# 

# REALISATION

## Mise en place de l’environnement de développement



### **Installation et configuration des outils**

* **WinDev**

**WinDev** est un  AGL édité par la société française [PC SOFT](https://fr.wikipedia.org/wiki/PC_SOFT) et conçu pour développer des applications, principalement orientées données pour Windows 10, 8, 7, Vista, XP, 2008, 2003, 2000, et également pour Linux, .NET et Java. Il propose son propre langage : le [WLangage](https://fr.wikipedia.org/wiki/WLangage). La première version de l'AGL est sortie en 1993.

**Configuration requis**

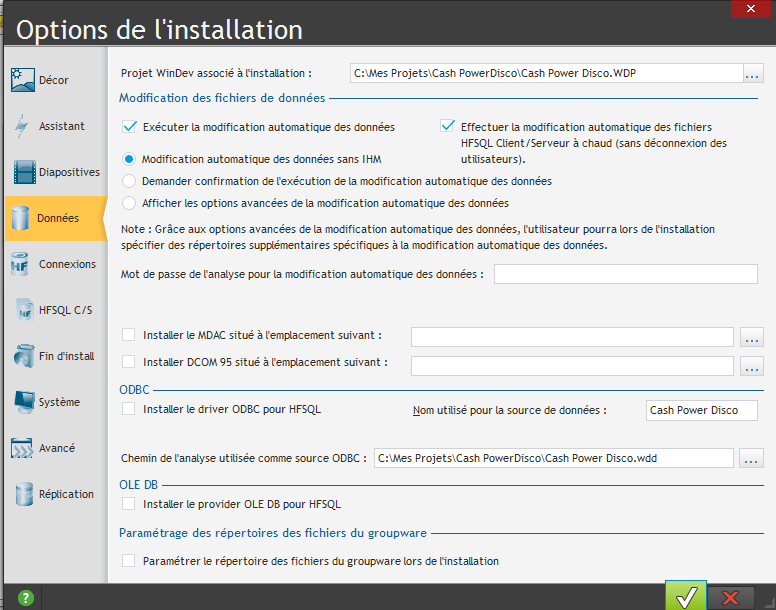
Le tableau x les configurations requis pour l’installation de WinDev

|  |  |  |
| --- | --- | --- |
|  | **Minimum** | **Recommandé** |
| RAM | 4 Go | 8 Go ou plus |
| Espace disque | 4 Go | 10 Go pour une installation complète |
| Système d’exploitation | Windows Vista/7 ou version plus récente (32 ou 64 bits) | |

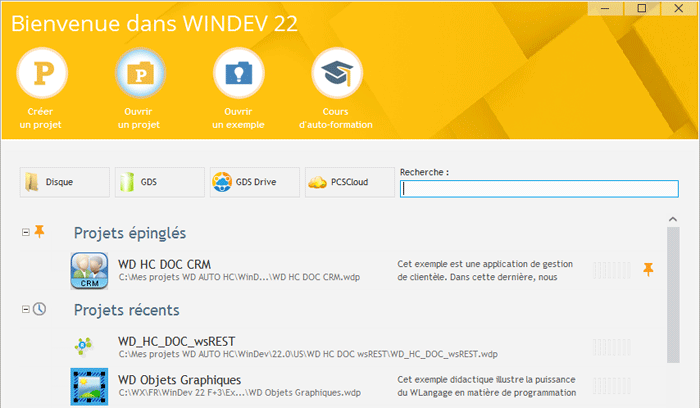
**Installation**

L'installation est simple, il suffit de suivre les instructions présentes sur le CD ROM fourni.

La figure x montre l’étape de l’installation de WinDev



La figure x présente la fenêtre principale de WINDEV



Il est à noter qu’une fois l’installation terminée, une clé électronique doit être brancher sur la poste de développement pour permettre d’utiliser le produit.

* **Visual studio code**

Visual Studio Code est présenté lors de la conférence des développeurs Build d'avril 2015 comme un éditeur de code [cross-platform](https://fr.wikipedia.org/wiki/Cross-platform), [open source](https://fr.wikipedia.org/wiki/Open_source) et gratuit, supportant une dizaine de [langages](https://fr.wikipedia.org/wiki/Langage_informatique).

Le [code source](https://fr.wikipedia.org/wiki/Code_source) est fourni sous la [licence libre](https://fr.wikipedia.org/wiki/Licence_de_logiciel_libre) MIT (plus précisément la [licence Expat](https://fr.wikipedia.org/wiki/Licence_Expat)) sur le site du projet sur [Github](https://fr.wikipedia.org/wiki/Github). En revanche, l'exécutable est proposé sur le site officiel de Microsoft sous une [licence privatrice](https://fr.wikipedia.org/wiki/Logiciel_privateur).

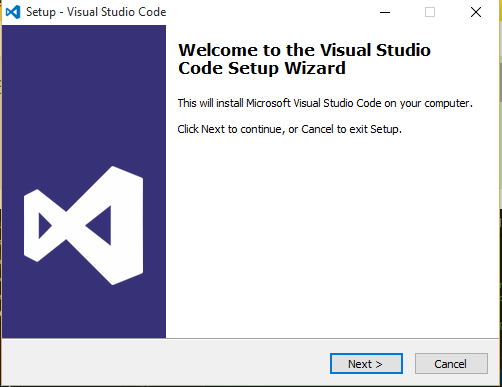
**Configuration requis :**

* 1.6 GHz or faster processor
* 1 GB of RAM
* OS X Yosemite
* Windows 7 (with .NET Framework 4.5.2), 8.0, 8.1 and 10 (32-bit and 64-bit)
* Linux (Debian): Ubuntu Desktop 14.04, Debian 7
* Linux (Red Hat): Red Hat Enterprise Linux 7, CentOS 7, Fedora 23

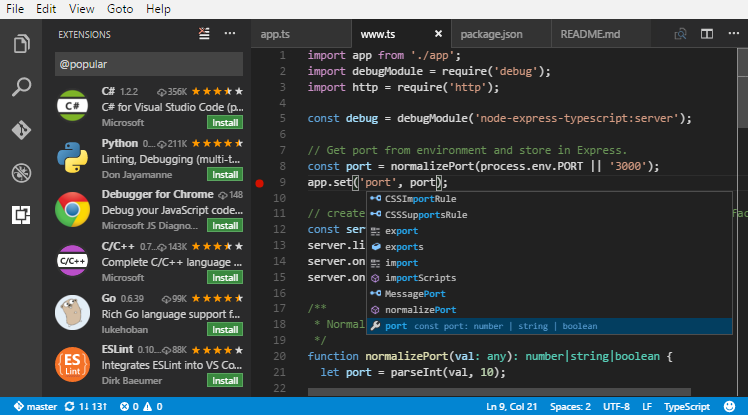
**Installation**

L’installation de Visual Studio Code nécessite avant tout que .NET Framework 4.5.2 soit déjà installé. Puis il suffit de télécharger l’exécutable et de l’exécuter suivant la plateforme choisie depuis le lien suivant : <https://code.visualstudio.com/Download>

La figure x présente le déroulement de l’installation de Visual Studio Code

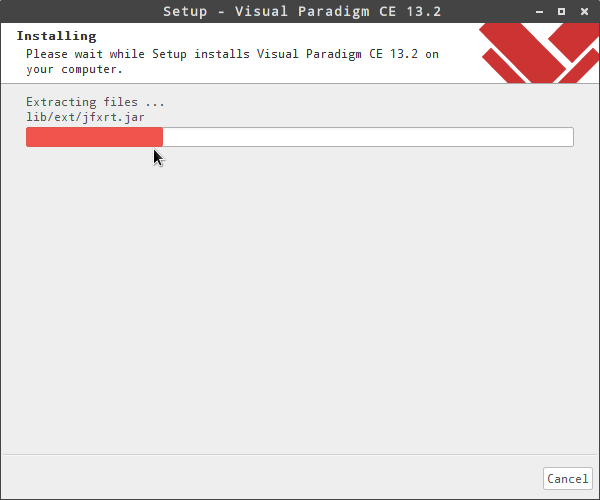
****

La figure x présente la fenêtre principale de Visual studio Code

****

* **VISUAL PARADIGM FOR UML :**

Visual paradigm étant un logiciel payant mais offrant une version communautaire gratuite, dans le cadre de notre projet nous avons choisi la version communautaire. Au niveau de la configuration, celle proposée par défaut nous a amplement convenus.

La figure 53 présente le déroulement de l’installation de Visual Paradigm

* **ORACLE 11G :**

### **Architecture de l’application**

## DEVELOPPEMENT DE L’APPLICATION



### **Création de la base de données**

Le Groupe STAR a depuis x ans opté pour le SGBD Oracle Database pour le stockage de ses données notamment pour son ERP Renaissance au niveau du siège mais aussi dans chaque site à cause de …. Et ….

Les tables suivantes seront alimentées à une intervalle de temps par l’application chargé puis chaque donnée sera utilisé par l’application web dont :

* La table **“AUDIT\_JOURNALIER” :**

CREATE TABLE "FINANCE"."AUDIT\_JOURNALIER"

(  "CACHE\_HIT\_RATIO" VARCHAR2(20 BYTE),

    "DATE\_AUDIT\_JOURNALIER" DATE NOT NULL ENABLE,

    "DIVISION" VARCHAR2(2 BYTE) NOT NULL ENABLE,

    "DIVISION\_NAME" VARCHAR2(64 BYTE),

    "DIVISION\_TYPE" VARCHAR2(6 BYTE),

CONSTRAINT "AUDIT\_JOURNALIER\_PK" PRIMARY KEY ("DIVISION", "DATE\_AUDIT\_JOURNALIER")

);

* La table **“AUDIT\_JOURNALIER\_OBJET\_DETAIL”**

CREATE TABLE "FINANCE"."AUDIT\_JOURNALIER\_OBJET\_DETAIL"

(   "ID\_OBJET\_DETAIL" NUMBER NOT NULL ENABLE,

    "TYPE\_OBJET" VARCHAR2(20 BYTE) NOT NULL ENABLE,

    "NOM\_OBJET" VARCHAR2(30 BYTE) NOT NULL ENABLE,

    "TAILLE\_OBJET" NUMBER,

    "NB\_EXTENT" NUMBER,

    "TAILLE\_UTILISE" NUMBER,

    "DATE\_AUDIT\_JOURNALIER" DATE NOT NULL ENABLE,

    "DIVISION" VARCHAR2(2 BYTE) NOT NULL ENABLE,

    CONSTRAINT "AUDIT\_JOURNALIER\_OBJET\_DET\_PK1" PRIMARY KEY ("ID\_OBJET\_DETAIL"),

CONSTRAINT "AUDIT\_JOURNALIER\_OBJET\_DE\_FK1" FOREIGN KEY ("DIVISION","DATE\_AUDIT\_JOURNALIER")

    REFERENCES "FINANCE"."AUDIT\_JOURNALIER" ("DIVISION", "DATE\_AUDIT\_JOURNALIER") ON DELETE CASCADE ENABLE

);

* La table **“AUDIT\_JOURNALIER\_DISQUE\_DETAIL”**

CREATE TABLE "FINANCE"."AUDIT\_JOURNALIER\_DISQUE\_DETAIL"

(   "ID\_DISQUE\_DETAIL" NUMBER NOT NULL ENABLE,

    "NOM\_DISQUE" VARCHAR2(10 BYTE),

    "ESPACE\_TOTAL" NUMBER,

    "ESPACE\_LIBRE" NUMBER,

    "SERVEUR" VARCHAR2(3 BYTE),

    "DATE\_AUDIT\_JOURNALIER" DATE NOT NULL ENABLE,

    "DIVISION" VARCHAR2(2 BYTE) NOT NULL ENABLE,

    CONSTRAINT "AUDIT\_JOURNALIER\_DISQUE\_DE\_PK" PRIMARY KEY ("ID\_DISQUE\_DETAIL"),

CONSTRAINT "AUDIT\_JOURNALIER\_DISQUE\_D\_UK2" UNIQUE ("NOM\_DISQUE", "DATE\_AUDIT\_JOURNALIER", "DIVISION")

);

### **Codage de l’application**

La figure x affiche un fragment de code du fichier de l’application chargé de collecter les données.



La figure x affiche un fragment de code du fichier de l’application chargé de consommer les données à partir de l’API REST.

Voici la requête SQL chargé de récupérer les informations :

------

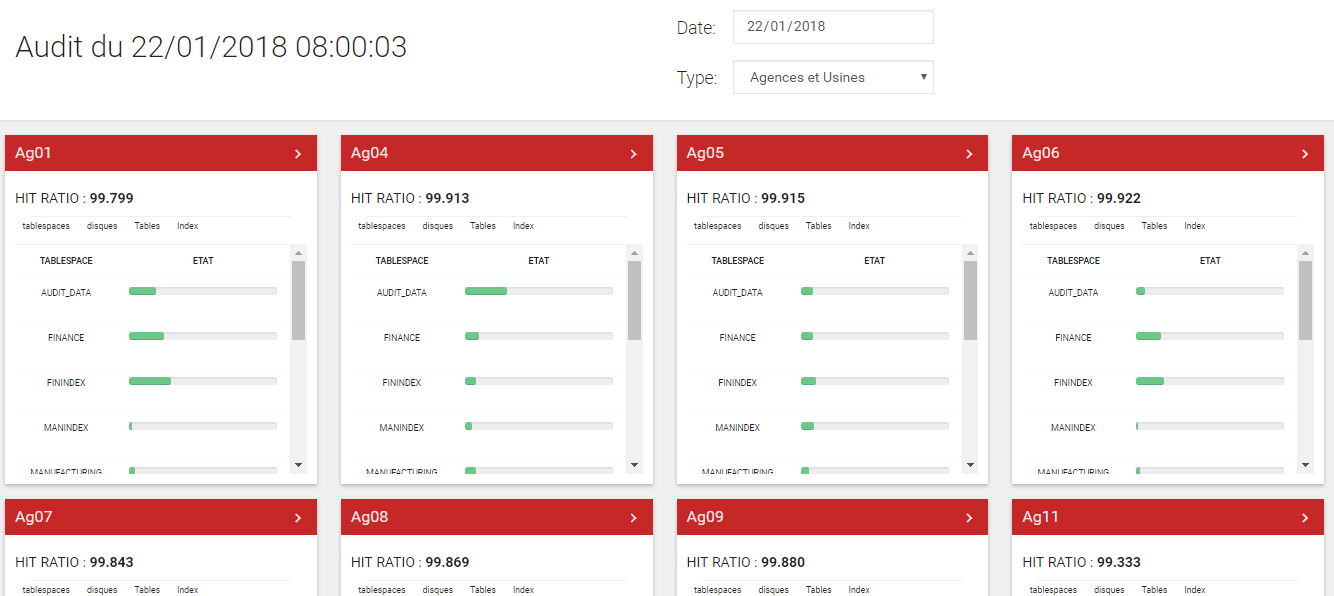
### **Présentation de l’application**

L’application d’alimentation ne présente pas d’interface utilisateur mais sera juste exécuté en tant que tâche planifié pour la collecte d’informations auprès des sites puis de remplir les tables présentés ci-dessus de divers donnée.

L’on présentera donc seulement l’application qui présentera les données à partir de l’interface web.

Voici quelques captures d’écran correspondants:

La figure x affiche la page principale



La figure x affiche la page d’authentification de l’application.

