

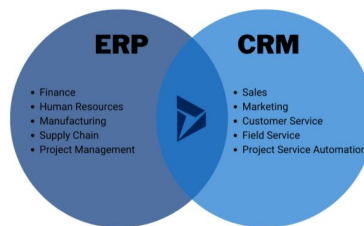
# SAE51 - projet 3

## Installation d'un ERP/CRM

IUT RT Rouen

### 1 Introduction

Toutes les entreprises ont besoin aujourd'hui d'un "progiciel de gestion" (aussi appelé ERP et/ou CRM<sup>1</sup>) pour gérer le flux d'informations (clients/fournisseurs, devis/factures, commandes/achats, production, stocks, etc.).



Plusieurs solutions s'offrent aux dirigeants. On peut faire appel à un prestataire externe type ESN<sup>2</sup>, qui proposera une solution complète "clés en mains", hébergée soit chez le prestataire, soit dans un cloud public. L'inconvénient c'est que d'une part on sera dépendant de ce prestataire, d'autre part, on a pas la maîtrise des données, ce qui peut être un point bloquant pour certains secteurs économiques dits "sensibles".

Une autre solution consiste à utiliser un logiciel dédié, qu'on pourra soit héberger soi-même dans un cloud public (avec encore une fois la problématique des données), soit hébergé en interne.

C'est cette dernière option qui sera étudiée dans ce projet.

L'objectif est de vous donner les compétences pour l'installation et la configuration d'un outil clé d'une entreprise, tel que devrait le faire un responsable informatique dans une PME.

### 2 Description du projet

Vous êtes responsable informatique dans l'entreprise XXX. La direction souhaite migrer d'une solution ERP/CRM externalisée vers une solution hébergée en interne, basée sur le progiciel "Dolibarr". Elle a réussi à récupérer un export des données du prestataire actuel, sous la forme d'un ensemble de fichiers CSV (fichiers clients, fournisseurs, factures, commandes, etc.)

La direction vous donne le cahier des charges suivants :

Faire l'étude de la mise en place d'un Dolibarr fonctionnel sur un serveur dédié hébergé dans l'entreprise. Ceci implique de considérer à la fois le côté installation, l'aspect import des données, et l'aspect sauvegarde des données.

L'objectif final est d'arriver à une solution respectant les contraintes suivantes :

---

1. voir [https://en.wikipedia.org/wiki/Enterprise\\_resource\\_planning](https://en.wikipedia.org/wiki/Enterprise_resource_planning) et [https://en.wikipedia.org/wiki/Customer\\_relationship\\_management](https://en.wikipedia.org/wiki/Customer_relationship_management)

2. [https://fr.wikipedia.org/wiki/Entreprise\\_de\\_services\\_du\\_num%C3%A9rique](https://fr.wikipedia.org/wiki/Entreprise_de_services_du_num%C3%A9rique)

- L'installation sera automatisée, via le lancement d'un unique script `install.sh` qui va à la fois installer Dolibarr mais aussi le SGBD nécessaire.
- L'import des données exportées depuis l'ancien système sera automatisé via un unique script `import_csv.sh`
- Afin de s'affranchir de l'OS sous-jacent du serveur, toute l'installation (Dolibarr + SGBD) sera de préférence "dockerisée".
- L'aspect sauvegarde est crucial. On veut pouvoir à un instant  $t$  faire une sauvegarde de toutes les données, et être capable, via la procédure d'installation ci-dessus, de repartir de 0 et tout récupérer (contexte d'un "PRA"<sup>3</sup> après incident).

Note : les scripts évoqués ci-dessus pourront bien sur être des scripts "maitres", qui en appellent d'autres.

Ce travail se fera en plusieurs étapes.

1. **Impératif** : Découverte de Dolibarr avec installation simple sur une VM (ou un conteneur) Debian.

Il faudra ensuite lancer les pages d'installation de Dolibarr et faire l'install en mode "manuel" : configuration de l'accès à la base de données, création du compte "superadmin" et d'un compte "user", ajout des modules, etc.

Puis on pourra passer à la phase d'importation des données, à partir de fichiers CSV. Ceci se fait via le menu "outils".

Le logiciel étant assez complexe, il est important d'arriver à en comprendre la philosophie avant de pouvoir envisager l'automatisation de l'installation.

2. Il faudra ensuite trouver un moyen d'importer les données existantes. Deux techniques peuvent être envisagées :

- Soit utiliser les fonctionnalités d'import intégrées à Dolibarr (menu "outils")
- Soit analyser les tables créées par Dolibarr de façon à pouvoir construire un script d'importation des données existantes directement dans le SGBD, en court-circuitant Dolibarr.

Ceci se fera en manipulant directement le SGBD, via ses fonctionnalités d'import/export en CSV ou autre.

La 1ère solution est certainement la plus simple, mais nécessite une opération manuelle depuis les menus de Dolibarr et n'est donc pas automatisable.

3. "Dockerisation" : création de Dockerfile pour installation sur un serveur, avec les scripts de lancement et d'importation des données. Différentes options sont possibles : installation de la totalité (Dolibarr + SGBD) dans un conteneur mais il est bien sur préférable de dissocier et d'avoir les deux dans deux conteneurs différents.

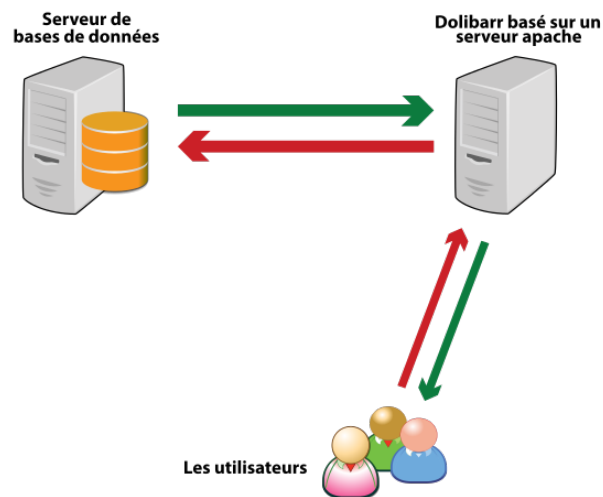
Vous trouverez aussi des images Docker de Dolibarr déjà préparées, il faudra faire des essais et évaluer la meilleure stratégie.

### 3 Description du logiciel Dolibarr

Dolibarr est un logiciel écrit en PHP, et utilise un SGBD distinct pour y stocker les données. Il est construit de façon modulaire : une fois la base installée, des modules complémentaires peuvent s'ajouter via le menu d'installation de l'utilisateur "superadmin".

---

3. [https://fr.wikipedia.org/wiki/Plan\\_de\\_reprise\\_d%27activit%C3%A9](https://fr.wikipedia.org/wiki/Plan_de_reprise_d%27activit%C3%A9)



Dolibarr est fourni en plusieurs versions :

- la version "source" : un fichier archive contenant l'ensemble des fichiers php à copier sur un serveur déjà doté du moteur PHP et du SGBD (cas classique de l'hébergement mutualisé). Dans le cas d'un serveur "nu", ceci implique d'installer les autres composants logiciels.
- la version "Dolideb", qui est un paquet Debian prêt à installer sur une machine Debian/Ubuntu, et comprenant (en théorie... <sup>4</sup>)l'ensemble de ce qui est nécessaire.

D'origine, il est prévu pour fonctionner avec un serveur Web Apache, mais il est tout à fait possible de le faire fonctionner avec Nginx. De même pour le SGBD, il peut fonctionner avec MySQL, MariaDB ou Postgres.

Voir les liens ci-dessous pour plus de détails

- <https://fr.wikipedia.org/wiki/Dolibarr>
- <https://www.dolibarr.org/>
- <https://github.com/Dolibarr/dolibarr/>
- <https://wiki.dolibarr.org/>

## 4 Modalités pratiques

1. Travail en équipe de deux **impérativement**, organisation libre, les deux personnes devant être du même groupe. Pour le groupe avec un nombre d'étudiants impair, il y aura une (une seule!) équipe de 3, à vous de vous organiser.
2. l'un des membres de l'équipe (à définir entre vous) devra jouer le rôle de "chef de projet" et aura pour mission d'assurer le suivi du projet (voir ci-dessous).
3. Tout les fichiers (scripts, documentation, essais divers, ...), seront versionnés sur un dépôt dont le nom sera `sae-dolibarr`, sur le compte de celui qui sera désigné Chef de Projet. (il faudra qu'il ajoute son/ses partenaires comme collaborateur sur ce dépôt). L'organisation en dossiers dans ce dépôt est laissé libre, mais il sera impératif d'avoir dans la racine du projet un fichier `readme.md` ainsi qu'un fichier `suivi.projet.md`. Vous pourrez vous inspirer pour le reste de quelque chose comme ceci :

---

4. En pratique : non

```
sae-dolibarr
|-- docs/
|-- sources/
|-- tests/
|-- data/
|-- sources.md
|-- readme.md
|-- suivi_projet.md
|-- ...
```

Le dossier **data** pouvant contenir les fichiers de données virtuels, le fichier **sources.md** contenant des liens vers des sources que vous avez utilisées, et le fichier le fichier **readme.md** étant le "point d'entrée" de la documentation du projet (et contenant donc des liens, si celle-ci est répartie sur plusieurs fichiers).

4. Suivi du projet : il est attendu que le chef de projet complète **en fin de chaque séance en autonomie** un "journal de bord" de ce qui a été fait sur la séance, avec éventuellement le détails de ce qui reste à faire, les difficultés rencontrées, etc. Vous prendrez comme modèle le fichier suivant :

[https://github.com/skramm/but3\\_rt/blob/main/suivi\\_projet.md](https://github.com/skramm/but3_rt/blob/main/suivi_projet.md)

5. Durée projet : à définir

6. Pour les fonctionnalités à mettre en place dans Dolibarr, on se limitera à une gestion des "Tiers" : une liste de clients/fournisseurs. Il s'agit ici de réaliser un "POC"<sup>5</sup>, pas de faire une vraie mise en production.

Pour les données CSV dans le cadre de ce projet, vous pourrez générer des données virtuelles (Raison sociale, adresse, téléphone, mail, ...) via une IA générative.

## 5 Références à consulter

- <https://all-it-network.com/installer-dolibar/>
- <https://libremaster.com/comment-installer-dolibarr-sur-un-vps-ovh-debi>
- <https://geekrewind.com/setup-dolibarr-erp-crm-on-ubuntu-16-04-17-10-18>

## 6 Annexes

Ici, quelques informations qui peuvent être utiles dans le cadre de ce projet.

### 6.1 Installation d'un paquet téléchargé

En dehors de l'installation de paquets via les dépôts de la distribution (avec **apt**), on peut aussi installer manuellement un paquet **.deb** téléchargé avec l'outil de gestion des paquets **dpkg** : `$ sudo dpkg -i <paquet.deb>`

Pour une présentation complète, voir <https://phoenixnap.com/kb/dpkg-command>

---

5. *Proof Of Concept*

## 6.2 Compléments sur la construction d'une image

Chaque étape de la construction d'une image via un Dockerfile va générer une image intermédiaire, qui va occuper de l'espace disque. Par exemple, ceci va générer 3 images intermédiaires

```
FROM Debian:11
RUN apt-get update
RUN apt-get install -y paquet1
RUN apt-get install -y paquet2
```

(On peut les visualiser avec `$ docker images -a`)

L'avantage, c'est que si une ligne a été réussie, lors de l'essai suivant, Docker va utiliser l'image intermédiaire et ne va pas recommencer depuis le début. Mais dans le cadre de la construction "finale", afin de minimiser l'espace disque, il sera préférable de faire :

```
FROM Debian:11
RUN apt-get update && apt-get install -y paquet1 paquet2
```

Cependant, dans certains cas, il faudra d'abord déterminer quels sont les paquets nécessaires à notre application. Si on fait `RUN apt-get install -y paquet1 paquet2 paquet3 paquet4` et que l'un deux n'existe pas ou que par exemple `paquet4` s'appelle en réalité `paquet4-v2`, alors, lors de l'essai suivant, il faudra re-télécharger les 3 paquets, ce qui fera perdre du temps.

Durant cette phase, il sera préférable de faire une ligne "RUN" par paquet, avant de les regrouper tous ensemble une fois qu'on est sûr que tout est bon<sup>6</sup>.

## 6.3 Installation d'un serveur MySql

Dans le cadre d'une install "test" dans une VM, et pour pouvoir effectuer la procédure d'installation de Dolibarr (via `http://localhost/dolibarr/install`), il est nécessaire de mettre en route d'abord le serveur de SGBD.

Sous Debian, le paquet `MySql` installe en fait `MariaDB`, qui est un *fork* de fonctionnalités similaires.

Une fois le paquet `default-mysql-server` installé, il faut le démarrer avec :

```
$ systemctl start mysqld.
```

Ensuite, l'accès au serveur implique de configurer les accès au préalable, via un programme spécifique. Taper : `$ sudo mariadb-secure-installation` et indiquer "root" comme `pwd`. Pour les questions suivantes, suivez les recommandations proposées ou répondez de façon la plus pertinente.

Tester ensuite : vous devriez pouvoir vous connecter au serveur avec :

```
$ mysql -u root -p'root'
```

Une fois ceci validé, la procédure d'install de Dolibarr doit "passer", en indiquant les bons *credentials* de connexion au SGBD.

Regarder aussi cette page, qui donne des détails la conteneurisation de `MySql` et `MariaDB` :

[https://github.com/skramm/but3\\_rt/tree/main/docker\\_BDD](https://github.com/skramm/but3_rt/tree/main/docker_BDD)

<sup>6</sup>. Penser à supprimer les images intermédiaires créées précédemment!