

# MS2R

## PPE 4 – GROUPE 1



<b>Présentation</b>	2
Contexte	2
<b>Gestion Du Projet</b>	3
Planification et répartition des tâches	3
Pourcentage de travail de chaque membre	4
<b>Partie Technique</b>	5
Mission 1	5
A. Mise en place du DNS	5
B. Création et configuration des utilisateurs	9
C. Configuration SSH	10
D. Développement d'un script pour la reprise sur l'incidents	11
Mission 2	13
A. Modèle Conceptuel de Données	13
B. Réalisation des maquettes	18
C. Implantation de la Base De Données	24
D. Développement de l'application	28
Connexion à l'application	28
Consultation des services	28
Modification d'un service	28
Suppression de service	29
Ajout d'un service	29
Consultation des matériels	30
Modification d'un matériel	30
Suppression de matériel	30
Ajout d'un matériel	30
Consultation des logiciels	31
Ajout d'un logiciel	31
Modification d'un logiciel	31
Suppression de logiciel	31
Fermeture de l'application / Déconnexion	32
<b>Conclusion</b>	33

## Présentation



- **Contexte**

Ce quatrième projet donné par la MS2R avait pour mission d'effectuer certaines tâches tout ceci découper en deux parties.

La première mission consistait à faire évoluer le site de la MS2R de manière à en faciliter la mise à jour de certaines informations, tout ceci en optant pour une solution ergonomique :

Pour cela il aura fallu effectuer une configuration du serveur de manière à octroyer de nouveau compte à destination de différents utilisateurs. Par la suite il a fallu mettre en place une solution pour avoir un accès au site web hébergés grâce à un nom. Avoir aussi par ailleurs un script de reprise sur incident permettant en cas de problème de récupérer les données perdus en mettant en place une opération de sauvegarde et de restauration de la base de données via un menu interactif. Et enfin interdire la connexion au serveur par le protocole SSH et le sécuriser en créant des comptes dédiés aux utilisateurs de la MS2R.

La deuxième mission quant à elle consistait à développer une application sous Visual Studio 2013 grâce à la Programmation Orientée Objet avec le langage Visual Basic.net en y proposant différentes fonctionnalités répondant aux besoins de la MS2R, pour concevoir cette application il a fallu :

Réaliser un Modèle Conceptuel de Données permettant de construire les données de manière à avoir une information cohérente et accessible, et en établir par la suite le schéma relationnel. Suite à cela il a fallu réaliser des maquettes de l'application permettant au responsable de la MS2R de visualiser l'application finale ainsi que toutes les fonctionnalités qui y seraient présenter, et puis enfin après validation du responsable l'application fut coder en travaillant sur des collections ainsi que la base de données implanter à partir du Modèle Conceptuel de Données construit.



- **Pourcentage de travail de chaque membre**

Chaque membre du Groupe 1 à effectuer une part égale de travail, ainsi tout le travail fut réparti de manière équitable selon le diagramme de Gantt. Mais certaines tâches n'ont par ailleurs pas été forcément réalisées par le membre destiné à celle-ci.

Voici donc un récapitulatif de toutes les tâches effectuées par chacun des membres

Membres de l'équipe	Tâches effectuées
M.COINDEVEL ≈ 25%	<ul style="list-style-type: none"> <li>- Création des scripts de sauvegarde</li> <li>- Création des scripts de restauration</li> <li>- Création du menu interactif</li> <li>- Configuration d'Apache</li> <li>- Mise en place du DNS</li> <li>- Configuration du DNS</li> </ul>
Mr AUGUSTIN ≈ 25%	<ul style="list-style-type: none"> <li>- Mise en place du DNS</li> <li>- Configuration du DNS</li> <li>- Création des utilisateurs</li> <li>- Configuration IP</li> <li>- Paramétrage des utilisateurs</li> <li>- Sécurisation de l'accès SSH</li> </ul>
Mr TREPORT ≈ 25%	<ul style="list-style-type: none"> <li>- Réflexion du MCD</li> <li>- Conception du MCD</li> <li>- Réflexion sur la conception des maquettes</li> <li>- Création des classes</li> <li>- Remplissage des collections</li> <li>- Insertion dans la base de données (VB.net)</li> <li>- Rédaction du document de validation</li> <li>- Développement général de l'application</li> </ul>
Mr LEBEAU ≈ 25%	<ul style="list-style-type: none"> <li>- Réflexion du MCD</li> <li>- Conception du MCD</li> <li>- Réflexion sur la conception des maquettes</li> <li>- Création des maquettes</li> <li>- Création du script SQL de la BDD</li> <li>- Insertion des différentes données</li> <li>- Création des classes</li> <li>- Remplissage des collections</li> <li>- Rédaction du document de validation</li> </ul>

## Partie Technique

### Mission 1

#### A. Mise en place du DNS

- **Configuration de l'adresse IP en mode statique :**

Pour donner une configuration IP statique à notre serveur nous accédons au fichier de configuration grâce à la commande suivante :

```
nano /etc/network/interfaces
```

Puis passons à la configuration de celui-ci en modifier et ajouter la nouvelle configuration IP.

```
#*****  
# Configuration lycée  
#*****  
auto eth0  
iface eth0 inetstatic  
    address 172.18.158.31  
    netmask 255.255.248.0  
    gateway 172.18.152.254
```

Une fois fait on relance la carte réseau avec la commande “/etc/init.d/networking restart”.

- **Configuration du DNS :**

Le DNS permet de simplifier l'accès à la page web de la ms2r, en liant l'adresse IP du serveur à un nom de domaine. Comme demandé dans le cahier des charges celui-ci sera ms2r1.fr.

Pour ce faire nous installons bind9 et débutons la configuration :

Nous allons d'abord vérifier l'existence des deux lignes suivantes dans le fichier “named.conf” situé dans le dossier bind9.

```
include "/etc/bind/named.conf.options";    //inclus les options de recherche  
include "/etc/bind/named.conf.local";      // inclus les zones de recherche
```

Ajoutons les zones de recherches dans le fichier “named.conf.local” grâce à la commande suivante “nano /etc/bind/named.conf.local”.

```

-----
// ici on créer les zones de recherche et leur type ainsi que leur fichier de configuration
zone "ms2r1.fr" {           // le nom de zone de recherche directe
type master;                // le type de la zone de recherche directe
file "/etc/bind/db.ms2r1.fr"; // le fichier de configuration de la zone directe
};

zone "158.18.172.in-addr.arpa." { // l'adresse de la zone de recherche inversée
type master;                    // le type de la zone de recherche inversée
file "/etc/bind/db.ms2r1.fr.inv"; // le fichier de configuration de la zone de recherche inversée
};
-----

```

Modifions le fichier `/etc/resolv.conf`, qui indique quels serveurs de noms utiliser.

```

search ms2r1.fr
nameserver 172.18.158.31

```

- **Configuration des fichiers de zones de recherche (directe/inversée)**

Tout d’abord avec le fichier de zones de recherche directe, grâce à la commande “*nano /etc/bind/db.ms2r1.fr*” :

```

-----
$TTL 604800

@      IN      SOA  ms2r1.fr. root.ms2r1.fr. (
                2011031801  ;
                604800      ;
                86400       ;
                2419200     ;
                604800 )    ;

;

@      IN      NS   debian.
@      IN      A    172.18.158.31

debian IN      A    172.18.158.31
-----

```

Puis le fichier de zone de recherche inversée, avec la commande “*nano /etc/bind/db.ms2r1.fr.inv*” :

---

\$TTL 604800

```
@      IN      SOA    ms2r1.fr. root.ms2r1.fr. (
                2011031801  ;
                604800      ;
                86400       ;
                2419200     ;
                604800)    ;
;
```

```
@      IN      NS     ms2r1.fr.
1      IN      PTR    ms2r1.fr
1      IN      PTR    debian.ms2r1.fr
```

---

Et enfin on redémarre le service DNS avec la commande “/etc/init.d/bind9 restart “.

- **Test du serveur DNS**

△ Les résultats suivants ont été effectués en dehors du réseau de la ms2r ce qui explique la différence d'adresse IP afficher ici.

Grâce à la commande “ping” on teste si la configuration du DNS est bonne

```
root@ServMs2r:/home/tectec# ping ms2r1.fr
PING ms2r1.fr (192.168.1.101) 56(84) bytes of data.
64 bytes from 192.168.1.101: icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from 192.168.1.101: icmp_seq=2 ttl=64 time=0.027 ms
64 bytes from 192.168.1.101: icmp_seq=3 ttl=64 time=0.130 ms
64 bytes from 192.168.1.101: icmp_seq=4 ttl=64 time=0.027 ms
^C
--- ms2r1.fr ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 0.019/0.050/0.130/0.046 ms
```

La commande “nslookup” permet de trouver l'adresse IP d'un hôte spécifique ou le nom de domaine d'une adresse IP spécifique, “une recherche inversée”.

```
root@ServMs2r:/home/tectec# nslookup ms2r1.fr
Server:          192.168.1.101
Address:         192.168.1.101#53

Name:   ms2r1.fr
Address: 192.168.1.101
```



La commande “Dig” permet d’avoir des informations sur le résolveur de DNS.

Petit rappel : Un **résolveur DNS** est un serveur, qui va résoudre un nom d’hôte en adresse IP.

```

root@ServMs2r:/home/tectec# dig ServMs2r.ms2r1.fr

; <<>> DiG 9.9.5-9+deb8u17-Debian <<>> ServMs2r.ms2r1.fr
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 58209
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;ServMs2r.ms2r1.fr.                IN      A

;; AUTHORITY SECTION:
ms2r1.fr.                604800  IN      SOA      ms2r1.fr. root.ms2r1.fr. 2011031801 60
4800 86400 2419200 604800

;; Query time: 0 msec
;; SERVER: 192.168.1.101#53(192.168.1.101)
;; WHEN: Fri May 17 19:55:50 +04 2019
;; MSG SIZE rcvd: 87

```

La commande host permet de rechercher un nom d'hôte mais aussi une adresse IP en effectuant une requête de serveur de noms.

```

root@ServMs2r:/home/tectec# host ms2r1.fr
ms2r1.fr has address 192.168.1.101

```

## B. Création et configuration des utilisateurs

Les utilisateurs suivants ont été créés grâce à la commande “adduser”, “ms2r”, “dev” et “tectec”. Suite à cela nous avons configuré la commande “sudo” pour que “tectec” puisse avoir les droits “root” temporairement lors de l'exécution de la sauvegarde et la restauration.

Grâce à la commande “nano /etc/sudoers”, on configure le fichier sudoers pour gérer les droits de “tectec”.

```
#utilisateur tectec  
tectec  ALL=(ALL)      ALL
```

## C. Configuration SSH

Il nous est demandé de bloquer l'accès SSH aux utilisateurs “dev” et “test”.

Pour ce faire nous avons configuré le fichier “sshd\_config”, grâce à la commande “nano /etc/ssh/sshd\_config”.

```
#####  
# Blocage de l'utilisateur  
#####  
#root  
PermitRootLogin no  
#test et dev  
DenyUsers test dev
```

## E. Développement d'un script pour la reprise sur l'incidents

Il nous avait demandé dans le cahier des charges par la suite, de faire un scripte sous un langage propre à lui même qui est le langage **"BASH"** propre à le shell de linux, premièrement c'était de faire des sauvegardes du site de la MS2R et de sa base de données et de pouvoir la restaurer à tout moment.

Au moment du lancement du script il va s'exécuter comme ceci.

```
Choisissais votre choix
1) Sauvegarde Site Web      3) Sauvegarde BDD          5) Quitter
2) Restauration Site Web    4) Restauration BDD
Choisissais votre choix
```

Ces choix correspondent à des commandes qui sont un peu similaires.

- Sauvegarder : en terme en générale en informatique signifie tout simplement que c'est une copie d'un contenu qu'on souhaite sauvegarder vers une redirection différente peu importe où qu'il soit comme par exemple (un disque dur physique ou sur le réseau).

→ La commande qui est dédié le choix 1 est la commande **"CP"** qui est "copie".

```
cp -r /var/www/ms2r1.fr/ms2r1.fr/ /home/tectec/Sauvegarde-SiteWeb/
```

→ Et pour la restauration c'est la même instruction mais en inversant les chemins qu'on utilise.

```
Sudo cp -r /home/tectec/Sauvegarde-SiteWeb/ /var/www/ms2r1.fr/ms2r1.fr/
```

→ Ensuite pour la sauvegarde de la base de données on utilise **"mysql dump"**.

```
mysqldump --user=dev --password=12-Soleil --databases ms2r_app > ms2r.sql
```

Pour la restauration de la base de données il suffit de faire en toute simplicité l'inverse on reprend la même commande mais juste en inversant le **"chevron"**= "**<**".

→ Et voici le script en intégralité :

```

#!/bin/bash
PS3='Choisissez votre choix'
options="Sauvegarde Site Web" "Restauration Site Web" "Sauvegarde BDD" "Restauration BDD" "Quitter";
select opt in $options; do
case $opt in
    "Sauvegarde Site Web")
        cp -r /var/www/ms2r1.fr/ms2r1.fr/ /home/tectec/Sauvegarde-SiteWeb/
        echo "Site Web Sauvegardé"
        ;;

    "Restauration Site Web")
        Sudo cp -r /home/tectec/Sauvegarde-SiteWeb/ /var/www/ms2r1.fr/ms2r1.fr/
        echo "Restauration du Site Web"
        ;;

    "Sauvegarde BDD")
        mysqldump --user=dev --password=[REDACTED] --databases ms2r_app > ms2r.sql
        echo "BDD Sauvegardé"
        ;;

    "Restauration BDD")
        mysqldump --user=dev --password=[REDACTED] --databases ms2r_app < ms2r.sql
        echo "Restauration de la BDD"
        ;;

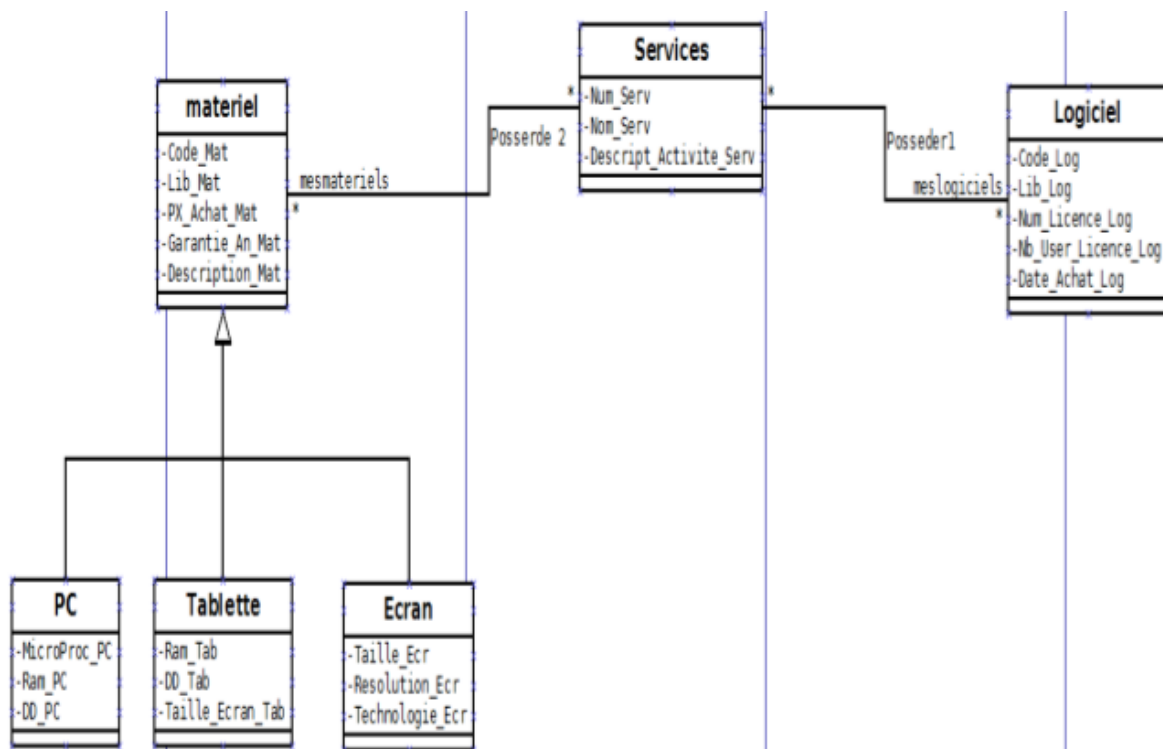
    "Quitter")
        break
        ;;
    *) echo "invalid option $REPLY";;
esac
done

```

## Mission 2

### A. Modèle Conceptuel de Données

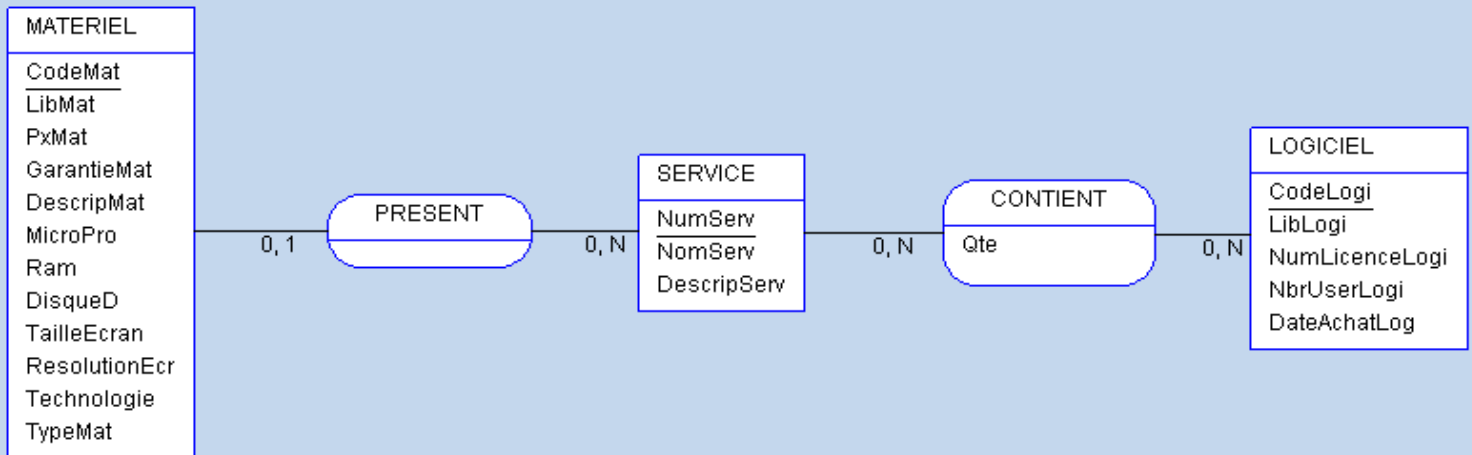
Le Modèle Conceptuel des Données a dû être établie en s’inspirant du diagramme de classe de l’application fournies dans le cahier des charges. Ce diagramme de classe est le suivant :



*Diagramme de classe : Un diagramme de classe est un schéma qui est créé lors de la mise en place de l’UML signifiant Unified Modeling Language (du français Langage de Modélisation Unifié). Un diagramme de classe permet de représenter les différentes classes ainsi que les différentes relations entretenues entre elles.*

Grâce à ce diagramme de classe nous pouvons donc voir que le MCD devra avoir une entité nommée “MATERIEL” permettant de répertorier chaque matériel de manière unique grâce à l’identifiant “Code\_Mat”. Par la suite on aurait une entité “LOGICIEL” permettant elle de répertorier chaque logiciel présent au sein de la MS2R, et enfin une entité “SERVICES” identifié par “Num\_Serv” permettant elle par la suite de répertorier chaque service qui contiendrait différents matériels ainsi que des différents logiciels, ce lien serait effectué par le biais d’une association.

Ainsi après réflexion et entretien avec un responsable de la MS2R nous sommes arrivés au MCD suivant qui permettaient d’organiser de manière la plus cohérente possible les données afin d’en constituer par la suite l’information :



L'entité "MATERIEL" contiendrait donc toutes les informations nécessaires à l'identification du matériel, ainsi si celui-ci était par exemple un PC les champs comme Technologie, ResolutionEcr ne seraient pas renseignés et ce type de matériel serait différencié sur l'application grâce à l'attribut 'TypeMat' qui contiendrait la référence du matériel en question. Cette entité est reliée à l'entité "SERVICE" contenant toutes les données en relations avec un service, un matériel peut donc être présent dans un ou aucun service et un service peut contenir aucun ou plusieurs matériels, en effet ceci est justifié par le fait qu'un matériel (hardware) ne peut pas être partagé entre plusieurs services à la fois et ne peut ainsi être présent que dans un et un seul service (exemple : un clavier ne peut pas être à deux endroits différents en même temps). Par ailleurs un service peut contenir plusieurs matériels ce qui est cohérent.

L'entité "LOGICIEL" quant à elle contenant toutes les données en lien avec différents logiciels, un logiciel comme nous le présente ce MCD peut se retrouver dans plusieurs ou aucun service et un service peut contenir aucun ou plusieurs logiciels. Ceci est justifié par le fait qu'un logiciel (software) peut être partagé entre différents services, en effet si un service a besoin de ce logiciel rien n'empêche un autre service de l'utiliser car ce bien est immatériel. Ce bien peut par ailleurs être limité dans le cas d'un logiciel propriétaire, dans ce cas de l'attribut 'Qte' dans l'association "CONTIENT" prend toute son importance, cela permettra de définir combien d'utilisateurs utilisent ce logiciel et dans quel service. Le nombre total de logiciel restant pouvant être octroyé peut quant à lui être retrouvé grâce à une simple opération.

Voici donc le schéma relationnel de ce Modèle Conceptuel de Données :

**MATERIEL** (CodeMat, LibMat, PxMat, GarantieMat, DescripMat, MicroPro, Ram, DisqueD, TailleEcran, ResolutionEcr, Technologie, TypeMat, #NumServ)

**SERVICE** (NumServ, NomServ, DescripServ)

**LOGICIEL** (CodeLogi, LibLogi, NumLicenceLogi, NbrUserLogi, DateAchatLog)

**CONTIENT** (#NumServ, # CodeLogi, Qte)

Pour vérifier que cette conception de la base de données est bien conforme, on le vérifie avec les 3 étapes de la forme de normalisation. Ses 3 étapes sont les suivantes :

- Pour commencer on effectue la première étape de la forme de normalisation qui consiste à regarder si toutes les relations avec la clé primaire et les autres rubriques sont des dépendances fonctionnelles ayant un sens logique et sont normalisés.
- Ensuite pour passer la deuxième étape d'une forme de normalisation il faut vérifier lorsque la clé primaire est composé si celle-ci a bien des dépendances élémentaires avec toutes ses rubriques, ou plus précisément qu'il faut les informations regroupées de la clé primaire pour obtenir les informations des rubriques de la table. Si par ailleurs la table n'a pas une clé primaire composé alors celle-ci passe automatiquement la deuxième étape de la forme de normalisation.
- Enfin il faut vérifier grâce à la 3<sup>ème</sup> étape de forme de normalisation que toutes les dépendances fonctionnelles liant les rubriques à la clé primaires sont bien directes.

Si la table respecte bien toutes ses contraintes alors notre MCD est bien conforme et les données seront organisés de manière cohérente.

### **La table MATERIEL**

CodeMat --> LibMat, PxMat, GarantieMat, DescripMat, MicroPro, Ram, DisqueD, TailleEcran, ResolutionEcr, Technologie, TypeMat, #NumServ

La table MARCHE respecte la 1<sup>ère</sup> forme de normalisation car la clé primaire CodeMat nous donne bien chaque information en relation avec le code du matériel choisit, en effet grâce à un code de matériel on peut retrouver :

- Son libellé,
- Son prix
- Sa garantie,
- Sa description,
- Les autres informations en lien avec le type de matériel (microprocesseur, disque dur,)
- On peut aussi voir à quel type de matériel celui-ci appartient
- On peut aussi voir le service dans lequel il est (ce champ peut par ailleurs être NULL)

La table MATERIEL passe automatiquement la 2<sup>ème</sup> forme de normalisation car il n'y a aucune dépendance élémentaire dans cette table du fait qu'il n'y a aucune clé composée.



La table MATERIEL respecte la 3<sup>ème</sup> forme de normalisation car la dépendance fonctionnelle qui lie les rubriques à la clé primaire sont toutes directes.

### **La table SERVICE**

NumServ --> NomServ, DescripServ

La table SERVICE respecte la 1<sup>ère</sup> forme de normalisation car la clé primaire NumServ nous donne bien chaque information en relation avec le numéro du service choisi, on peut retrouver grâce à celui-ci :

- Son nom
- Ainsi que sa description

Les relations en sont par ailleurs cohérentes, ainsi la table SERVICE passe la 1<sup>ère</sup> forme de normalisation.

La table SERVICE passe automatiquement la 2<sup>ème</sup> forme de normalisation car il n'y a aucune dépendance élémentaire dans cette table du fait qu'il n'y a aucune clé composée.

La table SERVICE respecte la 3<sup>ème</sup> forme de normalisation car la dépendance fonctionnelle qui lie les rubriques à la clé primaire sont toutes directes.

### **La table LOGICIEL**

CodeLogi --> LibLogi, NumLicenceLogi, NbrUserLogi, DateAchatLog

La table LOGICIEL respecte la 1<sup>ère</sup> forme de normalisation car la clé primaire CodeLogi nous donne bien chaque information en relation avec le code du logiciel choisi, on peut retrouver grâce à celui-ci :

- Son libellé,
- Son numéro de licence,
- Le nombre total d'utilisateur pouvant utiliser ce logiciel,
- La date d'achat du logiciel.

Les relations qu'entretiennent la clé primaire et les rubriques de la table LOGICIEL en sont donc cohérentes, ainsi la table LOGICIEL passe la 1<sup>ère</sup> forme de normalisation.

La table LOGICIEL passe automatiquement la 2<sup>ème</sup> forme de normalisation car il n'y a aucune dépendance élémentaire dans cette table du fait qu'il n'y a aucune clé composée.

La table LOGICIEL respecte la 3<sup>ème</sup> forme de normalisation car la dépendance fonctionnelle qui lie les rubriques à la clé primaire sont toutes directes.

### **La table CONTIENT**

#NumServ, # CodeLogi --> Qte

La table CONTIENT respecte la 1<sup>ère</sup> forme de normalisation car la clé primaire composé de NumServ et de CodeLogi nous permet bien de retrouver la rubrique Qte qui représente le nombre d'utilisateurs du logiciel dans un service. Ainsi cette relation en est cohérente et respecte bien la 1<sup>ère</sup> forme de normalisation.

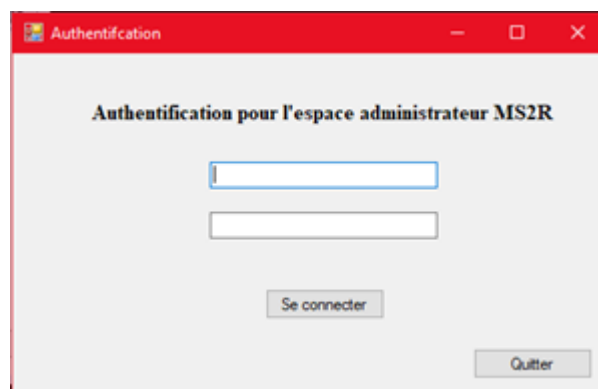
La table CONTIENT respecte la 2<sup>ème</sup> forme de normalisation car la dépendance liant la rubrique Qte à la clé primaire est une dépendance élémentaire. En effet il nous faut le numéro du service ainsi que le logiciel voulues pour voir combien d'utilisateurs utilise ce logiciel dans ce service.

La table CONTIENT respecte la 3<sup>ème</sup> forme de normalisation car la dépendance fonctionnelle qui lie les rubriques à la clé primaire sont toutes directes.

## B. Réalisation des maquettes

La réalisation des maquettes a dû être effectué de manière à répondre à tous les besoins voulues par la MS2R. Dans un premier temps, nous nous sommes concerté afin de voir la meilleure façon de proposer différentes fonctionnalités toutes en prenant compte du cahier des charges, des diagrammes de classe fournies dans le cahier des charges. Par la suite nous avons effectué un premier entretien avec un responsable de la MS2R afin de mieux connaître les besoins attendus. Cet entretien a donc conclu les attentes et besoin attendues, il a fallu donc par la suite créer les maquettes destinées à proposer aux responsables les fonctionnalités finales de l'application. Voici donc les maquettes validées par le responsable de la MS2R et une brève explication de celle-ci.

A l'ouverture de l'application, l'administrateur se retrouvera face à ce formulaire qui permet de sécuriser l'accès au différentes fonctionnalités de l'application. L'administrateur devra s'identifier pour pouvoir se connecter.



Authentification

Authentification pour l'espace administrateur MS2R

Champ de saisie pour l'identifiant

Champ de saisie pour le mot de passe

Se connecter

Quitter

Suite à la connexion de l'administrateur, celui-ci se retrouve sur le menu d'administration qui lui permettra d'utiliser toutes les fonctionnalités de l'application.



MenuAdmin

Bienvenue "Nom et Prénom de membre"

Services

Matériels

Logiciels

Consultation

Ajout d'un service

Ajout de matériel

Ajout logiciel

Se déconnecter

Quitter l'application

Lors de la consultation d'un service, celui-ci se retrouve sur ce formulaire, lui permettant d'effectuer différentes opérations sur les services.

**Gestion des services**

Créer service

Liste services

N°	Nom Service
1	Comptabilité 1
2	Ressource Humaine 1

Logiciels

Microsoft Office

Matériels

ASUS  
ACER  
TOSHIBA  
Fujitsu

Description

Service de gestion des factures

Modifier un service   Supprimer un service   Retour au menu

En double cliquant sur l'un des services, l'administrateur pourra donc voir les logiciels du service ainsi que les matériels qui le compose. Il pourra par ailleurs voir sa description et pourra par la suite modifier ce service ou le supprimer (avec confirmation avant suppression).

Si l'administrateur souhaite modifier la composition d'un service, il se retrouvera face à ce formulaire qui lui permet de basculer un matériel vers le stock ou le service ou encore de retirer ou ajouter une

quantité d'utilisateur pour un service. Il peut aussi par la suite modifier le nom du service ainsi que sa description s'il le souhaite.

Service :  Description:

Matériels

Matériel(s) du service		
N°	Nom matériel	Type
1	ASUS	PC
2	ACER	PC
3	TOSHIBA	PC
4	Fujitsu	PC

Matériel(s) disponible(s)

N°	Nom matériel	Type
----	--------------	------

>

<

Logiciels

Logiciel(s) du service		
N°	Nom logiciel	Utilisateurs
3	Microsoft Office	10

Logiciel(s) disponible(s)

N°	Nom logiciel	Utilisateurs
2	Windows 7 Professionnel	25
3	Microsoft Office	10
4	CCleaner Pro	100

>

<

Quantité

Terminer

Lorsque l'administrateur souhaite créer un service il sera dirigé sur ce formulaire, qui lui permettra d'entrer un nom et une description et d'ajouter directement s'il le souhaite des matériels ainsi que des logiciels grâce au bouton radio.

### Création d'un nouveau service

Nom du nouveau service

Description du service

Ajout de(s) logiciel(s) et matériel(s)

☐ Procéder à l'ajout

☒ Ajouter plus tard

Valider

Annuler

Si l'administrateur souhaite consulter les différents matériels, il sera sur ce formulaire qui lui permet en double cliquant de modifier le matériel, de le supprimer ou de voir si ce matériel est dans un service ou est en stock.

#### Matériels

N°	Nom matériel	Type
1	ASUS	PC
2	ACER	PC
3	TOSHIBA	PC
4	Fujitsu	PC
5	Alienware	PC
6	Macbook	PC

Le service utilisant ce matériel :

Comptabilité 1

Supprimer le matériels

Modifier matériels

Retour au menu

Lorsque l'administrateur souhaitera rajouter un matériel, il sera confronté à ce formulaire qui lui permettra de renseigner toutes les informations en lien avec un matériel. (Nom, type, quantité, informations en lien avec le type de matériel, ...)

### Ajout matériel

**Informations**

Nom du matériel:

Prix:

Garantie: dimanche 19 mai 2019

Type

☐ PC

☐ Tablette

☒ Ecran

Description:

**Ecran**

Taille de l'écran:

Résolution:

Technologie:

Lorsque l'administrateur souhaitera par ailleurs consulter les logiciels, il sera dirigé sur ce formulaire qui lui permettra en double cliquant sur un logiciel devoir dans quel service ce logiciel est, mais aussi de voir le nombre total d'utilisateurs pouvant utiliser ce logiciel ainsi que le total de ceux qui l'utilise. L'administrateur pourra aussi voir en effectuant un double clic sur le service, voir combien d'utilisateurs l'utilise dans ce service. Par ailleurs, l'administrateur pour aussi supprimer ou modifier un logiciel.

### Liste des logiciels

N°	Nom du logiciel	Nombre d'utilisateurs	Date
1	Windows 10 Professionel	-28	23/01/2019
2	Windows 7 Professionel	25	23/01/2019
3	Microsoft Office	10	23/01/2019
4	CCleaner Pro	100	31/05/2019

**Informations complémentaires**

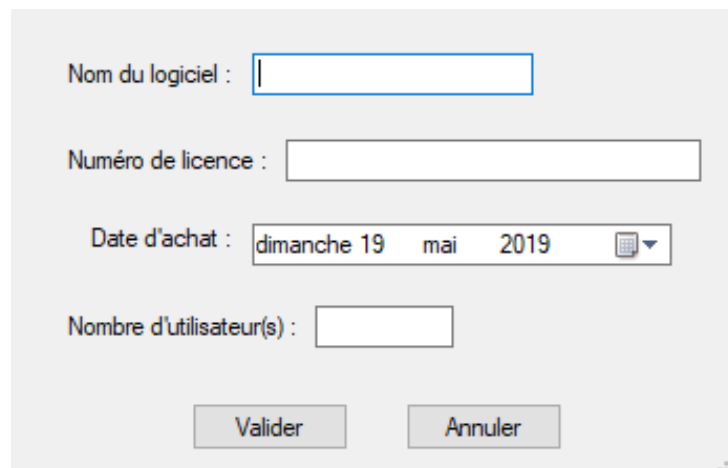
Nombre d'utilisateur restant : 25

Nombre d'utilisateur du logiciel: 0

Le(s) service(s) utilisant ce logiciel :

N°	Nom du service

Enfin si l'administrateur souhaite ajouter un logiciel, il sera envoyé sur ce formulaire, lui permettant d'ajouter un logiciel ainsi que toutes les informations en lien avec celui-ci. (Nom du logiciel, quantité d'utilisateurs pouvant être octroyés, date d'achat, ...)



A screenshot of a software license activation dialog box. The dialog has a light gray background and contains the following fields and buttons:

- Nom du logiciel :** A text input field with a blue border.
- Numéro de licence :** A text input field.
- Date d'achat :** A date picker showing "dimanche 19 mai 2019" with a calendar icon and a dropdown arrow.
- Nombre d'utilisateur(s) :** A text input field.
- Buttons:** Two buttons at the bottom, "Valider" (Validate) and "Annuler" (Cancel), both with a light gray background and a thin border.

Nous allons poursuivre avec les quelques insertions dans la base de données, la partie fonctionnement de l'application est expliqué dans la "Partie I - D) Développement de l'application".

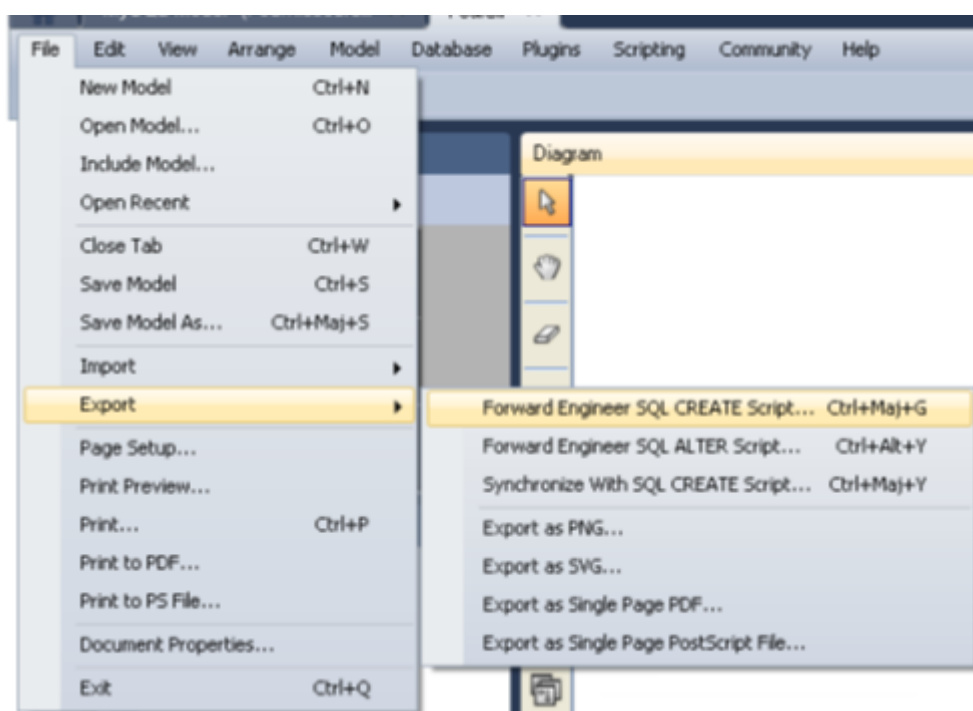


## C. Implantation de la Base De Données

Pour procéder à l'implantation de la base de données, il a fallu procéder de manière à éviter les erreurs. Pour cela il a fallu créer les tables de façon à ce que les références de clés étrangères soient préalablement créées, ainsi voici comment nous avons procédé pour créer la BDD.

Tout d'abord nous avons créé le schéma sous MySQL Workbench de manière à ce qu'on puisse par la suite en générer un script SQL.

Après avoir réalisé le schéma sous WorkBench fut construit, il a fallu générer le script SQL grâce à l'application. Pour ce faire, il a fallu aller dans l'onglet « Files » puis choisir le dossier puis « Export » et enfin choisir l'option comme ci-dessous :



Par la suite il a fallu préciser le dossier de réception du script SQL afin d'accueillir le script d'insertion de la base de données qui est le suivant.

Lors de la création de la base de données, on procède tout d'abord à la création des tables n'ayant aucune contrainte, comme la table "Admin" contenant les identifiants et mot de passe des administrateurs qui servira lors de la connexion à l'application :

```
CREATE TABLE IF NOT EXISTS `admin` (
  `Login` varchar(50),
  `MDP` varchar(50),
  PRIMARY KEY (`Login`)
);
```

On créera la table Logiciel par la suite :

```
CREATE TABLE IF NOT EXISTS `logiciel` (
  `CodeLogi` int NOT NULL AUTO_INCREMENT,
  `LibLogi` varchar(150) NOT NULL,
  `NumLicenceLogi` varchar(100),
  `NbrUserLogi` int DEFAULT NULL,
  `DateAchatLog` date DEFAULT NULL,
  PRIMARY KEY (`CodeLogi`)
);
```

Et enfin la table Service :

```
CREATE TABLE IF NOT EXISTS `service` (
  `NumServ` int NOT NULL AUTO_INCREMENT,
  `NomServ` varchar(50) ,
  `DescripServ` varchar(200) ,
  PRIMARY KEY (`NumServ`)
);
```

On poursuit sur la création des tables ayant des contraintes de clés étrangères, sur ses tables, comme la table “Materiel” qui a une contrainte de clé étrangère sur la table Service car il faut lier un matériel à un service si celui-ci est présent dans celui-ci, par ailleurs ce champ peut-être ‘NULL’ car le matériel peut tout simplement être en stock et n’est donc pas encore au sein d’un service :

```
CREATE TABLE IF NOT EXISTS `materiel` (
  `CodeMat` int,
  `LibMat` varchar(100),
  `PxMat` int NOT NULL,
  `GarantieMat` date DEFAULT NULL,
  `DescripMat` varchar(200) ,
  `MicroPro` varchar(50) ,
  `Ram` varchar(50) ,
  `DisqueD` varchar(50),
  `TailleEcran` varchar(50),
  `ResolutionEcran` varchar(50),
  `Techonologie` varchar(100),
  `TypeMat` varchar(15) ,
  `NumService` int,
  PRIMARY KEY (`CodeMat`)
);
```

Pour finir, on crée la table “Contient” qui permet de lier un service à un logiciel et de retrouver la quantité d'utilisateurs de ce même logiciel au sein d'un service (le nom de la table a par ailleurs été changé pour plus de lisibilité) :

```
CREATE TABLE IF NOT EXISTS `service_has_logiciel` (
  `SERVICE_NumServ` int(11) NOT NULL,
  `LOGICIEL_CodeLogi` int(11) NOT NULL,
  `Utilisateurs` int(11) NOT NULL,
  PRIMARY KEY (`SERVICE_NumServ`, `LOGICIEL_CodeLogi`)
);
```

Lorsque la création de la base de données fut terminée, nous avons pu insérer des données à l'intérieur de manière à agrémenter le logiciel au démarrage. Il a fallu ici aussi respecter l'ordre des insertions afin de ne pas avoir d'erreurs. Voici donc ci-dessous un extrait.

### Étape 1 :

Tout d'abord, on commence par la table Admin qui ne contient aucune contrainte.

```
INSERT INTO `admin` (`Login`, `MDP`) VALUES
('k.champagnat', '12-Soleil&'),
('s.kosmalski', '12-Soleil&');
```

### Étape 2 :

Ensuite on insère les données pour la table Logiciel qui elle aussi ne contient aucune contrainte de clé étrangère.

```
INSERT INTO `logiciel` (`CodeLogi`, `LibLogi`, `NumLicenceLogi`,
`NbrUserLogi`, `DateAchatLog`) VALUES
(1, 'Windows 10 Professionel', 'NZXT-NTM8-69SX-BYTE', 20, '2019-01-23'),
(2, 'Windows 7 Professionel', 'UI06-TYKO-D7R5-AHOK', 10, '2019-01-23');
```

### Étape 3 :

On continue les insertions avec la table Service.

```
INSERT INTO `service` (`NumServ`, `NomServ`, `DescripServ`) VALUES
(1, 'Comptabilité 1', 'Service de gestion des factures'),
(2, 'Ressource Humaine 1', 'Service de gestion des employés');
```

### Étape 4 :

Par la suite on peut commencer à insérer dans les tables ayant des contraintes de clés étrangères. On commence ici avec la table “Matériel” ayant une contrainte sur le champ ‘NumServ’ en relation avec le champ ‘NumServ’ de la table “Service”.

```
INSERT INTO `matériel` (`CodeMat`, `LibMat`, `PxMat`, `GarantieMat`,
`DescripMat`, `MicroPro`, `Ram`, `DisqueD`, `TailleEcran`, `ResolutionEcran`,
`Technologie`, `TypeMat`, `NumService`) VALUES
(1, 'ASUS', 5000, '2020-04-23', 'Ordinateur très performant ', 'Intel Core I9',
'64GB', '10To', NULL, NULL, NULL, 'PC', 1),
```

```
(2, 'ACER', 2000, '2020-04-23', 'Ordinateur performant ', 'Intel Core I5 ',  
'16GB', '1To', NULL, NULL, NULL, 'PC', 1);
```

**Étape 5 :**

Enfin, on insère les données dans la table “Service\_has\_logiciel” qui contient les logiciels présents au sein de chaque service ainsi que la quantité d’utilisateurs utilisant ce logiciel au sein de ce même service.

```
INSERT INTO `service_has_logiciel` (`SERVICE_NumServ`, `LOGICIEL_CodeLogi`,  
`Utilisateurs`) VALUES  
(1, 2, 5),  
(1, 3, 10);
```

## D. Développement de l'application

Après la validation des maquettes finales par les responsables de la MS2R. Nous avons débuté le développement de l'application qui se base sur le diagramme de classe donné dans le cahier des charges.

### Connexion à l'application

En premier temps, nous avons établi la connexion à la base de données pour remplir les collections de l'application afin de les manipuler. Suite à cette connexion établie, nous avons créé les différentes classes ainsi que les héritages de classes qui permettra la création des objets. Lorsque tout ceci fût terminé nous avons entamé le remplissage des collections grâce à la connexion à la base de données qui permet par le biais de requête SQL de récupérer les informations et de les mettre dans les collections correspondantes. Voici un exemple de requête SQL contenues dans l'application de la MS2R :

```
requete = "SELECT * FROM materiel"
```

Lorsque chacune des collections est remplies avec les informations contenues dans la base de données, le formulaire de connexion s'affiche. Suite à cela si l'administrateur renseigne un mot de passe ou un pseudo erroné qui n'est pas le même que celui contenus dans les collections il ne peut pas accéder aux fonctionnalités de l'application. Par ailleurs lorsque celui-ci renseigne les bonnes informations il accède au menu de l'application.

### Consultation des services

Quand l'administrateur souhaitera consulter les différents services, l'application ira chercher chaque service dans la collection contenant tous les services pour pouvoir remplir la liste des services, les boutons de suppression et de modification seront par ailleurs grisés car aucun élément de la liste n'aura été sélectionné.

Si l'administrateur double clic par ailleurs sur un de ses services, en fonction de l'élément sélectionné l'application ira chercher dans la collection de matériel du service ainsi que la collection de logiciel du service tous ceux qui sont présent dans ce même service et l'ajoutera dans les listes de matériels et logiciel du service, ce double clic permettra aussi par la suite de voir la description du service.

Lorsque l'administrateur aura par ailleurs sélectionné un service, il pourra cliquer sur les boutons permettant la modification ainsi que la suppression d'un service.

### Modification d'un service

Après avoir choisi le service à modifier, un autre formulaire apparaîtra. Sur ce formulaire comportera quatre listes :

- Une contenant tous les logiciels du service
- Une contenant tous les matériels du service

- Une contenant tous les logiciels disponibles pouvant être attribuer à un service
- Une contenant tous les matériels disponibles (en stock)

Lors du chargement les listes de matériels du service et les listes de logiciels du service seront remplies en fonction du service sélectionné. Par ailleurs les matériels ne seront affichés que s'ils sont disponibles (en stock) tout comme les logiciels.

Ainsi par le biais de deux boutons, l'administrateur pourra basculer un matériel soit vers le stock soit vers le service. Cette action aura pour effet de rafraîchir les deux listes pour donner les informations à jour mais aura surtout pour effet de supprimer le logiciel du stock pour l'attribuer au service.

Par ailleurs, lorsque l'administrateur souhaitera par le biais des deux boutons de la liste des logiciels basculer un logiciel dans le service ou l'enlever il devra saisir grâce à une zone de texte le nombre d'utilisateur à enlever ou à mettre. De cette manière les deux listes seront mis à jour. Un traitement est aussi effectué à ce niveau afin d'empêcher l'administrateur de saisir un nombre d'utilisateurs qui serait plus grand que ce qu'il a de disponible et permet aussi d'enlever le logiciel si celui-ci n'est plus disponible. S'il s'avère qu'un logiciel est déjà présent dans le service, alors le nombre d'utilisateurs sera ré ajuster en cause.

Les boutons sont par ailleurs accessibles que si l'administrateur clique sur un élément d'une liste et lorsque ses boutons sont utilisés les modifications sont effectuées tout de suite sur les collections de logiciels du service et de matériels du service.

Pour finir l'administrateur pourra aussi changer le nom du service ainsi que sa description par le biais de deux zones de textes.

### **Suppression de service**

Lorsque l'administrateur voudra supprimer un ou plusieurs services, l'élément sélectionné sera recherché, une fois trouvé il sera supprimé et les logiciels contenus dans ce service ainsi que les matériels qui le composent seront ré attribués à ce qu'il y a de disponibles.

Par la suite le service sera supprimé de la collection de service.

### **Ajout d'un service**

Quand l'administrateur souhaitera ajouter un service, il devra en renseigner le nom ainsi que sa description. L'administrateur pourra par la suite choisir d'attribuer au service qu'il a créé différents matériels et logiciels ou de le faire plus tard.

Le service en sera par la suite ajouté à la collection de service. Si l'administrateur veut par ailleurs ajouter à ce service différents matériels et logiciels alors le déroulé sera le même que pour la Modification d'un service.

### **Consultation des matériels**

Si l'administrateur souhaite consulter les matériels, le formulaire chargera tous les matériels depuis la collection de matériel en remplissant la liste des matériels.

Suite à cela lorsque l'administrateur effectuera un double clic sur un des matériels de la liste, il verra si ce matériel est dans un service ou non.

Lors de la sélection d'un matériel, l'administrateur pourra accéder au boutons de suppression et de modification de matériel qui seront dégriser.

### **Modification d'un matériel**

Après avoir sélectionné un matériel, si l'administrateur souhaite le modifier il aura accès à un formulaire qui récupérera les informations du matériel. Il ne pourra pas par ailleurs en modifier le type. Quand l'administrateur validera sa modification les outils permettant de modifier les différentes informations seront repris et les informations du matériel contenus dans la collection de matériel ainsi que les informations des classes héritées de matériels en seront modifiés.

### **Suppression de matériel**

Après avoir sélectionné un matériel, si l'administrateur le supprimer, il pourra après confirmation supprimer ce matériel définitivement si celui-ci n'est pas présent dans un service. Cela aura pour effet de le supprimer de la collection de matériel.

Mais si ce matériel est présent dans un service une autre confirmation est demandé à l'administrateur afin de supprimer réellement ce matériel. S'il clique sur "OUI" alors le matériel sera supprimé de la collection et du service dans lequel il se trouve, cela aura donc pour effet de le supprimer de la collection de matériel mais aussi de la collection de matériel du service qui a ce matériel.

### **Ajout d'un matériel**

Si l'administrateur souhaite ajouter un matériel, il le pourra par le même formulaire de modification sauf qu'ici les champs seront vides et il pourra déterminer le type de matériel qu'il souhaite ajouter.

Lorsque l'administrateur validera son choix, le matériel sera ajouté à la collection de matériel et en fonction de son type, il sera ajouté à la collection correspondante (Collection de PC, Collection de Tablette, Collection d'écran)

### Consultation des logiciels

Quand l'administrateur souhaitera consulter les logiciels, le formulaire se chargera avec la collection de logiciel et en remplira la liste des logiciels.

Lorsque l'administrateur cliquera sur l'un des logiciels les deux boutons de suppression et de modification seront dégrisés, s'il double clic par ailleurs sur l'un des logiciels il pourra voir ses différentes informations :

- Le nombre de licence pouvant être donné (nombre d'utilisateurs pour un service)
- Le nombre d'utilisateurs total utilisant ce logiciel
- Le(s) service(s) dans lequel il se trouve

Quand il effectuera un double clic sur l'un des services utilisant ce logiciel il pourra par ailleurs voir combien d'utilisateurs sont utilisateurs de ce logiciel au sein du service sélectionné en allant chercher l'information dans la collection de logiciels du service.

### Ajout d'un logiciel

Lors de l'ajout d'un logiciel, l'administrateur pourra entrer toutes les informations en lien avec le logiciel comme le nom, la date d'achat de ce logiciel, ...

Quand celui-ci validera ce logiciel sera enregistrer dans la collection de logiciel.

### Modification d'un logiciel

Si l'administrateur souhaite modifier un logiciel sélectionné dans la consultation de logiciel, il le pourra par le même formulaire que l'ajout 'un logiciel. Sauf qu'ici les champs seront pré remplis des informations en lien avec le logiciel sélectionné.

Quand l'administrateur validera son choix l'objet contenant toutes les informations du logiciel en sera modifier.

### Suppression de logiciel

Si l'administrateur souhaite supprimer un logiciel, il devra confirmer son action. Suite à cela si le logiciel n'est dans aucun service il en sera définitivement supprimer de la collection.

Mais si le logiciel est présent dans un ou plusieurs services une autre confirmation sera demandé à l'administrateur afin de le prévenir que ce logiciel est présent au sein de service et que sa suppression entraînera aussi la suppression du logiciel dans ses services. Si l'administrateur confirme son choix suite à cela alors le logiciel sera supprimé de la collection de logiciel mais sera aussi supprimer de la collection de logiciel de tous les services qu'il contient.



**Fermeture de l'application / Déconnexion**

L'administrateur, une fois toutes ses opérations terminées pourra se déconnecter ou quitter l'application par le biais d'un bouton. Cette action aura pour effet de supprimer toutes les données présentes dans un ordre bien précis de manière à ne générer aucune erreur.

Suite à la suppression toutes les informations contenues dans les collections seront insérés dans la base de données de manière ordonnée pour ne générer ici aussi aucune erreur.

Ainsi, quand un autre administrateur souhaitera se connecter les informations qui ont été modifiés dans la base de données seront recharger et donc toutes les modifications apportées seront visible.

## **II. Conclusion**

Dans sa globalité, le projet fut réalisé à peu près dans les temps avec la rencontre de quelques problèmes et difficultés comme un serveur qui n'était pas au point et qui a dû être changé où des problèmes liés à la configuration du DNS.

D'autres problèmes perçues aussi en développement qui étaient la manipulation de certains outils sous Visual Studio 2013 ou encore la manipulation des collections issue d'un héritage et le remplissage de la base de donnée à sa fermeture.

Toutefois, le serveur est opérationnel et l'application est au point. Ce projet personnalisé encadrée 4 a pu être terminé dans les temps et est donc prêt à être présenter ce mardi 21 mai 2019.