

# Documentation de l'application intranet : 'InfoSecu'



## Projet professionnel – 'InfoSecu'
















Développement d'une application web intranet basée sur le Framework .NET ainsi que sur des tables de données Access. Celle-ci est codée en javascript pour obtenir une application mono-page (librairie jQuery) et utilise un script PowerShell afin d'assurer une mise en forme des données pour impression.

**Mai-Juin 2017 / Janvier-Février 2018**






Crété Jonathan – Philips Lightning

**Cahier des charges**

## Besoins essentiels :

-  Concevoir une application contraignant les entrants sur le site à prendre part des consignes de sécurité.
-  Créer une application simple et intuitive (gros boutons, image simplifiant une phrase)
-  L'application doit être dynamique (monopage, js, jQuery).
-  L'application devra être disponible en plusieurs langues.
-  Afficher les instructions chauffeur de manière succinctes et claires.
-  Respecter la norme W3C.
-  Créer une vue pour le gardien des entrants enregistrés.
-  Le Gardien peut modifier les entrants enregistrés.
-  Le Gardien peut ajouter manuellement un entrant.
-  Le Gardien peut imprimer les informations d'un entrant sous forme d'un formulaire Word pré-rempli.
-  Une base SQLserver ou des tables Access peuvent être utilisées.
-  Un lien ODBC avec le registre Windows doit être créé ( application interne)
-  Design : Utilisation du Framework Bootstraps
-  Le langage côté serveur est VB.NET
-  Vérifier dynamiquement la cohérence des champs de formulaire (Reg)

## Besoins secondaires (évolutifs) :

-  Le gardien peut rechercher un entrants/sortant dans la base et l'afficher sur sa vue (Développé).
-  Créer un système d'authentification pour les entrants revenant sur le site sous un délai inférieur à 14 jours de manière à ce qu'ils ne relisent pas les consignes de sécurité.(A faire)
-  Des données statistiques peuvent être créées.(A faire)
-  Les données statistiques peuvent être extraites vers un document Excel.(A faire)
-  Créer une vue montrant au gardien les visiteurs du jour.(A faire)

# Description de l'application

L'application 'InfoSecu' permet de vérifier que les chauffeurs ou visiteurs entrants sur l'entrepôt de VSG ont bien prit connaissance des consignes de sécurité.

L'application intranet se découpe en 2 parties :

## I. Le Front-Office :

The screenshots illustrate the user flow for the 'InfoSecu' application:

- 1. Registration Form:** A form titled 'Information(s) Transporteur' with fields for 'Véhicule', 'Nom', 'Prénom', and 'Matr.' It includes a 'Services Concernés' section with checkboxes for 'Expédition', 'Réception', 'Stock', 'Entretien', and 'Maintenance'. There are also sections for 'Types d'opérations' and 'Transport de matières dangereuses?'.
- 2. Security Confirmation:** A screen asking 'Voulez vous revoir les consignes de sécurité ?' with 'Oui' and 'Non' buttons.
- 3. Acknowledgment of Rules:** A screen titled 'VOUS ÊTES RESPONSABLE' listing responsibilities: 'De la mise à quai de votre véhicule', 'Du freinage et du calage/décalage de votre véhicule', and 'De tout incident ou accident survenu de votre fait aux installations, marchandises ou auprès de notre personnel'. It includes a signature line and a date field.
- 4. Safety Instructions:** A screen displaying safety rules: 'Le port de CHAUSSURES de SECURITE sont OBLIGATOIRES en dehors de votre cabine ainsi que sur les quais', 'Le port de gilet de visualisation est obligatoire, restez vigilant à proximité des quais', 'Utiliser des GANTS de SECURITE pour toute opération de manutention', and 'Utiliser les escaliers et voies matérialisées pour vos déplacements'. It ends with a red button 'J'ai pris connaissance des consignes'.

Cette partie est destinée aux chauffeurs visiteurs entrant sur le site.Elle représente un ensemble de succession de pages ( chargées dynamiquement, " application mono-page"). Elle est utilisable en 6 langues à savoir : *français, anglais, espagnol, italien, polonais, néerlandais* .

L'application est constituée des étapes /cheminement suivant :

1. L'utilisateur choisit sa langue déclenchant une mise en forme des textes à partir de sa sélection.
2. L'utilisateur choisit s'il est chauffeur ou visiteur (redirection vers le formulaire correspondant)
3. S'il est chauffeur, il remplit correctement le formulaire (1). S'il saisit une information au mauvais format il en est informé immédiatement par l'apparition de classes d'erreurs sinon apparition de classes valides. Certains champs sont obligatoires.
4. Le chauffeur valide son formulaire en cliquant sur le bouton suivant. A cet instant, on vérifie (à l'aide de fonctions reprenant des expressions régulières javascript) toute sa saisie et il est informé de sa validité.
5. Les consignes de sécurité propres au chauffeur s'affichent sur « 4 écrans » (4) . Sur chacun, le chauffeur doit lire les consignes et valider sa bonne lecture par un clic sur le bouton « j'ai pris

part des consignes ». Il ne peut pas accéder à l'écran suivant tant qu'il n'a pas validé sa lecture.

6. S'affiche alors les instructions liées aux opérations que le chauffeur va effectuer sur le site de VSG. Celles-ci s'affichent avec un intervalle de temps entre chacune d'entre elles (utilisation d'un timer) . Le chauffeur valide sa bonne lecture des instructions.
7. Celui-ci arrive désormais sur le dernier « écran » : il signe numériquement ce qui engage sa responsabilité, puis confirme son entrée par le clic du bouton « suivant » (3).
8. Le processus est de même nature pour le visiteur, seules les informations demandées et les consignes changent

### Extensions

#### « VSGcode »

*Un chauffeur peut être amené à venir régulièrement sur le site de Villeneuve-Saint-Georges.*

**3bis.** Dès que le chauffeur revient sur le site et lors de sa saisie d'informations, il a la possibilité de rentrer son VSGcode. S'il est correct alors son prénom, nom et sa société sont pré-remplis.

Si la date à laquelle il a lu les consignes est inférieure à 14 jours, l'application lui demande alors s'il souhaite revoir les consignes de sécurité (2) :

S'il répond oui la date à laquelle il lit les consignes est mise à jour par la date actuelle.

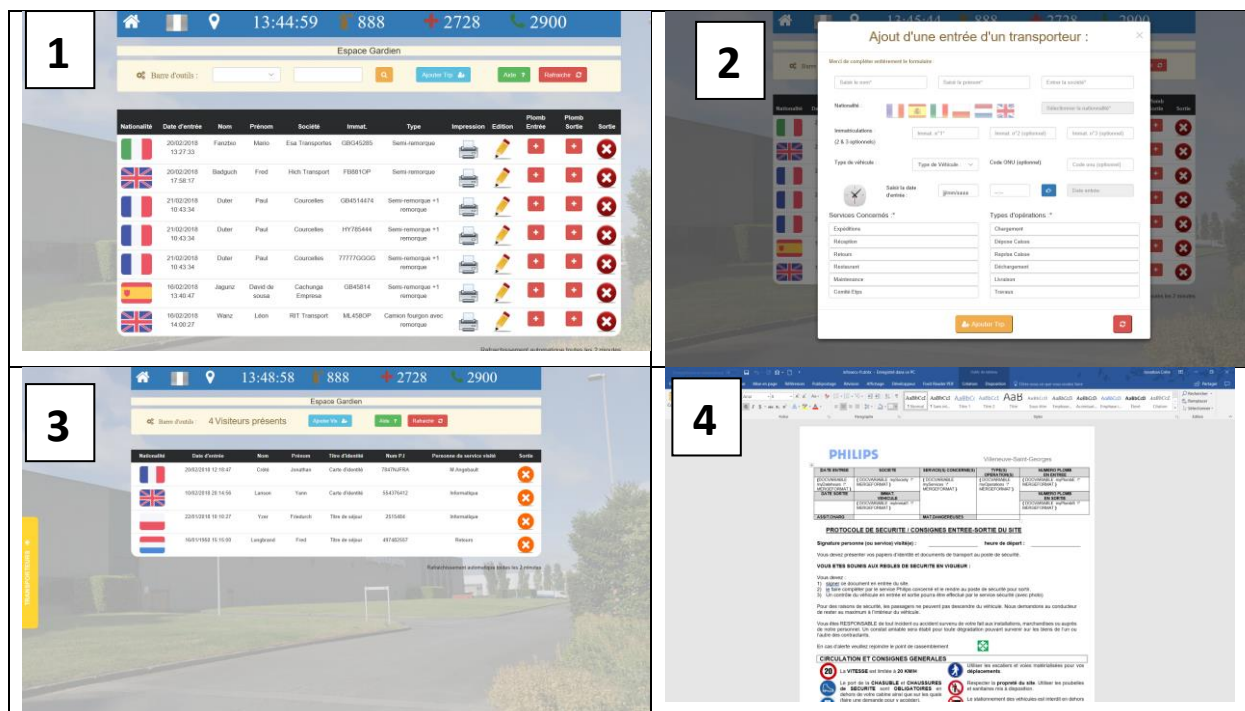
Sinon, il accède directement au dernier 'écran' pour confirmer son formulaire.

**7bis.** Dès sa première entrée et confirmation de son formulaire, chaque chauffeur dispose d'un « VSGcode » créé par l'application et permettant de l'identifier ( 3 premières lettres prénom + 3 premières lettres nom + 3 premières lettres de sa société en MAJ )

#### Navbar

- Le chauffeur peut changer, s'il le souhaite, de langue à tout moment en rappelant la fenêtre modale par le clic sur bouton « flags »
- Le chauffeur peut afficher le plan de VSG à tout moment en appelant la fenêtre modale associé par le clic sur le bouton « map »
- Le chauffeur peut revenir à l'écran d'accueil à tout moment par le clic sur le bouton « home »

## II. Le « Back-Office »

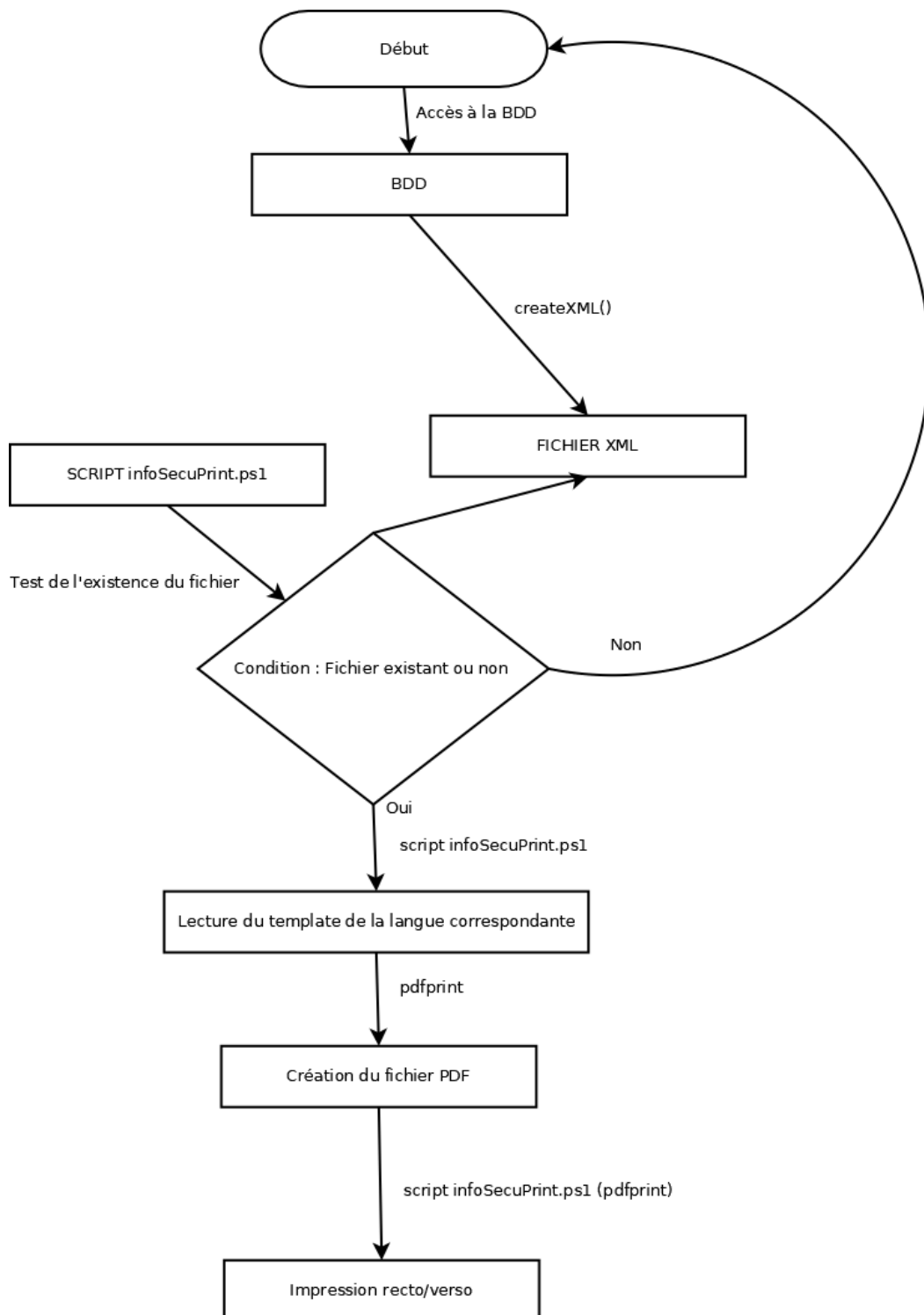


Cette partie est une interface destinée au Gardien/agents du poste de garde. Elle a pour objectif de faciliter l'enregistrement des chauffeurs/ visiteurs présents ou non sur le site pour le gardien.

Cette interface dispose de diverses fonctionnalités :

- ✓ Le détail des transporteurs/visiteurs présents sur le site (dont leurs dates de sortie sont nulles) sous forme d'un tableau de données.
- ✓ La sortie d'un chauffeur symbolisée par un clic sur un bouton (x) et mettant à jour la date de sortie au moment du clic (événements).
- ✓ L'ajout d'un transporteur/visiteur par un formulaire dans une fenêtre modale (2).
- ✓ L'édition des informations d'un transporteur par un formulaire issu d'une fenêtre modale.
- ✓ Ajout rapide des numéros de plomb en entrée et sortie par la simple saisie d'un champ.
- ✓ Possibilité de rechercher un chauffeur par différents critères (nom, prénom, société) qui pour afficher ses informations qu'il ait une date de sortie nulle ou non.
- ✓ L'impression des informations chauffeurs :
  - Le Gardien clique sur le bouton « impression »
  - Les données sont envoyées des tables Access vers un fichier XML crée par une méthode VB.net createXML()
  - Les données du fichier XML sont traitées à l'aide d'un programme PowerShell InfoSecu.ps1
  - Puis à l'aide d'un modèle Word (contenant des variables Doc Variable) préalablement conçu, les données sont remplies dans un nouveau fichier PDF (Instance Word) (4)
  - Le fichier PDF est enregistré
  - Le programme déclenche le processus d'impression à l'aide de la fonction "print-document" et du plugin « pdfprint » (impression recto/verso avec la commande duplex)
  - Les fichiers XML sont supprimés.

- Le programme PowerShell InfoSecu-print.ps1 est exécuté toutes les minutes à l'aide du planificateur de tâches Windows.



## Structure des répertoires sur le serveur LDWDIT01FRVW028

Nom	Modifié le	Type	Taille
img	16/02/2018 17:25	Dossier de fichiers	
OLD_V1	21/02/2018 15:06	Dossier de fichiers	
checkFieldsFunctions.js	01/02/2018 09:45	Fichier de JavaScript	4 Ko
clock.js	21/01/2018 12:38	Fichier de JavaScript	1 Ko
Default.htm	29/11/2017 18:00	Fichier HTM	1 Ko
infosecu.aspx	16/02/2018 17:17	Fichier ASPX	32 Ko
infosecu.aspx.vb	12/05/2017 18:42	Visual Basic Sourc...	1 Ko
infosecu.css	16/02/2018 15:05	Document de feui...	12 Ko
infosecu.js	16/02/2018 17:14	Fichier de JavaScript	17 Ko
infosecuGard.css	16/02/2018 14:14	Document de feui...	5 Ko
infosecuGardien.html	16/02/2018 16:18	Fichier HTML	36 Ko
infosecuGardien.js	16/02/2018 16:14	Fichier de JavaScript	41 Ko
InfosecuReq.aspx	18/05/2017 10:01	Fichier ASPX	1 Ko
InfosecuReq.aspx.vb	16/02/2018 20:32	Visual Basic Sourc...	79 Ko
infosecuTrp.js	14/02/2018 15:42	Fichier de JavaScript	10 Ko
infosecuVis.js	29/01/2018 13:57	Fichier de JavaScript	2 Ko
RegFunctions.js	31/01/2018 17:05	Fichier de JavaScript	2 Ko

**D:\DP-Batch** répertoire contenant les procédures de démarrage et d'arrêt de tous les processus détachés de DP.

**D:\DP-Ope-Data\DB\** Répertoire contenant les tables Access (InfoSecu.accdb)

**D:\DP-Ope-Web\InfoSecu\** (Ensemble de programmes et ressources de l'application)

**\img** Répertoire contenant les images de l'application.

**D:\DP-Ope-Web\css** Répertoire contenant les feuilles de styles propres à Bootstrap utiles à l'application.

**D:\DP-Ope-Web\js** Répertoire contenant les fichiers javascript propres à Bootstrap utiles à l'application.

**D:\DP-Ope-Prg** Répertoire contenant tous les programmes et procédures opérationnels pour le fonctionnement de l'application sur le serveur Windows

**\InfoSecu\Prg\v1.0\** Répertoire contenant le fichier PowerShell servant par exemple pour la mise en forme d'un document Word à partir du Template et à son impression. (InfoSecuPrint.ps1).

**D:\ToDo\InfoSecu** Répertoire contenant les fichiers xml créés par le programme infoSecuPrint.ps1

# Processus de InfoSecuPrint

Sources présentes dans le répertoire \DP-Ope-Prg\InfoSecu\

## **Fichiers associés :**

- InfoSecu-Print.ps1 Fichier contenant les routines pour la lecture des fichier xml, la mise en forme des documents Word, et l'impression de ceux-ci.

## **Clé de registre associé**

- HKLM:\SOFTWARE\vsg\InfoSecu-ope

## **Principe du programme :**

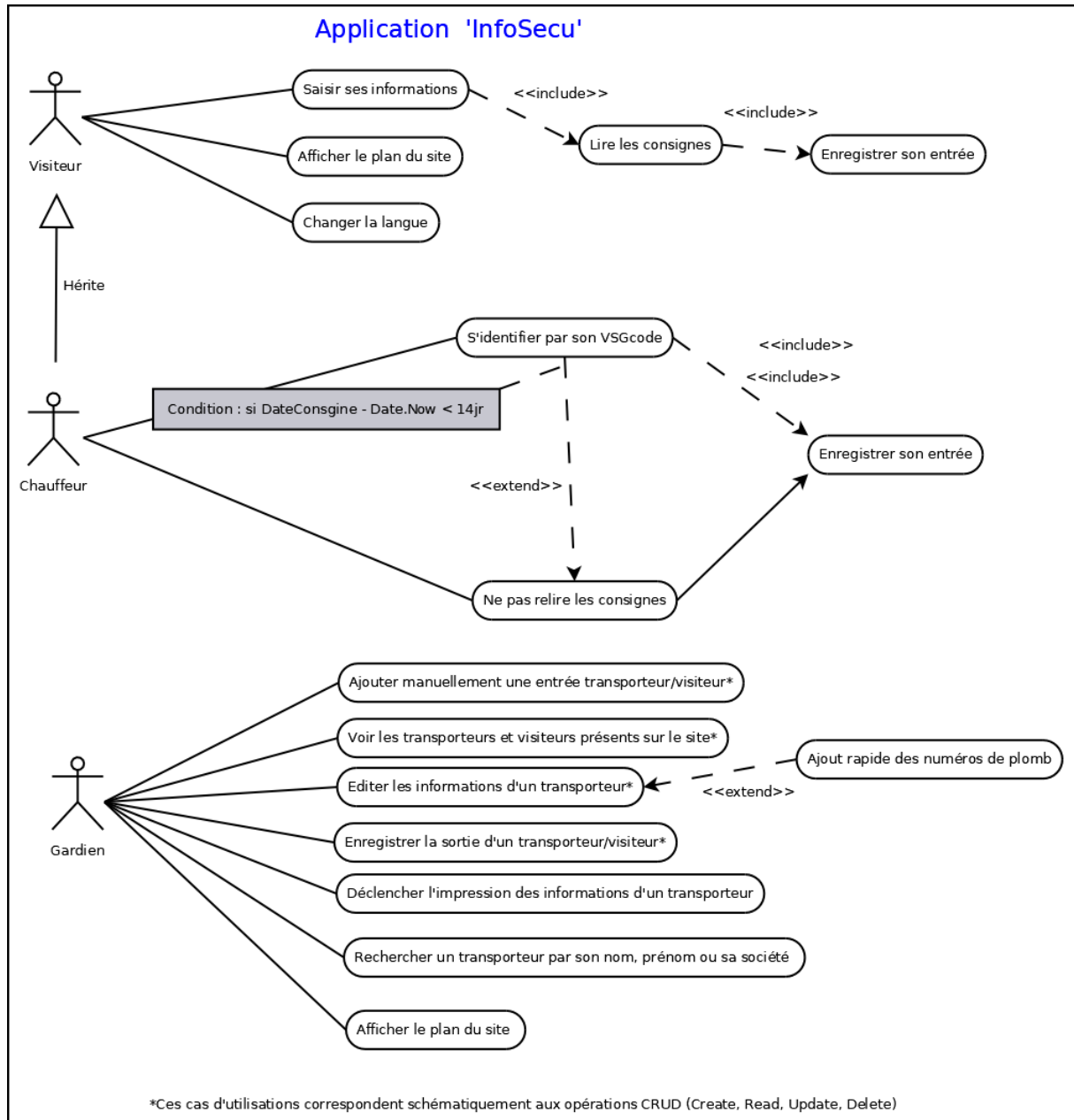
1. Recherche de la présence des clés de registre (contenant, nom imprimante, noms templates dotx et svg ,nom fichierd xml, docx et chemins win)
  - a. Si non présente => Arrêt du processus
2. Recherche de la présence d'un fichier de type **infosecu\_\*.xml** dans le répertoire

### **D:\ToDo\InfoSecu**

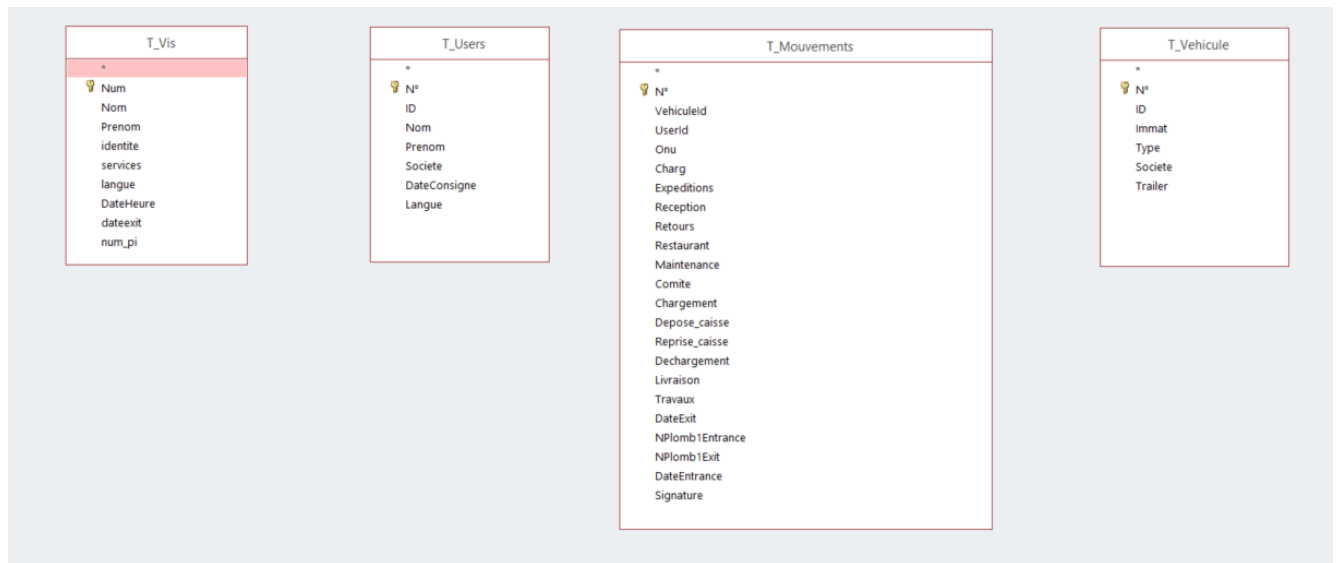
- a. Si non Présence du fichier **xml** → Arrêt du processus
- b. Si Présence du fichier **infosecu\_\*.xml**
  - Chargement du fichier Xml
  - Formatage des différents documents PDF en se servant des modèles présents dans le répertoire **D:\DP-Ope-Prg \InfoSecu\Prg\v1.0\** suivant la langue
  - Sauvegarde du document PDF dans le répertoire **D:\ToDo\InfoSecu**
  - Impression du document du répertoire **D:\ToDo\InfoSecu**
  - Suppression le fichier XML associé
  - Retour en 1
- c. Si Absence de fichier → Attente de x secondes puis retour en 1



# Diagramme de cas d'utilisation



## Diagramme des entités - SQL



### 4 Tables : T\_Vis, T\_USers, T\_Mouvements, T\_Vehicule

*T\_Vis* : Table simple contenant les informations liées aux visiteurs

*T\_Users* : Table simple contenant les informations liées aux chauffeurs

*T\_Mouvements* : Table « liée » contenant les informations propres à **un** mouvement. Un chauffeur peut effectuer un mouvement avec une ou plusieurs immatriculations

*T\_Vehicule* : Table simple contenant les informations liées aux véhicules

### Modèle Relationnel :

*T\_Vis* (Num, Nom, Prenom, identite, services, langue, DateHeure, dateexit, num\_pi)

*T\_Users* (N°, ID, Nom, Prenom, Societe, DateConsigne, Langue)

*T\_Mouvements* (N°, VehiculeId, UserId, Onu, Charg, Expéditions, Reception, Retours, Restaurant, Maintenance, Comite, Chargement, Comite, Chargement, Dechargement, Livraison, Travaux, DateExit, NPlomb1Entrance, NPlomb1Exit, DateEntrance, Signature)

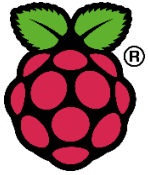
*T\_Vehicule* (N°, ID, Immat, Type, Societe, Trailer)

### Relations entre les tables :

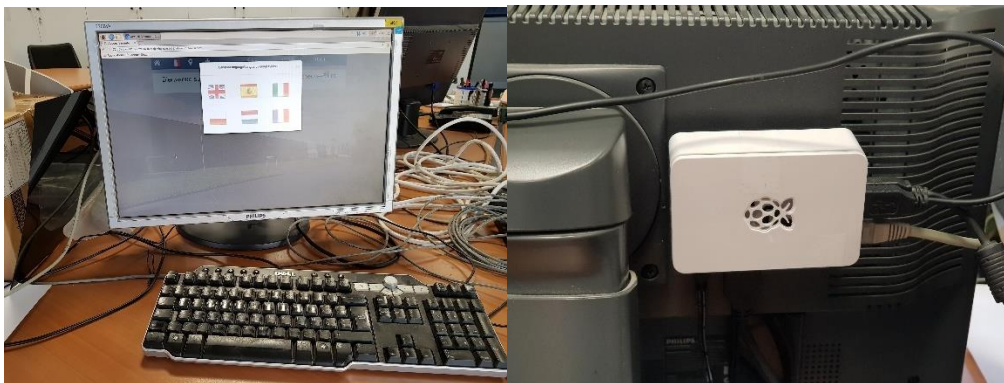
Les champs *T\_Users*.ID, *T\_Mouvements*.VehiculeId, *T\_Mouvements*.UserId, *T\_vehicule*.ID sont indexés à « Oui sans doublons ».

Les données de la base seront exploitées par l'intermédiaire d'objets (instances de classes métier)

## Mise en production



L'application sera chargée sur un **Raspberry pi 2**



### Lignes de commande utiles :

- 1) Installation de Chromium

```
sudo apt-get install chromium-browser
```

- 2) Afin d'avoir un plus bel affichage, Installation des polices MS Core

```
sudo apt-get install ttf-mscorefonts-installer
```

- 3) Installation serveur RDP sur PI

```
sudo apt-get install xrdp
```

- 4) Modification du fichier de démarrage

```
cd /etc/xdg/lxsession/LXDE-pi
```

```
Modifier le fichier autostart
```

```
#@lxpanel --profile LXDE-pi
```

```
#@lxmanfm --desktop --profile LXDE-pi
```

```
#xscreensaver -no-splash
```

```
@xset s off
```

```
@xset s -dpms
```

```
@xset s noblank
```

```
Chromium -kiosk -a http://192.168.1.1
```

5) Suppression de l'écran de veille et de l'économiseur d'énergie

```
@xset s off
```

```
@xset -dpms
```

```
@xset s noblank
```

6) Rebooter

```
sudo shutdown -r now (or sudo reboot)
```

7) Arrêter (Mode Halt)

```
sudo shutdown -h now (or sudo halt)
```

8) Ajouter un utilisateur

```
Sudo adduser NomUtilisateur
```

9) Supprimer un utilisateur (--remove permet de supprimer les fichiers du contexte)

```
sudo deluser --remove-home NomUtilisateur
```

10) Donner les droits sudo

```
sudo visudo
```

Ajouter la ligne suivante :

```
nomUtilisateur ALL=(ALL) NOPASSWD : ALL
```

### Problèmes rencontrés et correction :

- ✗ Problème de droit d'accès au fichier data .acddb car celui-ci a été créé sur une session Windows avec tous les droits administrateurs.
  - ✓ Accord des droits nécessaires par ligne de commande.
- ✗ Les temps de chargement et animations de l'application sont assez long et non-fluide. Cela est en partie dû à des images trop volumineuses et divers éléments dynamiques.
  - ✓ Redimensionnement des images trop volumineuses et suppression des éléments dynamiques non-indispensables.
- ✗ Le Raspberry Pi 2 ne sait pas mettre son horloge à l'heure sans internet (ntp status)
  - ✓ Mise à l'heure manuel : `sudo date 0223155218` (mois/jour/heure/année)
- ✗ La clé « DelaiConsigne » dans le fichier Web.config est manquant.
- ✗ Soucis d'impression des documents word sur l'imprimante voulue (mauvaise config)
  - ✓ Utilisation d'un plugin « pdfprint » qui crée et imprime des fichiers pdf en recto/verso