

Stock Simple + Inventaire

Module Odoo 17 (simple_stock_inventory)

Auteur : *Anas Slimani / Badr Ezziyati* | Encadrant : *Nidal Lamghari*

École Nationale des Sciences Appliquées de Khouribga

28 décembre 2025



Version : 17.0.1.0.0 **Catégorie :** Inventory **Licence :** LGPL-3

Plan

- 1 Introduction
- 2 Présentation d'Odoo
- 3 Analyse des besoins
- 4 Architecture du module
- 5 Modèles de données
- 6 Vues et interfaces
- 7 Sécurité
- 8 Workflow des mouvements
- 9 Démonstration
- 10 Extraits de code (Python/XML)
- 11 Installation
- 12 Conclusion

Contexte & objectifs

Contexte

Odoo est une plateforme ERP modulaire. Dans plusieurs organisations, la gestion de stock complète peut être trop riche pour un besoin simple (produits, emplacements, mouvements, inventaire).

Objectifs du module

- Proposer une **gestion de stock simplifiée** et maîtrisée
- Couvrir le cycle : **produits** → **mouvements** → **inventaire**
- Garantir la **traçabilité** via le chatter (mail.thread)
- Appliquer une **sécurité claire** (Utilisateur / Manager)

Qu'est-ce qu'Odoo ?

- ERP open-source (apps : CRM, Vente, Achat, Stock, Comptabilité, RH...)
- **Architecture modulaire** : un module = modèles + vues + sécurité + données
- Basé sur :
 - Python (ORM)
 - XML (views / actions / menus)
 - PostgreSQL (données)

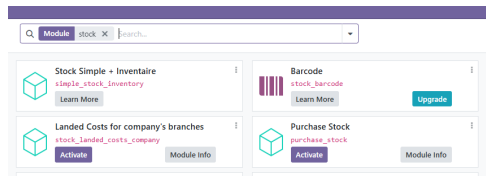


Figure – Module visible dans la liste des Apps

Pourquoi un module personnalisé ?

Adapter l'outil au besoin métier, réduire la complexité et gagner en productivité.

Problématique & besoins fonctionnels

Problématique

Mettre en place une gestion de stock **simple**, centrée sur l'essentiel, tout en conservant : **contrôle, workflow, inventaire, traçabilité**.

Besoins fonctionnels

- ➊ Gestion des produits (prix de revient / vente + stock calculé)
- ➋ Gestion des emplacements (Interne / Fournisseur / Client / Inventaire)
- ➌ Mouvements de stock avec workflow (Brouillon → Validé → Annulé)
- ➍ Vérification stock avant sortie
- ➎ Inventaire physique avec calcul d'écarts
- ➏ Génération automatique des mouvements d'ajustement

Architecture technique (vue d'ensemble)

Composants

- **Modèles Python** : ssi.product, ssi.location, ssi.quant, ssi.move, ssi.inventory, ssi.inventory.line
- **Vues XML** : tree/form + menus + actions
- **Sécurité** : 2 groupes + access rules
- **Suivi** : mail.thread + mail.activity.mixin

Règles importantes

- Séquences automatiques (références)
- Contrainte SQL : unicité (produit, emplacement) dans les quants
- Validations métier (ex : stock insuffisant)

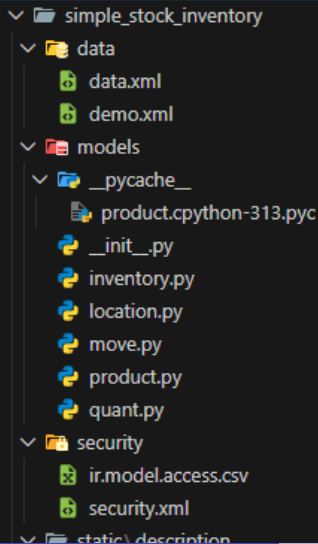
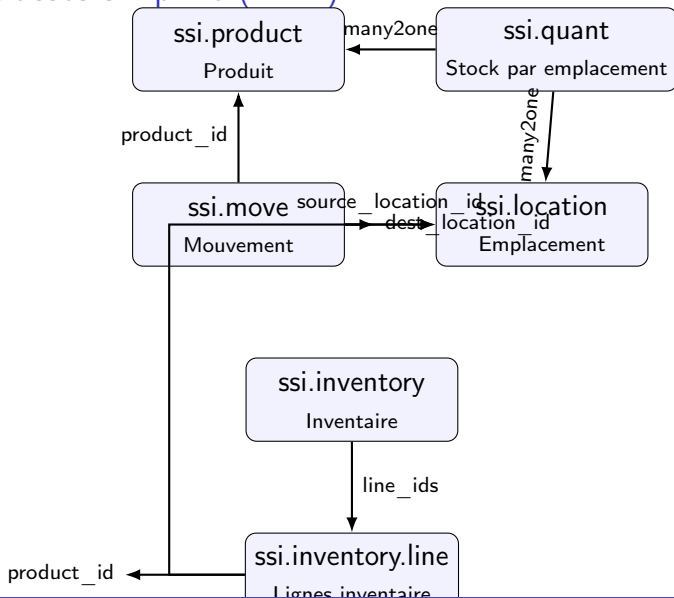


Diagramme de classes simplifié (TikZ)



Modèles : vue synthèse

Modèle	Rôle principal
ssi.product	Définition produit + prix + stock calculé
ssi.location	Emplacements (interne, fournisseur, client, inventaire)
ssi.quant	Quantité disponible par (produit, emplacement)
ssi.move	Entrée / sortie / transfert / ajustement + workflow
ssi.inventory	Session d'inventaire (préparation, validation)
ssi.inventory.line	Quantité théorique vs réelle, calcul d'écart

Champs clés

Champ	Description
name (Char)	Nom du produit
default_code (Char)	Référence interne (séquence)
cost_price (Float)	Prix de revient
sale_price (Float)	Prix de vente
qty_available (Float, computed)	Stock total calculé depuis ssi.quant
active (Boolean)	Actif / Archivé

Suivi

`mail.thread + mail.activity.mixin` : historique, messages, activités.

Types d'emplacements

- **Interne** : stock réel de l'entreprise
- **Fournisseur** : origine d'entrée
- **Client** : destination de sortie
- **Inventaire** : emplacement technique pour ajustement

Champs

Champ	Description
name (Char)	Libellé emplacement
usage (Selection)	internal / supplier / customer / inventory
active (Boolean)	Actif / Archivé

ssi.quant — Quantité par produit/emplacement

Objectif

Centraliser le stock disponible par couple (**product_id**, **location_id**).

Champs

Champ	Description
product_id (Many2one)	Produit concerné
location_id (Many2one)	Emplacement concerné
quantity (Float)	Quantité disponible

Contrainte SQL

Unicité : **un seul quant** par (produit, emplacement).

ssi.move — Mouvement de stock

Workflow

Brouillon → Validé → Annulé

Champs principaux

Champ	Description
name (Char)	Référence (séquence)
move_type (Selection)	entrée / sortie / transfert / ajustement
product_id (Many2one)	Produit
quantity (Float)	Quantité
source_location_id (Many2one)	Source
dest_location_id (Many2one)	Destination
state (Selection)	draft / done / cancel

Règle métier

Avant validation d'une **sortie**, vérification du stock disponible (sinon blocage).

Principe

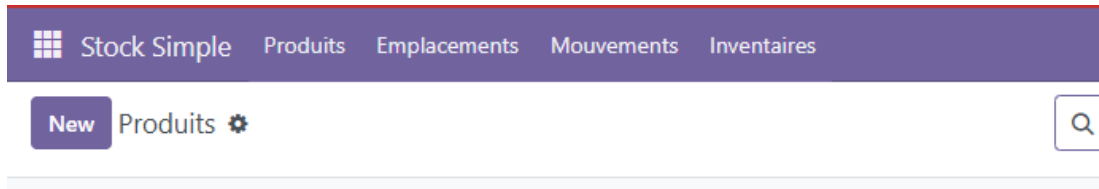
Ajout automatique de tous les produits, comparaison **théorique vs réelle**, calcul de l'écart, puis génération des mouvements d'ajustement.

Modèle	Champs clés
ssi.inventory	name, date, location_id, state, line_ids
ssi.inventory.line	product_id, theoretical_qty, counted_qty, difference_qty

Automatisation

Lors de la validation : création automatique des mouvements (ajustement) vers/depuis l'emplacement **Inventaire**.

Menu principal (Stock Simple)



- Accès direct aux : Produits, Emplacements, Mouvements, Inventaires

Produits : liste et formulaire

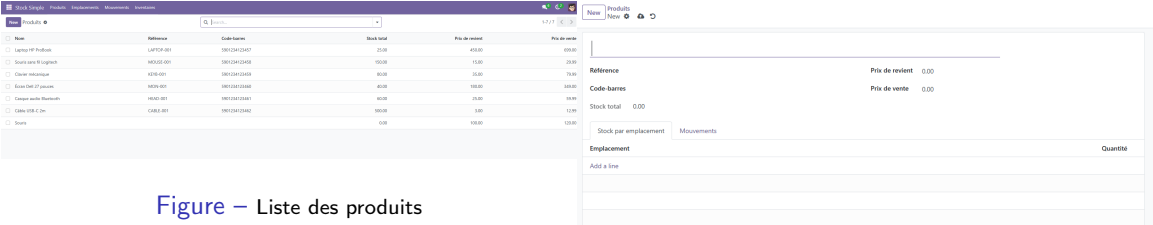


Figure – Liste des produits

Figure – Formulaire produit (prix + stock)

Emplacements : liste et formulaire

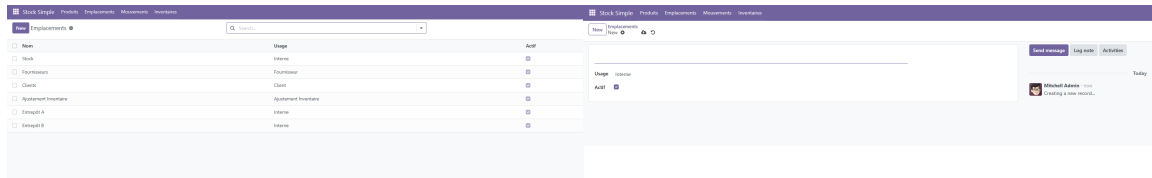


Figure – Liste des emplacements

Figure – Formulaire emplacement (type usage)

Groupes & droits d'accès

Groupes

- **Stock Simple / Utilisateur** : accès lecture/écriture aux opérations courantes
- **Stock Simple / Manager** : droits étendus (validation, configuration, suppression si autorisée)

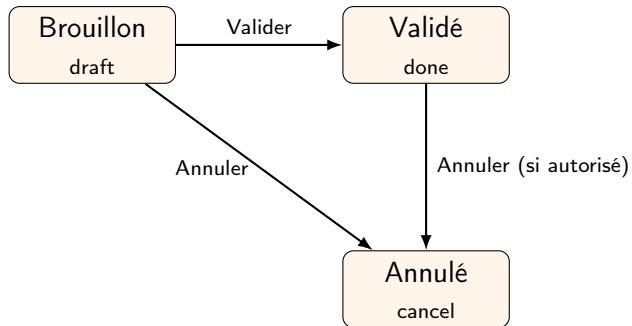
Mécanisme

- **ir.model.access.csv** : droits CRUD par modèle
- (Optionnel) **record rules** : filtrage par société / périmètre

The screenshot shows the Odoo Groups configuration page. At the top, there's a navigation bar with tabs: Settings, General Settings, Users & Companies, Translations, Gamification Tools, and Technical. Below this, there's a 'New' button and a dropdown menu showing 'Groups' and 'Stock Simple / Manager'. The main content area has a form for creating a group. The 'Application' field is set to 'Stock Simple' and the 'Name' field is set to 'Manager'. There's a 'Share Group' checkbox which is currently unchecked. Below the form, there's a tabbed interface with tabs: Users, Inherited, Menus, Views, Access Rights, Record Rules, and Notes. The 'Users' tab is active, showing a table of users. The table has columns: Name, Login, Language, Latest authentication, and Status. One user is listed: Mitchell Admin, with login anasslimani039@gmail.com, language English (US), latest authentication 12/28/2025 17:37:08, and status Confirmed. There's a close button (X) next to the status.

Name	Login	Language	Latest authentication	Status
Mitchell Admin	anasslimani039@gmail.com	English (US)	12/28/2025 17:37:08	Confirmed

Diagramme d'états : Mouvement (TikZ)



Contrôle avant validation

Pour une **sortie**, si stock insuffisant : message d'erreur et état conservé en **brouillon**.

Scénario 1 : création & suivi (chatter)

Étapes

- 1 Créer un produit
- 2 Vérifier le stock calculé
- 3 Observer l'historique (messages / tracking)

Valeur ajoutée

Traçabilité des modifications et communication interne via le chatter.

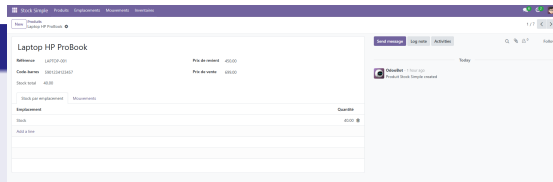


Figure – Exemple de chatter

Scénario 2 : mouvements (brouillon → validé)

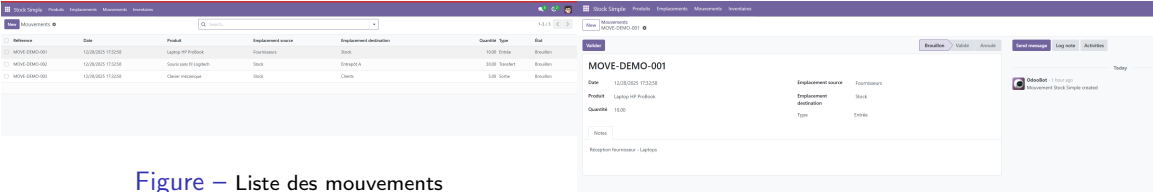


Figure – Liste des mouvements

Figure – Form mouvement (brouillon)

Scénario 2 (suite) : validation du mouvement

New

Mouvements
MOVE-DEMO-001 ⚙

Annuler

Brouillon

Validé

Annulé

Send me

MOVE-DEMO-001

Date

12/28/2025 17:32:58

Produit

Laptop HP ProBook

Quantité

10.00

Emplacement source

Fournisseurs

Emplacement destination

Stock

Type

Entrée

Notes

Réception fournisseur - Laptops

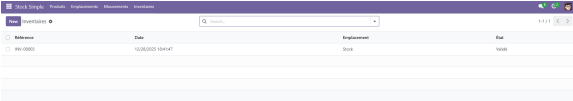
Mit

Ode

Mon

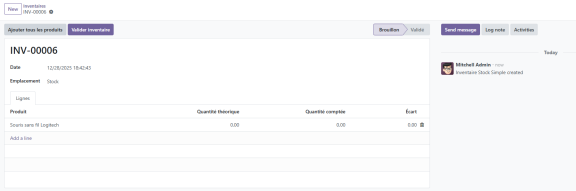
Figure – Mouvement validé (stock mis à jour)

Scénario 3 : inventaire (préparation & lignes)



Reference	Date	Emplacement	Stat
INV-0003	12/08/2025 15:41:47	Stock	Valid

Figure – Liste des inventaires



INVENTAIRE

INV-00006

Date

12/08/2025 18:42:43

Emplacement

Stock

Lignes

Produit

Quantité théorique

Quantité comptée

Écart

Souris sans fil Logitech

0.00

0.00

0.00

Add a line

Send message

Log note

Activities

Today

Michael Admin

Inventaire Stock Simple created

Figure – Inventaire brouillon + lignes

Scénario 3 (suite) : validation inventaire & ajustements

Brouillon

Validé

INV-00003

Date

12/28/2025 18:41:47

Emplacement

Stock

Lignes

Produit	Quantité théorique	Quantité comptée	Écart
Laptop HP ProBook	0.00	5.00	5.00

Figure – Inventaire validé (écarts traités)

Extrait Python : validation d'une sortie (exemple)

Idée

Empêcher la validation si la quantité demandée dépasse le stock disponible sur l'emplacement source.

Listing 1 – Exemple simplifié (ssi.move)

```
1  def action_validate(self):
2      for move in self:
3          if move.move_type == 'out':
4              available = move.product_id.qty_available
5              if move.quantity > available:
6                  msg = "Stock insuffisant: disponible=%s, demande
                          =%s"
7                  raise ValidationError(msg % (available, move.
                          quantity))
8          move.state = 'done'
9          move._apply_quant_update()
```


Extrait XML : vue formulaire (exemple)

Listing 2 – Exemple simplifié d'une vue produit

```
1 <odoo>
2   <record id="ssi_product_form_view" model="ir.ui.view">
3     <field name="name">ssi.product.form</field>
4     <field name="model">ssi.product</field>
5     <field name="arch" type="xml">
6       <form string="Produit">
7         <sheet>
8           <group>
9             <field name="name"/>
10            <field name="default_code"/>
11          </group>
12          <group>
13            <field name="cost_price"/>
14            <field name="sale_price"/>
15            <field name="qty_available" readonly="1"/>
16          </group>
17        </sheet>
18      </form>
19    </field>
20  </record>
21 </odoo>
```

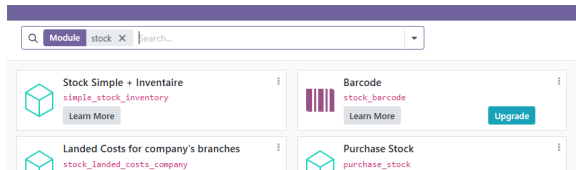
Installation du module

Prérequis

- Odoo 17 installé (environnement dev)
- Accès administrateur pour installer / mettre à jour Apps

Étapes

- 1 Copier `simple_stock_inventory/` dans le dossier addons
- 2 Redémarrer Odoo
- 3 Mode développeur → **Update Apps List**
- 4 Apps → rechercher **Stock Simple** → Installer



Bilan & perspectives

Bilan

- Module fonctionnel couvrant : Produits, Emplacements, Mouvements, Inventaires
- Workflow + contrôles métier (stock insuffisant)
- Traçabilité via mail.thread (chatter)
- Sécurité structurée (Utilisateur / Manager)

Perspectives

- Ajout d'un reporting (mouvements par période / valorisation)
- Support multi-entrepôts / multi-sociétés
- Import/export avancé + QR/Barcode (optionnel)