



ENTERPRISE RESOURCE PLANNING: ODOO

Présenté par : REMACI ZEYNEB YASMINA

ENTREPRISE

- Une entreprise est un groupe de personnes organisé de manière hiérarchique qui utilise des ressources intellectuelles, physiques et financières pour extraire, produire, transformer et distribuer des richesses dans le but d'atteindre des objectifs spécifiques.
- Exemple : Amazon, Volkswagen, AGRO Industrie

ENTREPRISE

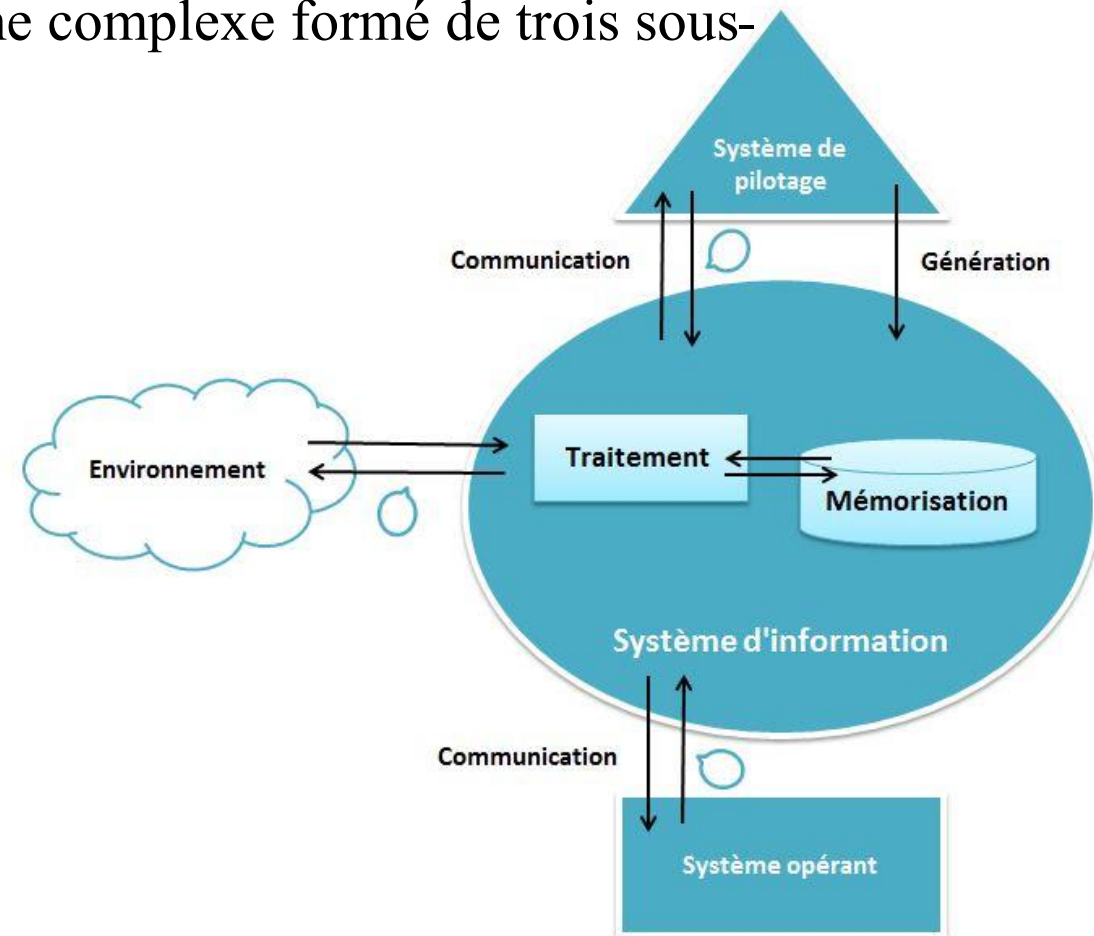
En générale la définition de l'entreprise dépend de l'approche que l'on choisit.

- **Selon l'approche économique:** l'entreprise est une unité de production.
- **Selon l'approche sociologique:** l'entreprise est composée de 3 acteurs principaux (Apporteurs de capitaux, Dirigeants, Salariés) et chaque acteur à leurs objectifs propres.
- **Selon l'approche systémique (L'entreprise est un système) :** L'entreprise est considérée comme un ensemble d'éléments (des moyens humains, matériels, financiers et techniques) en interrelations.

L'ENTREPRISE VUE EN TANT QUE SYSTÈME

L'entreprise peut être considérée comme un système complexe formé de trois sous-systèmes:

1. Le système de décision (ou de pilotage).
2. Le système opérant.
3. Le système d'information.



L'ENTREPRISE VUE EN TANT QUE SYSTÈME

- 2. Le système opérant:** est l'ensemble des processus par lesquels l'information est convertie en action.
- Reçoit les informations émises par le système de pilotage .
 - Se charge de réaliser les tâches qui lui sont confiées.
 - Génère à son tour des informations en direction du système de pilotage.
 - Contrôler les écarts et agir en conséquence.

L'ENTREPRISE VUE EN TANT QUE SYSTÈME

- 1. Le système de décision (ou de pilotage):** finalise l'entreprise en lui assignant ses objectifs (court, moyen et long terme). Il analyse l'environnement et le fonctionnement interne de l'entreprise.

SYSTÈME D'INFORMATION

3. **Le système d'information (SI):** est un ensemble organisé de ressources : matériel, personnel, données, procédures permettant d'acquérir, de traiter, de stocker, de communiquer des informations pour supporter le fonctionnement de l'organisation.

Les fonctions du système d'information:

1. **Acquisition** : Collecter des informations par saisie et/ou consultation.
2. **Mémorisation** : Enregistrer des informations sur des supports.
3. **Traitement** : Transformer des informations par des traitements informatiques ou des traitements manuels.
4. **Communication** : Transmettre d'informations entre différents acteurs ou fonctions.

SYSTÈME D'INFORMATION ET INFORMATIQUE

- Un système informatique est un ensemble de matériels, logiciels, réseaux et procédures permettant d'élaborer, traiter, stocker, ou diffuser des données.
- Le système informatique constitue l'infrastructure technique du système d'information de l'organisation



EXEMPLE: SYSTÈME D'INFORMATION

1. Entreprise : Amazon.com

Composants du Système d'Information :

- 1. Matériel** : Des **centres de données** massifs pour héberger leurs sites web, des **entrepôts automatisés** pour le stockage et la gestion des produits, **des serveurs** pour gérer les transactions en ligne, des **appareils électroniques** (Kindle, Echo, etc.) pour les clients, etc.
- 2. Personnel** : Des **ingénieurs logiciels**, des **spécialistes en logistique**, des experts en **sécurité informatique**, des équipes **de service client**, etc.
- 3. Données** : Une vaste base de données de **produits**, **d'informations sur les clients**, **d'historiques de commandes**, etc.

EXEMPLE: SYSTÈME D'INFORMATION

Un client visite le site web d'Amazon pour acheter un livre. Voici comment le système d'information intervient dans ce processus :

- 1.Acquisition d'informations :** Le client recherche le livre, lit les avis d'autres clients, ajoute le livre à son panier, et fournit ses informations de paiement et de livraison.
- 2.Traitement des informations :** Le système d'information d'Amazon calcule le coût total de la commande, vérifie la disponibilité du livre en stock, et détermine les options de livraison.
- 3.Stockage des informations :** Les détails de la commande, les informations sur le client, et les données sur le produit sont enregistrés dans la base de données d'Amazon.
- 4.Communication des informations :** Le client reçoit une confirmation de commande par e-mail, et le système d'information coordonne l'expédition depuis l'entrepôt le plus proche.

EXEMPLE: SYSTÈME OPÉRANT

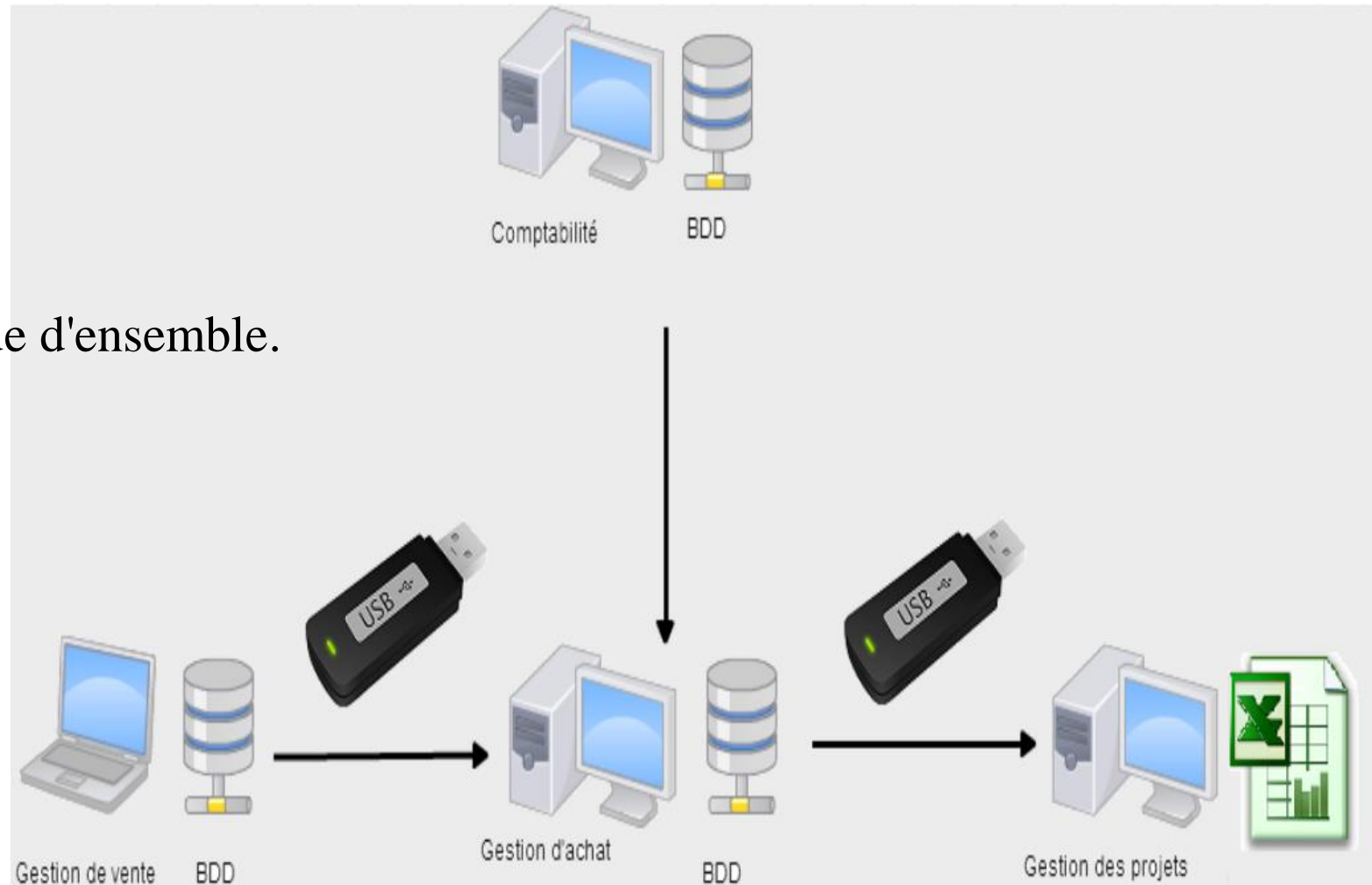
Le système opérant dans le contexte d'Amazon englobe l'ensemble **des opérations quotidiennes** et **les activités pratiques**. Il comprend par exemple les employés qui travaillent dans les entrepôts pour gérer les stocks, préparer et expédier les commandes (livre), gérer la logistique, etc.

EXEMPLE: SYSTÈME DE PILOTAGE

1. Amazon a identifié une forte demande pour les livres de cuisine végétarienne et a décidé de lancer une nouvelle collection de livres de recettes végétariennes en investissant 10 millions de dollars sur les deux prochaines années pour développer cette gamme.

LES PROBLÈMES DES LOGICIELS CLASSIQUE

1. Manque d'intégration.
2. Redondance des données.
3. Complexité.
4. Difficulté à obtenir une vue d'ensemble.
5. Évolutivité limitée.



SOLUTION



QU'EST CE QU'UN ERP?

Le terme ERP est un acronyme qui signifie Enterprise Resources Planning. Solution logicielle qui regroupe en son sein les principales composantes fonctionnelles de l'entreprise.

PGI : Progiciel de Gestion Intégré

ERP : ARCHITECTURE MODULAIRE

1. Chaque fonction de l'entreprise correspond un module indépendant.
2. Ces modules partagent la même base de données.
3. Modules compatibles entre eux.



POINTS FORTS DES ERP

1. Intégration complète.
2. Automatisation des processus.
3. Meilleure prise de décision.
4. Amélioration de la collaboration .

POINTS FAIBLES DES ERP

1. Coût élevé.
2. Complexité de mise en œuvre.
3. Adaptation aux besoins spécifiques.
4. Résistance au changement.
5. Dépendance vis-à-vis du fournisseur.
6. Sécurité des données.

ERP PROPRIÉTAIRE



Les ERP propriétaires

ERP OPENSOURCE



Moqui

Dolibarr ERP/CRM



odoo



ERPNext

Tryton

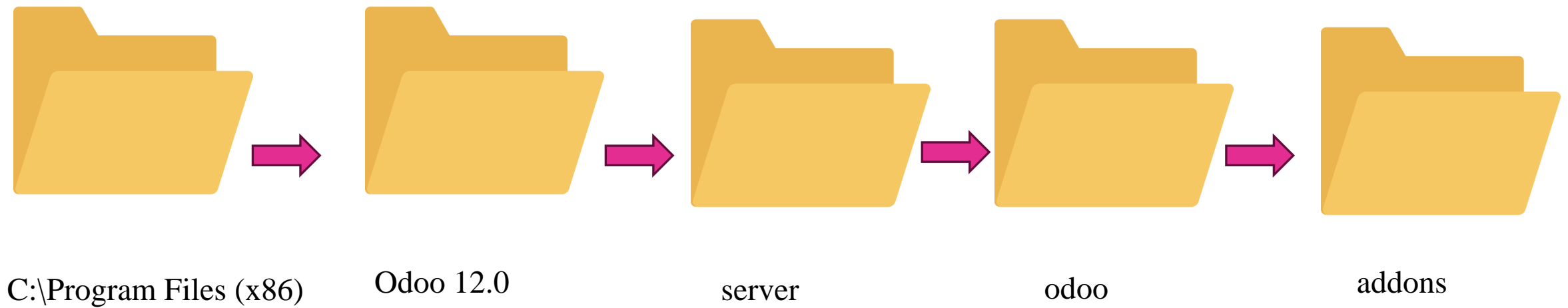
ODOO

- Odoo est un système de gestion d'entreprise open source qui offre une suite complète de modules et d'applications.
- Anciennement connu sous le nom d'OpenERP et Tiny ERP, Odoo propose des solutions pour la comptabilité, la gestion des ventes, la gestion des achats, la gestion des stocks....
- Odoo est disponible en deux versions : une version **communautaire** (Odoo Community) qui est **open source** et **gratuite**, et une version **entreprise** (Odoo Enterprise) qui offre des fonctionnalités supplémentaires et un support technique, mais qui est **payante**.

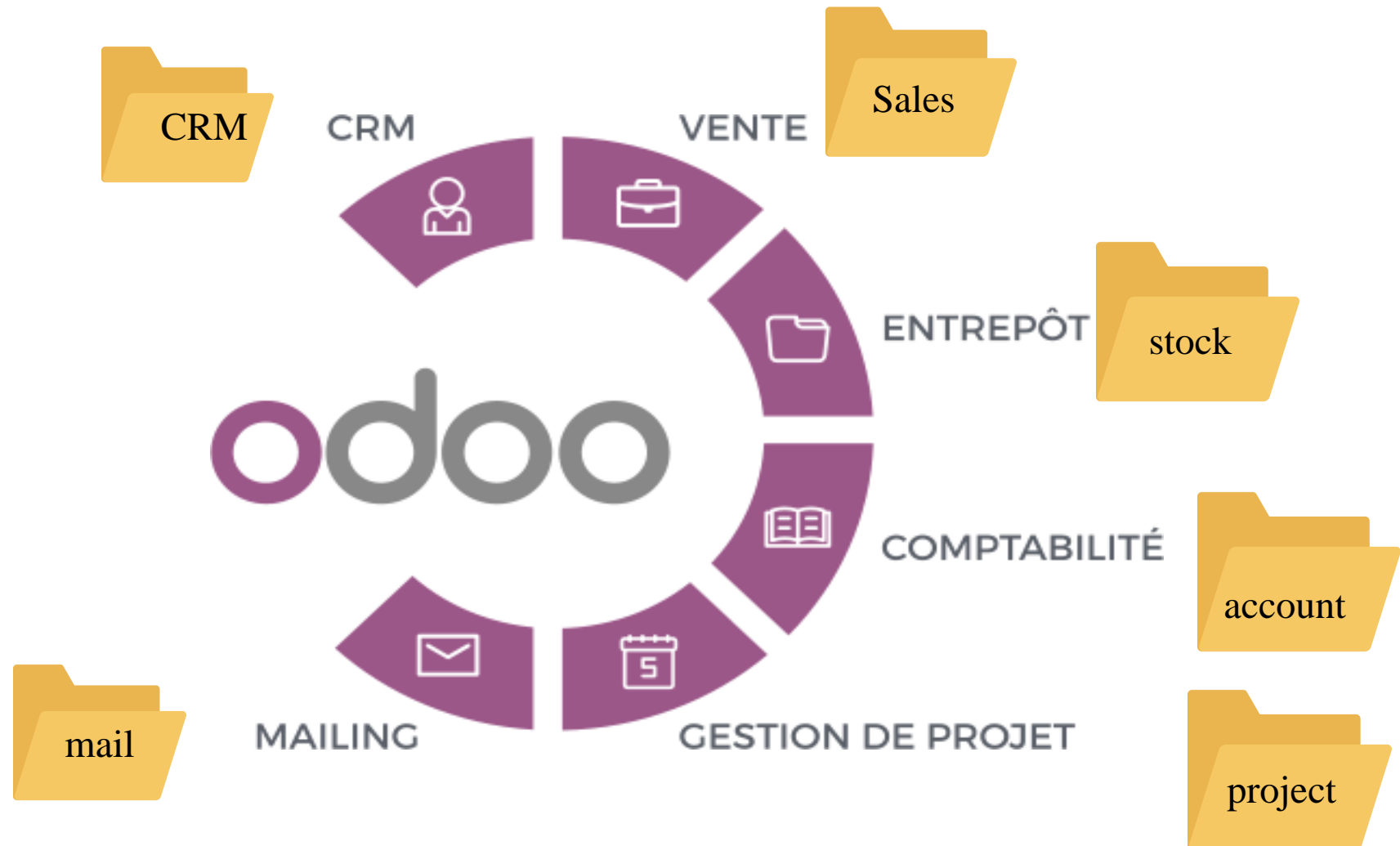
ODOO EN PRATIQUE



ODOO EN PRATIQUE



EXEMPLE



MODULE DE ODOO



Front End

xml

Vs



Back End

python

MODULE DE ODOO

Applications



Gestionnaire des tâches












Fichier Options Affichage

Processus Performance Historique des applications Démarrage Utilisateurs Détails Services

Nom	PID	Description	Statut	Groupe
odoo-server-12.0	24840	odoo-server-12.0	En cours d'exécution	
odoo-server-13.0	4864	odoo-server-13.0	En cours d'exécution	
odoo-server-14.0	4856	odoo-server-14.0	En cours d'exécution	
odoo-server-15.0	4908	odoo-server-15.0	En cours d'exécution	
OneSyncSvc		Hôte de synchronisation	Arrêté	Unist...
OneSyncSvc_5b2e980	15964	Hôte de synchronisation_5b2e980	En cours d'exécution	Unist...
ose64		Office 64 Source Engine	Arrêté	
p2pimsvc		Identity Manager réseau homologue	Arrêté	Local!
p2psvc		Groupe de mise en réseau de pairs	Arrêté	Local!
PcaSvc	1684	Service de l'Assistant Compatibilité des programm...	En cours d'exécution	Local!
perceptionsimulation		Service de simulation de perception Windows	Arrêté	
PerfHost		Hôte de DLL de compteur de performance	Arrêté	
PhoneSvc	1164	Service téléphonique	En cours d'exécution	Local!
PimIndexMaintenanceSvc		Données de contacts	Arrêté	Unist...
PimIndexMaintenanceSvc_5...	15964	Données de contacts_5b2e980	En cours d'exécution	Unist...
pla		Journaux & alertes de performance	Arrêté	Local!
PlugPlay	620	Plug-and-Play	En cours d'exécution	Dcom
PNRPAutoReg		Service de publication des noms d'ordinateurs PN...	Arrêté	Local!
PNRPsvc		Protocole PNRP	Arrêté	Local!
PolicyAgent	3500	Agent de stratégie IPsec	En cours d'exécution	Netw
PostgreSQL_For_Odoo	5468	PostgreSQL_For_Odoo - PostgreSQL Server 12	En cours d'exécution	
Power	620	Alimentation	En cours d'exécution	Don

Moins de détails | Ouvrir les services

MODULE DE ODOO

 data	Répertoire des fichiers XML (enregistrements)
 i18n	Répertoire de traduction (.po)
 models	Répertoire des fichiers Python (Objets)
 report	Répertoire des fichiers XML (Qweb).
 security	Répertoire des fichiers XML et CSV (Droit d'accès)
 static	Déclarer tout les fichier python utiliser par le module.
 tests	Répertoire des fichiers de testes.
 views	Répertoire des fichiers XML (Interfaces).
 wizard	Répertoire des fichiers PY et XML destinés au wizard
 __init__	Déclarer tout les fichier python utiliser par le module.
 __manifest__	Déclaration et description du module.

ODOO.CONF & ODOO.LOG

```
[options]
addons_path = C:\Program Files (x86)\Odoo 12.0\server\odoo\addons
admin_passwd = admin
bin_path = C:\Program Files (x86)\Odoo 12.0\thirdparty
csv_internal_sep = ,
data_dir = C:\Users\h\AppData\Local\OpenERP S.A\Odoo
db_host = localhost
db_maxconn = 64
db_name = False
db_password = odoo12
db_port = 5432
db_sslmode = prefer
db_template = template0
db_user = odoo12
dbfilter =
demo = {}
email_from = False
geoip_database = /usr/share/GeoIP/GeoLite2-City.mmdb
http_enable = True
http_interface =
http_port = 8096
import_partial =
limit_memory_hard = None
limit_memory_soft = None
limit_request = None
limit_time_cpu = None
limit_time_real = None
limit_time_real_cron = None
list_db = True
log_db = False
```

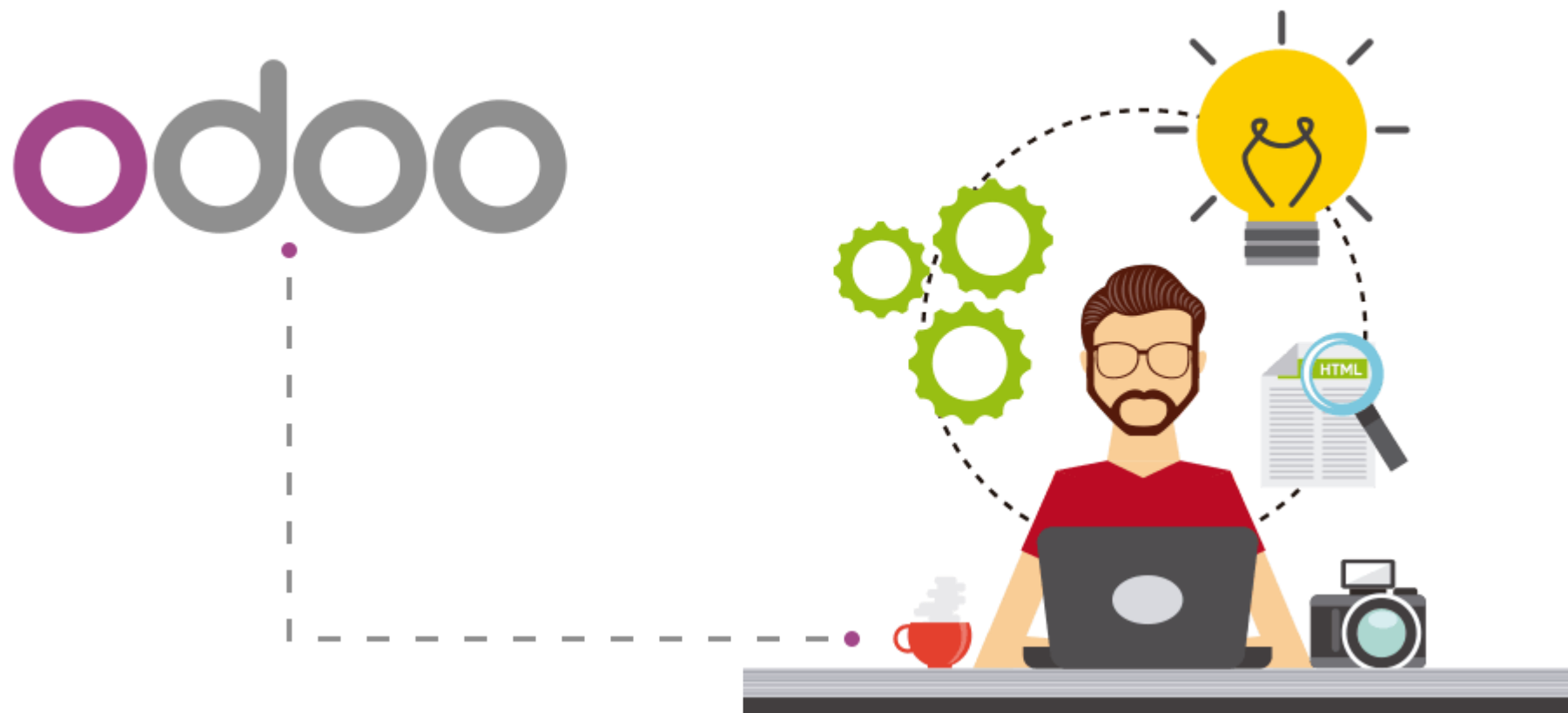
```
Email Queue Manager .
2023-09-30 19:24:24,340 6112 INFO bdd_test odoo.addons.base.models.ir_cron: Starting job `Partner
Autocomplete : Sync with remote DB`.
2023-09-30 19:24:24,358 6112 INFO bdd_test odoo.addons.base.models.ir_cron: Starting job
`Snailmail: process letters queue`.
2023-09-30 19:24:24,374 6112 INFO bdd_test odoo.addons.base.models.ir_cron: Starting job
`Calendar: Event Reminder`.
2023-09-30 19:24:24,390 6112 INFO bdd_test odoo.modules.registry: At least one model cache has
been invalidated, signaling through the database.
2023-09-30 19:40:53,658 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:53] "GET /
HTTP/1.1" 200 - 2 0.008 0.195
2023-09-30 19:40:53,748 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:53] "GET /
web HTTP/1.1" 303 - 1 0.000 0.016
2023-09-30 19:40:55,045 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/login HTTP/1.1" 200 - 175 0.353 0.620
2023-09-30 19:40:55,261 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/content/588-f421656/web.assets_common.0.css HTTP/1.1" 200 - 11 0.148 0.038
2023-09-30 19:40:55,455 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/content/672-4dd6991/web.assets_frontend.0.css HTTP/1.1" 200 - 5 0.004 0.069
2023-09-30 19:40:55,457 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/content/673-4dd6991/web.assets_frontend.js HTTP/1.1" 200 - 5 0.013 0.044
2023-09-30 19:40:55,529 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/content/590-f421656/web.assets_common.js HTTP/1.1" 200 - 5 0.017 0.121
2023-09-30 19:40:55,797 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:55] "GET /web
/webclient/qweb?mods= HTTP/1.1" 200 - 1 0.000 0.008
2023-09-30 19:40:56,017 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:56] "POST /
web/webclient/bootstrap_translations HTTP/1.1" 200 - 1 0.000 0.007
2023-09-30 19:40:56,046 6112 INFO bdd_test werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:56] "GET /web
/binary/company_logo HTTP/1.1" 200 - 2 0.031 0.014
2023-09-30 19:40:56,411 6112 INFO ? werkzeug: 127.0.0.1 - - [30/Sep/2023 19:40:56] "GET /web/static
/src/img/favicon.ico HTTP/1.1" 200 - - - -
```

A complex network diagram with numerous nodes of varying sizes (black, blue, and grey) connected by thin grey lines. Some nodes are highlighted with larger concentric circles. The background is light grey with faint, larger-scale network patterns.












ENTERPRISE RESOURCE PLANNING: ODOO

Presented by: REV. ...

CRÉATION D'UN MODULE ODOO



MODULE DE ODOO

 data	Répertoire des fichiers XML (enregistrements)
 i18n	Répertoire de traduction (.po)
 models	Répertoire des fichiers Python (Objets)
 report	Répertoire des fichiers XML (Qweb).
 security	Répertoire des fichiers XML et CSV (Droit d'accès)
 static	Déclarer tout les fichier python utiliser par le module.
 tests	Répertoire des fichiers de testes.
 views	Répertoire des fichiers XML (Interfaces).
 wizard	Répertoire des fichiers PY et XML destinés au wizard
 __init__	Déclarer tout les fichier python utiliser par le module.
 __manifest__	Déclaration et description du module.

__manifest__.py

name	Le nom du module.
version	La version du module, sur deux chiffres (exp. 1.1 ou 2.3).
description	La description du module y compris la documentation sur l'utilisation du module.
author	L'auteur du module.
website	Le site Web du module.
licence	La licence du module (par défaut: GPL-2).
installable	True ou False. Indique si le module est installable ou non.
category	La catégorie dans laquelle le module va être placée.
application	True ou False. Pour que le module soit identifié comme application. Seul Odoo délivre les certificats qualifiant un module d'application.
data	Liste de fichiers xml contenant les interfaces
...	

__MANIFEST__.PY

```
1  # -*- coding: utf-8 -*-
2  {
3  'name': 'gestion de bibliothèque',
4  'version': '1.0',
5  'category': 'bibliothèque',
6  'author': 'REMACI ZEYENEB',
7  'depends': ['project'],
8  'description': """ce module est destiné pour gérer des LA bibliothèque """,
9  'data' :[],
10 'installable': True,
11 'application': True,
12 'auto_install': False,
13 }
```

__init__.py

```
# -*- coding: utf-8 -*-  
from . import models  
from . import wizard
```

Dossier models

1. Chaque fichier python contient un ou plusieurs objets.
2. Chaque objet est représenté par une classe python, qui hérite de la classe « `models.Model` ».
3. Le nom de l'objet est déclaré dans l'attribut « `_name` ».
4. Les champs peuvent être de plusieurs types (`Char`, `Text`, `Integer`, `Float`, `Boolean`, `Date`, `Selection`, `binary` ...)
5. Chaque champ est personnalisé par des attributs (`string`, `required`, `readonly`, `default`, `compute` ...).
6. Ce dossier doit contenir un fichier ?

```
# -*- coding: utf-8 -*-
from odoo import models, fields

class Auteur(models.Model):
    _name = 'auteur'

    nom=fields.Char(string="Nom")
    prenom=fields.Char(string="Prénom")
    date_naissance=fields.Date(string="Date de naissance")
    nationalite=fields.Char(string="Nationalité")
    sexe= fields.Selection([
        ('homme', 'Homme'),
        ('femme', 'Femme')
    ])

```

L'interface: views

1. Les objets sont affichés dans le navigateur par les vues standards de Odoo.
2. Un objet peut être représenté dans une ou plusieurs vues (form, tree, kanban, search, calendar, graph, gantt).
3. Les vues sont des enregistrements dans le modèle « ir.ui.view ».

```
<?xml version="1.0" encoding="utf-8"?>
<odoo>
  <data>
    <record model="ir.ui.view" id="auteur_form_view">
      <field name="name">auteur.form.view</field>
      <field name="model">auteur</field>
      <field name="arch" type="xml">
        <form>
          <sheet>
            <group>
              <group>
                <field name="nom"/>
                <field name="prenom"/>
                <field name="date_naissance"/>
              </group>
              <group>
                <field name="nationalite"/>
                <field name="sexe"/>
              </group>
            </group>
          </sheet>
        </form>
      </field>
    </record>
  </data>
</odoo>
```

```
<record model="ir.ui.view" id="auteur_list_view">
  <field name="name">auteur.list.view</field>
  <field name="model">auteur</field>
  <field name="arch" type="xml">
    <tree>
      <field name="nom"/>
      <field name="prenom"/>
      <field name="date_naissance"/>
      <field name="nationalite"/>
      <field name="sexe"/>
    </tree>
  </field>
</record>
```

L'interface: views

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <odoo>
3      <data>
4
5          <record model="ir.ui.view" id="auteur_form_view">
6              <field name="name">auteur.form.view</field>
7              <field name="model">auteur</field>
8              <field name="arch" type="xml">
9                  <form>
10                      <sheet>
11                          <group>
12                              <group>
13                                  <field name="nom"/>
14                                  <field name="prenom"/>
15                                  <field name="date_naissance"/>
16                              </group>
17                              <group>
18                                  <field name="nationalite"/>
19                                  <field name="sexe"/>
20                              </group>
21                          </group>
22                      </sheet>
23                  </form>
24              </field>
25          </record>
26
27          <record model="ir.ui.view" id="auteur_list_view">
28              <field name="name">auteur.list.view</field>
29              <field name="model">auteur</field>
30              <field name="arch" type="xml">
31                  <tree>
32                      <field name="nom"/>
33                      <field name="prenom"/>
34                      <field name="date_naissance"/>
35                      <field name="nationalite"/>
36                      <field name="sexe"/>
37                  </tree>
38              </field>
39          </record>
40
41      </data>
42  </odoo>
```

L'interface: Menu et Action

1. Les menus ont une structure hiérarchiques.
2. Le menu qui n'a pas de parent est le « TOP MENU » (généralement le nom du module).
3. La création d'un menu se fait par la balise « menuitem » avec les attributs suivant : (id, name, parent, sequence, action ...)
4. Une action est un événement déclenché suite à un click.
5. Les actions sont des enregistrements dans le modèle « ir.actions.act_window ».
6. Ce modèle possède des champs obligatoire et des champs optionnelles (name, res_model, view_type, view_mode).

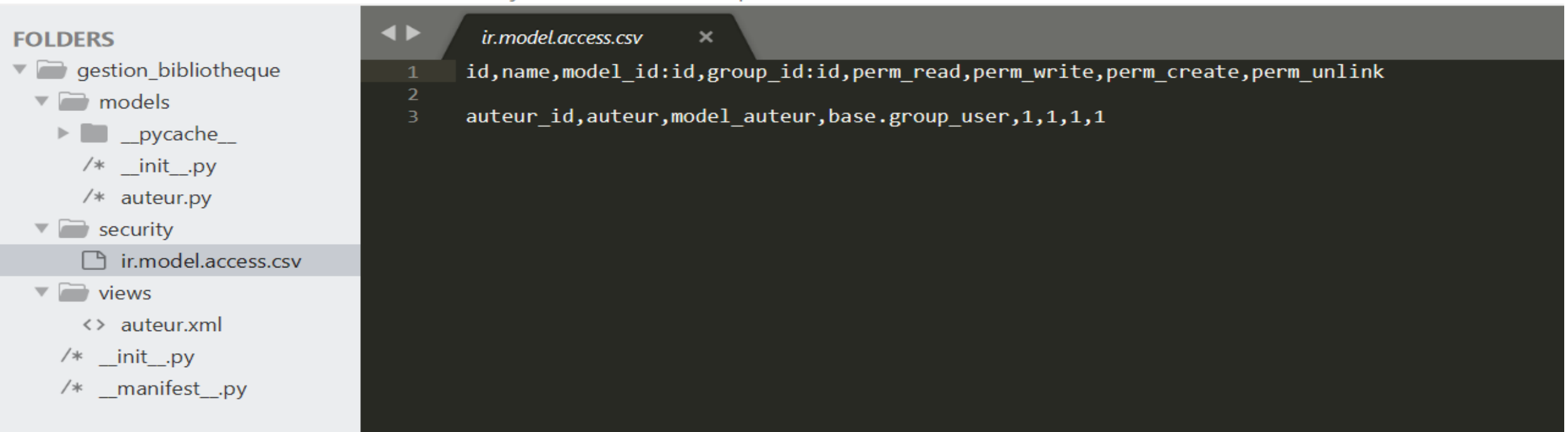
L'interface: Menu et Action

```
<record model="ir.actions.act_window" id="auteur_actions">
  <field name="name">Auteur</field>
  <field name="res_model">auteur</field>
  <field name="view_mode">tree,form</field>
  <field name="view_type">form</field>
  <field name="view_id" ref="auteur_list_view"/>
</record>
```

```
<menuitem id="gestion_bibliotheque" name="Gestion de bibliothèque "/>
<menuitem id="auteur" name="Auteur" parent="gestion_bibliotheque" action="
auteur_actions"/>
```

DROIT D'ACCÈS (1)

1. Pour affecter les propriétés de permission (lecture, écriture, création, suppression) a ces groups on passe par un fichier CSV comme le fichier suivant :



Module gestion de bibliothèque

```
# -*- coding: utf-8 -*-  
from . import models
```

```
< >  _init_.py x  
1 # -*- coding: utf-8 -*-  
2 from . import auteur
```

```
# -*- coding: utf-8 -*-  
{  
    'name': 'gestion de bibliothèque',  
    'version': '1.0',  
    'category': 'bibliothèque',  
    'author': 'REMACI ZEYENEB',  
    'depends': ['project'],  
    'description': """ce module est destiné pour gérer des LA bibliothèque """,  
    'data' : ['views/auteur.xml'],  
    'installable': True,  
    'application': True,  
    'auto_install': False,  
}
```

FOLDERS

- ▼ gestion_bibliotheque
 - ▼ models
 - ▶ __pycache__
 - /* _init_.py
 - /* auteur.py
 - ▼ security
 - ir.model.access.csv
 - ▼ views
 - <> auteur.xml
 - /* _init_.py
 - /* _manifest_.py