

@etats.C1_B312_1=0;
@etats.C2_B312_2=0;
@etats.C3_B312_3=0;
@etats.C4_B312_4=0;
@etats.C5_B312_5=0;
@etats.C6_B312_6=0;
@etats.C7_B312_7=0;
@etats.C8_B312_8=0;
@etats.C9_B312_9=0;
@etats.C10_B312_10 =0;

PROGRAM("START","WEBVUE","");
PROGRAM("START","POSITION","");
PROGRAM("PRELOAD","LINKS","");
End sub

```

Le programme doit avoir au moins une fonction, appelée MAIN, qui est le point d'entrée de l'exécution. Les variables PC-VUE sont référencées en faisant précéder leur nom d'un '@', par exemple : @etats.C1\_B312\_1=0;

Il y a 5 d'initiales conditions de travail qui doivent être remplies quand démarrage ce projet, à savoir :

- Déclaration de variable texte de WEBVUE.IEPATH. Cette variable d'accéder au navigateur web qui sera utilisée lors de l'exécution le site du projet web superviseur.
- Initialiser la valeur de variable mesure de Nb\_cycle au zéro.
- Initialiser des positions de chariot gauche et chariot droit au celles position par défaut.
- Initialiser les valeurs des capteurs avant de lancement des chariots au zéro.
- Charger et l'exécute des programmes WEBVUE et POSITION, et aussi charger le programme LINKS qui ont utilisé dans ce projet depuis le démarrage.

### ***Le programme LINK***

Le programme LINKS est utilisé pour relier le projet de web superviseur avec le Serveur WebVue. Données stockées sur le Serveur Webvue peut être appelé via localhost en raison d'Internet Information Services (IIS).

```

Sub StartWebVue()
    Dim cIEpath As Str;
    Dim cProject As Str;
    Dim cURL As Str;

```

```

    if (LANGUAGE("GET")==1) then
        cProject=GetArg("ARG1");
    else
        cProject=GetArg("ARG2");
    end if

    cIEpath = @WEBVUE.IEPATH;
    cURL=ADDSTRING("http://localhost/svweb_stage/index.html?project=",cProject);

    Application ("LOAD", cIEpath, cURL);
End sub

Sub StartIISPortal()
    Dim cIEpath As Str;
    cIEpath = @WEBVUE.IEPATH;
    Application ("LOAD", cIEpath, "http://localhost/classes/");
End sub

```

Les variables propres au programme (dite *de travail*) sont déclarées grâce au mot réservé `dim`: `Dim cIEpath As Str`;

Ce programme va générer une adresse URL « `http://localhost/svweb_stage/index.html?project=` » qui est déclaré dans ce programme et puis suivi d'argument du nom du projet qui est écrit sous l'onglet forçage programme. Ensuite, le programme LINK va activer IIS portail pour accès le répertoire virtuel qui a été créé. `classes` est un répertoire virtuel qui est utilisé par le client WebVue pour charger l'applet Java.

### ***Le programme POSITION***

Le programme POSITION est utilisé pour placer la position du chariot sur le changement de l'état. Par exemple, si la valeur du variable de l'état `C2_B312_2` est égale à un, de position du chariot gauche va déplacer à position 275.

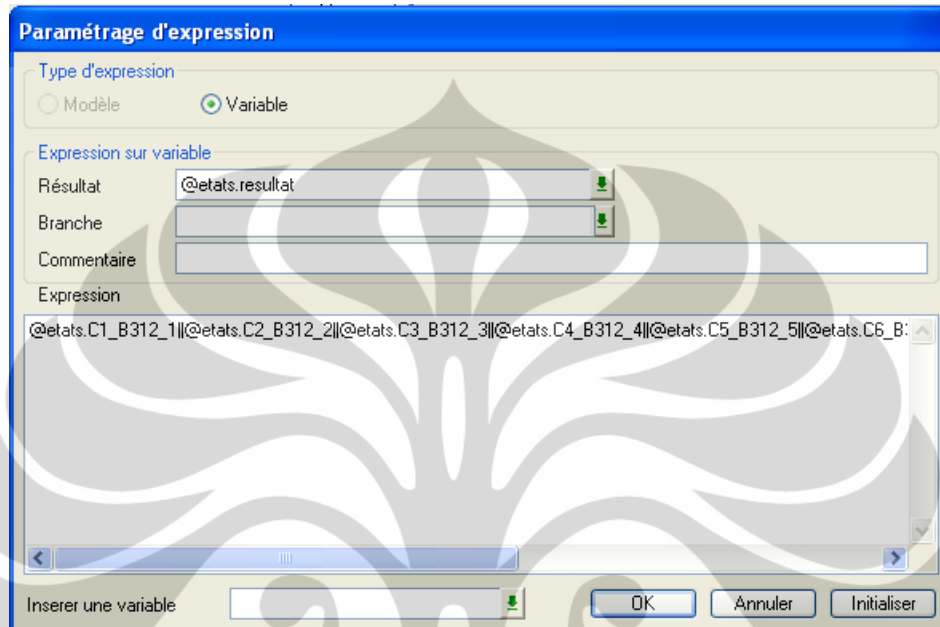
```

Submain()
    if(@etats.C1_B312_1==1)then
        @mesures.posCG=206;
        @etats.C1_B312_1=0;
    end if
    if(@etats.C2_B312_2==1)then
        @mesures.posCG=275;
        @etats.C2_B312_2=0;
    end if
    .
    .
    .
    if(@etats.C10_B312_10==1)then
        @mesures.posCD=844;
        @etats.C10_B312_10=0;
    end if
End sub

```



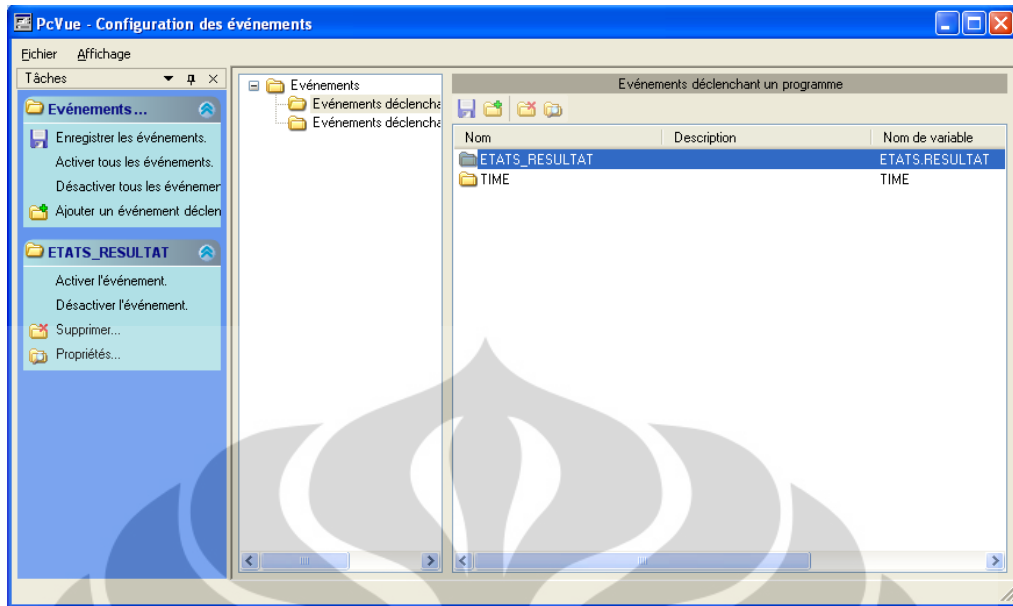
Afin de placer la position des chariots sur le changement de l'état du capteur comme il a été écrit dans le programme POSITION, on doit créer un événement qui sera appelé quand le programme POSITION exécuté. Pour cela on a besoin d'une expression qui permettra automatiquement évaluée à chaque fois que l'une de ses variables change.



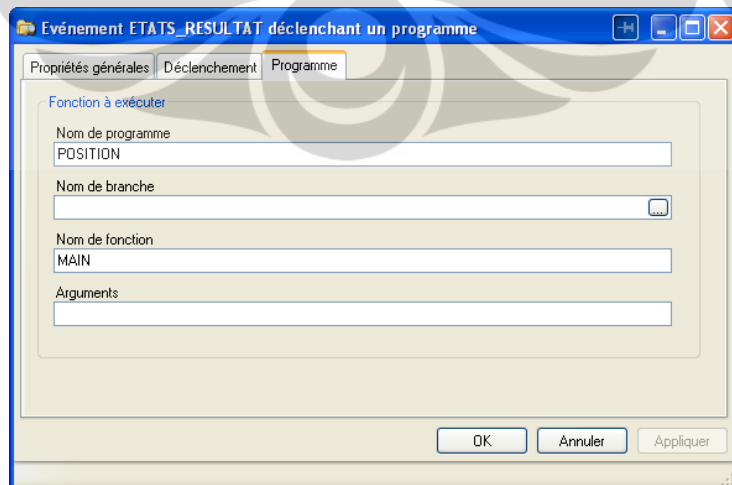
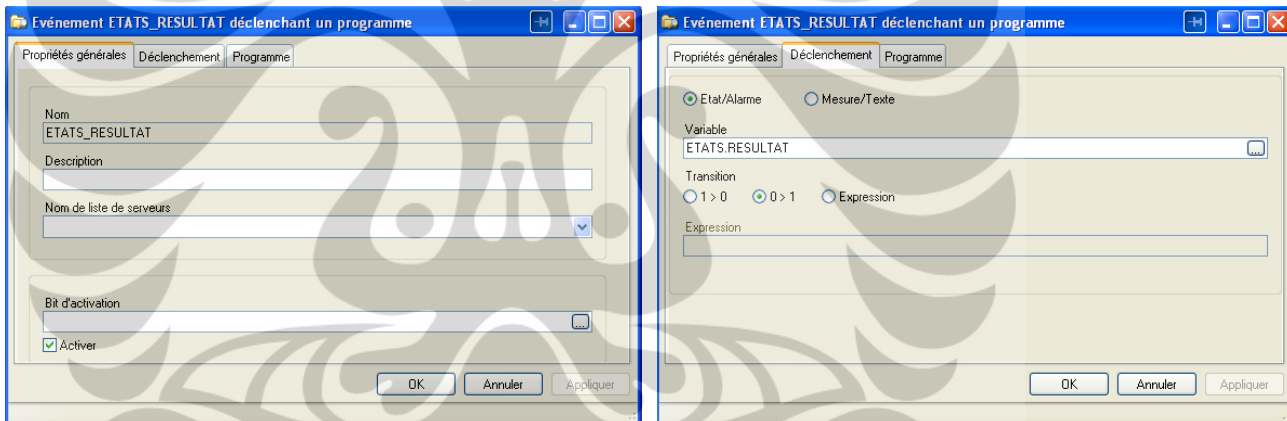
Paramètre :

- Résultat - Affiche le sélecteur de variable et permet de sélectionner une variable dans laquelle le résultat de l'expression est écrit. La variable n'a pas besoin d'être en "commande".
- Branche - Affiche le sélecteur de branche et permet de sélectionner une branche de la base de données. La branche est appliquée à tous les éléments de l'expression.
- Expression - L'expression à définir.
- Insérer une variable - Affiche le Sélecteur des Variables et permet de sélectionner une variable. Le nom de la variable est ajouté dans l'expression à la position courante du curseur.

Pour appeler à chaque occurrence d'un événement, le program est ajoute aux **Actions événementielles** de la supervision. Il suffit alors d'appeler la commande **Configuration>Actions>Événementielles**, puis de préciser le nom du programme ainsi que l'événement d'appel.



La boîte de dialogue Création d'un événement déclenchant un programme, s'affiche.



## ***Le Programme WEBVUE***

Le programme WEBVUE est utilise pendant supervision sur le navigateur web. Il donne d'animation sur le bouton dans l'interface graphique et mesure le nombre d'utilisateur web.

```
Sub Main ()
    @WEBVUE.IEPATH      =      "C:\\Program      Files\\Internet
Explorer\\IEXPLORE.EXE";
End Sub

Sub StartWebVue ()
    Application          ( "LOAD",          @WEBVUE.IEPATH,
"http://localhost/svhmiweb_stage/");
End Sub

Sub EndWebVue ()
    Dim cIEpath As Str, cMimic As Str;
    Dim iUser As Integer;

iUser = GetArg("WEB");
cMimic = GetArg ("WINDOW");

    If (iUser != 0) Then          '----- Opened from IE
cIEpath = @WEBVUE.IEPATH;
        Application ("UNLOAD", cIEpath);
    Else          '----- Opened from the
Supervisor
        Window ("OPEN", "Home", "");
        Window ("CLOSE", cMimic, "");
    End If
End Sub

Sub Stats()

    Dim lHandle as Long;
    Dim cWebdata As Str;
    Dim iResult As Integer;

lHandle = Alloc_Buffer ( 256 );
iResult = WebVue ( "LIST", lHandle);
cWebdata = Cget_Buffer ( lHandle, 0, 255);

    @WEBVUE.NUMUSERS = Tos(iResult );

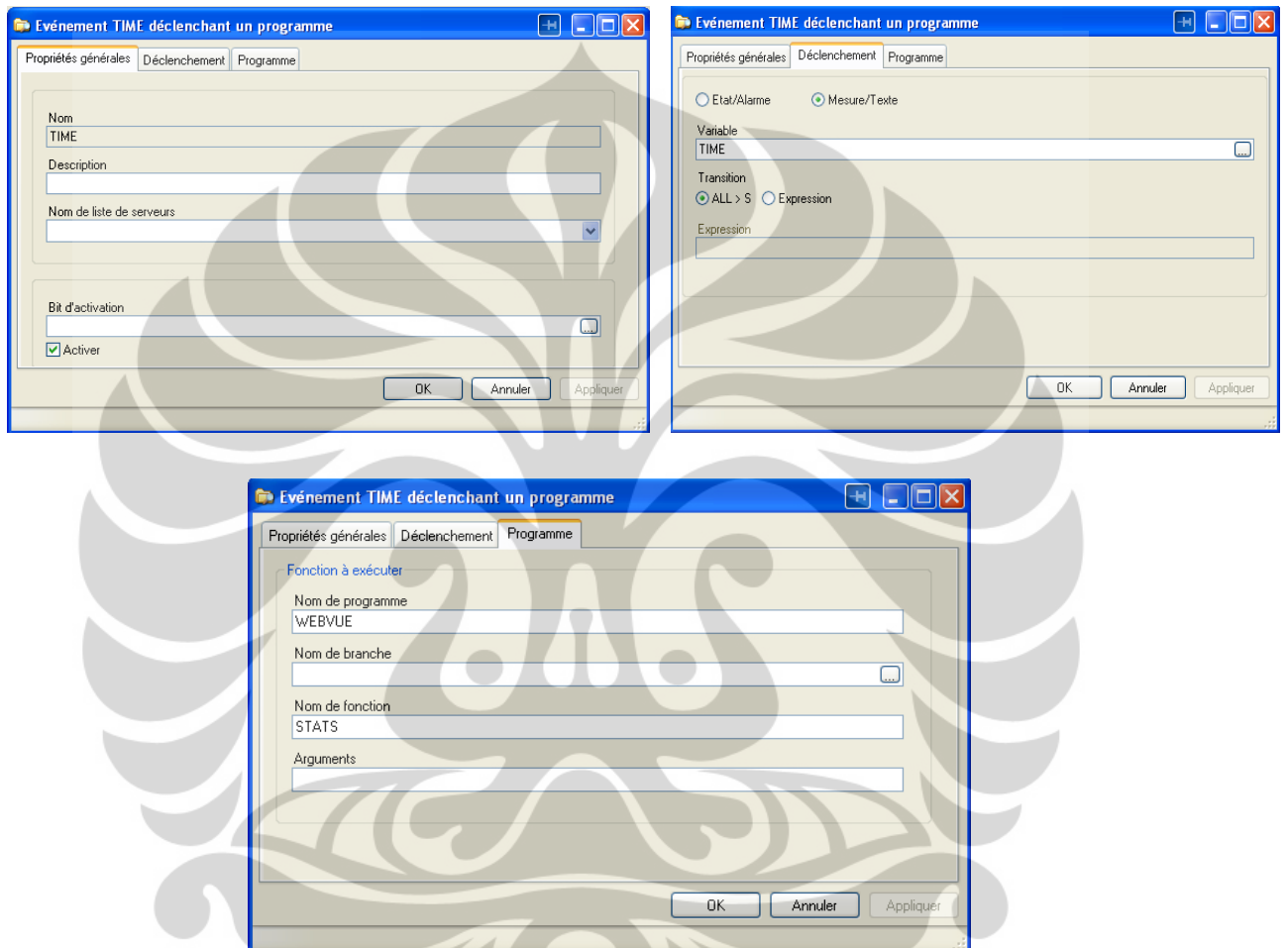
Free_Buffer (lHandle);

End Sub
```

Il y a 4 sub programme dans le programme WEBVUE. Chacun sub programme sera utiliser comme fonction dans l'onglet de forçage programme quand faire d'animation dans le synoptique.

Il existe sub main programme qu'il s'appelle « StartWebVue » et « EndWebVue ». Les deux sub main programme vont utiliser pour commencer et se termine le projet qui a démarré dans le navigateur web.

Le sub main programme « Stats » est nécessaires pour mesure le nombre de client WebVue qui a connectée. Il nécessaire quand déclencher le programme WebVue sur un événementielle du projet.





## **L A M P I R A N**

### **Google Web Toolkit (GWT)**

# Construire une application GWT

## Étape 1: Conception de l'application

1. L'examen des exigences fonctionnelles
2. Identifier les éléments de la conception de l'interface

## Étape 2: créer l'interface utilisateur

1. Sélection des widgets GWT pour mettre en œuvre les éléments d'interface
2. Sélection des panneaux GWT pour disposer les éléments d'interface utilisateur
3. Intégrer l'application dans la page d'accueil
4. La mise en œuvre des widgets et des panneaux
  - Instancier chaque widget et le panneau
  - Créer une table
  - Disposer les widgets
  - Associez le panneau principal avec le panneau de Racine
  - Déplacer le focus curseur vers la zone de saisie
5. Test de la disposition

## Étape 3: Gérer les événements sur le client

1. Revoir les exigences pour la gestion des événements
2. Ecoute des événements
3. Répondant aux événements utilisateur
4. Test de gestion des événements

## Étape 4: Fonctionnalité de code sur le client

1. Ajoutant et en supprimant
2. Rafraîchissant

3. Ajoutant besoins supplémentaires

#### **Step5: Debug a GWT Application**

1. Trouver le bug
2. Correction du bogue
3. Test de la correction d'un bug dans le mode de développement

#### **Etape 6:Appliquer le style**

1. Associer les feuilles de style avec un projet
2. Changer le thème
3. Associer des règles de style avec GWT-éléments HTML généré
4. Création de styles secondaires dépendant d'un style principal
5. Mise à jour dynamiquement les styles
6. Définition des attributs d'un élément HTML
7. Ajout d'images ou d'autres éléments HTML statiques

#### **Etape 8:compilation d'une application GWT**

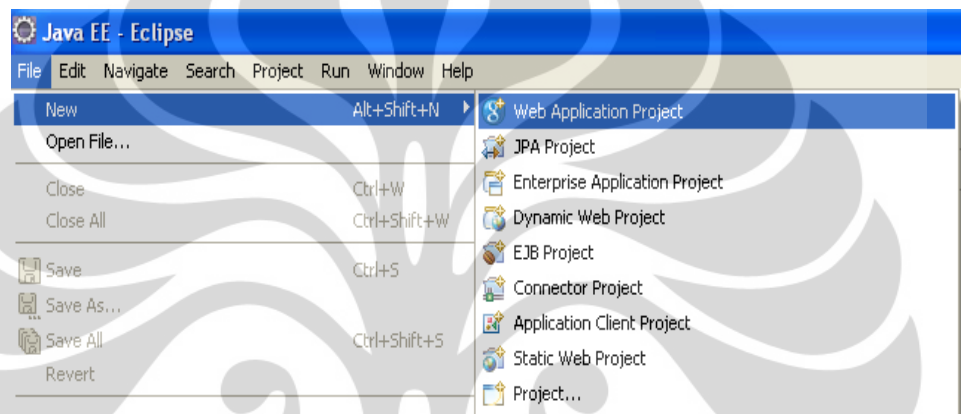
1. La compilation Java à JavaScript
2. Test en mode de production
3. Déploiement de l'application sur un serveur Web

## Créer votre première application GWT

Ce qui suit est une description de l'utilisation du plug-in Google pour créer une nouvelle application GWT.

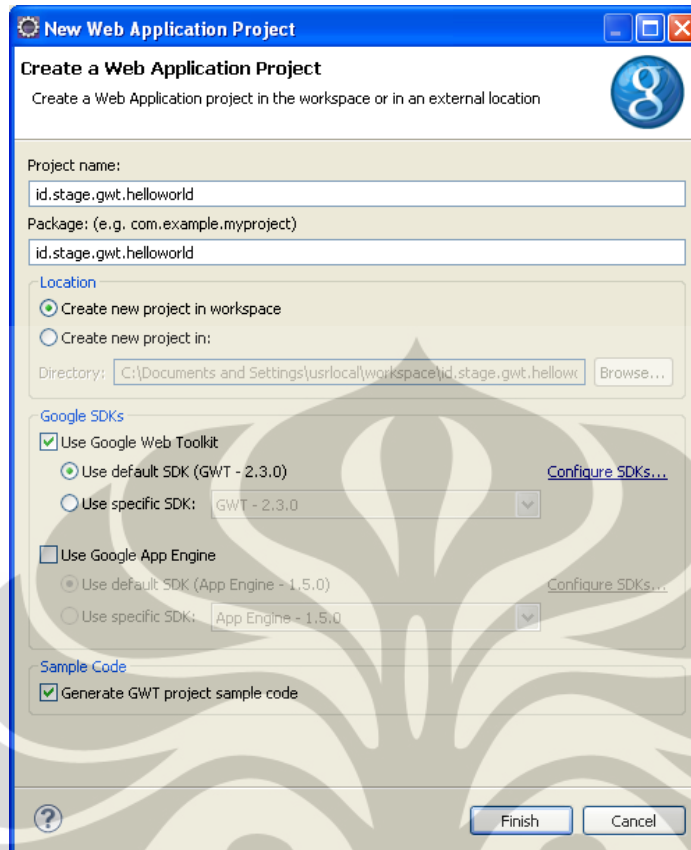
### 1. Projet

Choisissez File > New > Web Application Project.



Entrez le nom de "id.stage.gwt.helloworld" pour votre projet et le package java. Ne sélectionnez "Utiliser Google Web Toolkit", que nous n'utiliserons pas le Google App Engine dans ce tutoriel.

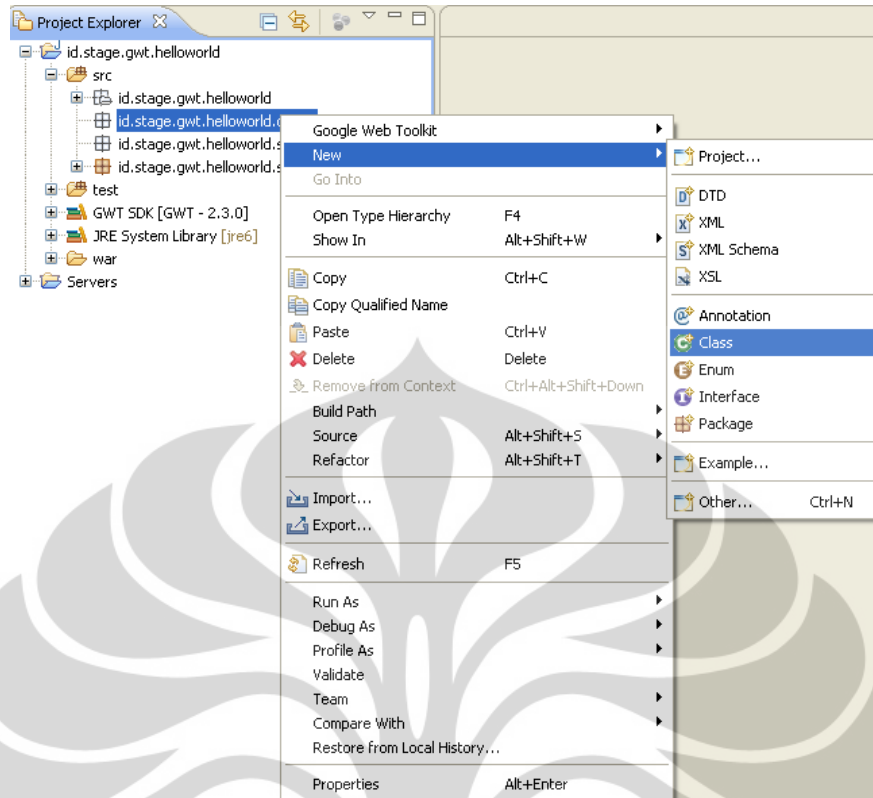




Supprimez tous les fichiers java dans le cadre du paquet «client» et «server».

## 2. GWT Entry Point

Créez une classe Java suivante "HelloGwt" dans le paquet "id.stage.gwt.helloworld.client"



```

package id.stage.gwt.helloworld.client;

import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.event.dom.client.ClickEvent;
import com.google.gwt.event.dom.client.ClickHandler;
import com.google.gwt.user.client.Window;
import com.google.gwt.user.client.ui.Button;
import com.google.gwt.user.client.ui.Label;
import com.google.gwt.user.client.ui.RootPanel;

public class HelloGwt implements EntryPoint {

    @Override
    public void onModuleLoad() {
        Label label = new Label("Hello GWT !!!");
        Button button = new Button("Say something");
        button.addClickHandler(new ClickHandler() {
            @Override
            public void onClick(ClickEvent event) {
                Window.alert("Hello, again");
            }
        });

        RootPanel.get().add(label);
        RootPanel.get().add(button);
    }
}

```

Créez le fichier "Id\_stage\_gwt\_helloworld.gwt.xml" dans le paquetage "id.stage.gwt.helloworld" comme suivi.

```
<?xml version="1.0" encoding="UTF-8"?>
<module rename-to='id_stage_gwt_helloworld'>
<!-- Inherit the core Web Toolkit stuff. --
>
<inherits name='com.google.gwt.user.User' />

<inherits name='com.google.gwt.user.theme.standard.Standard' />

<!-- Specify the app entry point class. --
>
<entry-point class='id.stage.gwt.helloworld.client.HelloGwt' />
</module>
```

Ce fichier définit votre point d'entrée à votre application. Dans GWT le point d'entrée est similaire à la méthode principale de Java.

### 3. HTML page

Dans la guerre dossier / WEB-INF, vous devriez trouver un dossier avec le fichier "Id\_stage\_gwt\_helloworld.html". Modifier ce fichier comme suivi..

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<link type="text/css" rel="stylesheet"
href="Id_stage_gwt_helloworld.css">
<title>My First GWT applicaton</title>
<script type="text/javascript" language="javascript"
src="id_stage_gwt_helloworld/id_stage_gwt_helloworld.nocache.js"><
</script>
</head>

<body>

<!-- OPTIONAL: include this if you want history support -->
<iframe src="javascript:''" id="__gwt_historyFrame" tabIndex='-1'
style="position:absolute;width:0;height:0;border:0"></iframe>
<h1>My First GWT applicaton</h1>

</body>
</html>
```

#### 4. web.xml

Le plug-in Google a également créé un fichier web.xml et placé une référence à une servlet (que nous avons supprimé). Modifier le fichier web.xml comme suivi.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>

    <!-- Default page to serve -->
    <welcome-file-list>
        <welcome-
file>Id_stage_gwt_helloworld.html</welcome-file>
    </welcome-file-list>
</web-app>
```

#### 5. Modifier le CSS

Changer le CSS de l'application. Dans le dossier vous trouverez un fichier "Id\_stage\_gwt\_helloworld.css". Remplacez-le par le texte suivant.

```
h1 {
font-size: 2em;
font-weight: bold;
color: #777777;
}

.gwt-Label {
    color: #DF0101;
    font: normal 12px tahoma, arial, helvetica, sans-serif;
    border: 1px solid #99bbe8;
    padding: 14px;
}

.gwt-Button {
    height: 5.7em;
    margin-bottom: 5px;
    padding-bottom: 3px;
    font-size: 12px;
    font-family: arial, sans-serif;
}
```

#### 6. Run

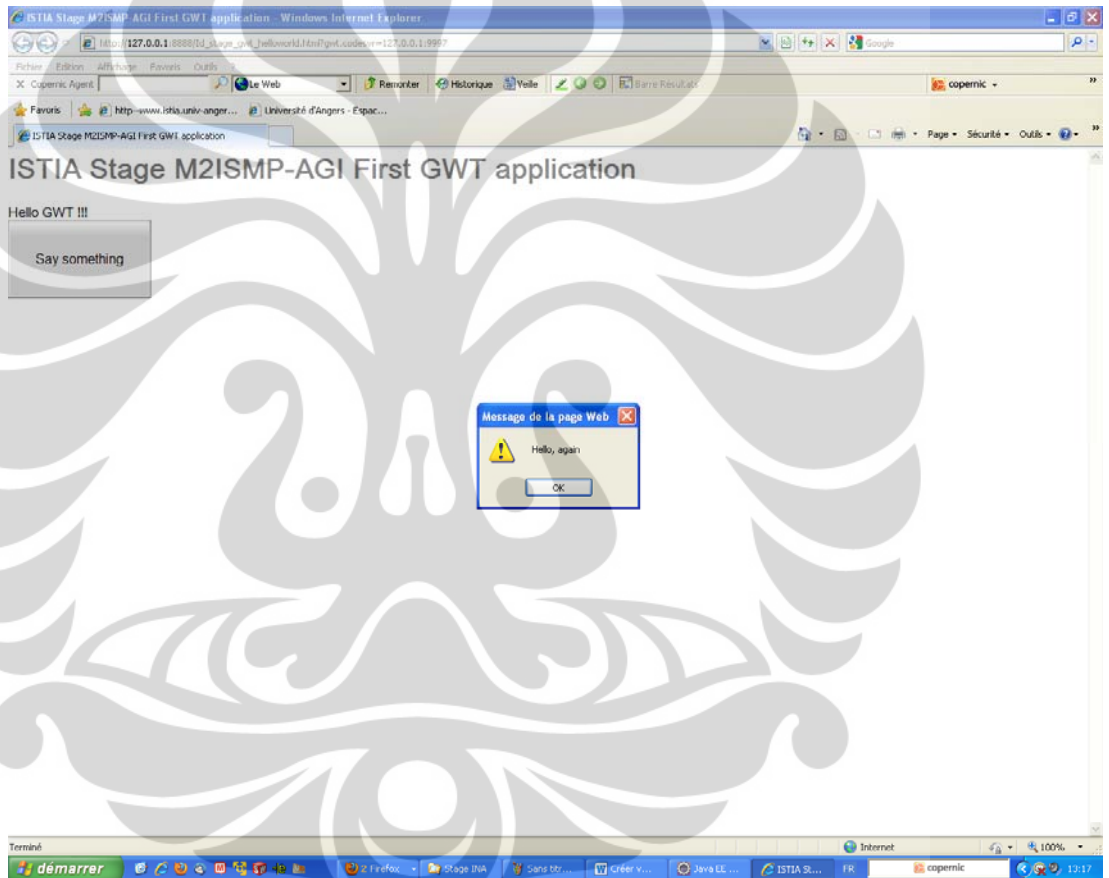
Pour exécuter l'application sur, cliquez droit sur le projet et sélectionnez Run as-> "Web application".

Cela ouvre une nouvelle vue "Development Mode". Copiez l'URL de ce point de vue.



Coller cette URL dans votre navigateur

Le résultat devrait ressembler à ceci:



## REFERENCES

### Sites Web visités :

#### **Pour WebVue**

[http://www.ac-grenoble.fr/ecole.entreprise/CRGE/cteressources/docs2007/pcvue/FORMATION\\_PcVue\\_WebVue.pdf](http://www.ac-grenoble.fr/ecole.entreprise/CRGE/cteressources/docs2007/pcvue/FORMATION_PcVue_WebVue.pdf)

#### **Pour Google Web Toolkit (GWT)**

<http://code.google.com/intl/fr/webtoolkit/doc/latest/tutorial/index.html>

<http://www.vogella.de/articles/GWT/article.html>

<http://moritan.developpez.com/tutoriels/java/gwt/premier/projet/>

<http://jl2tho.blogspot.com/2007/09/tutorial-gwt-g0.html>

<http://www.objjs.com/formation-java/tutoriel-formation-gwt.html>

<http://altair.cs.oswego.edu/~tenberge/tenbergen.org/misc/DB-Access-in-GWT-The-Missing-Tutorial.pdf>

#### **Livre :**

S. Lahaye, **Prise en Main du logiciel PCVUE32**, Module de supervision industrielle,  
ISTIA – Département Automatisation