

R5.D.04 : Développement pour progiciels

Projet : Réalisation d'un module personnalisé lié avec Wordpress

Cyberware



VINET LATRILLE Jules

TD D Alt

Semestre 5

15 / 02 / 2026

TABLE DES MATIERES

MISE EN PLACE ET ENVIRONNEMENT 3

GROUPE DE SECURITE ERREUR ! SIGNET NON DEFINI.

IL Y A AUX MOINS DEUX GROUPE DE SECURITE.....**ERREUR ! SIGNET NON DEFINI.**

DES DROITS D'ACCES DIFFERENTS AUX MODELES ONT ETE ACCORDES AUX GROUPE DE SECURITE . **ERREUR ! SIGNET NON DEFINI.**

MODELES..... ERREUR ! SIGNET NON DEFINI.

IL Y A PLUSIEURS MODELES.**ERREUR ! SIGNET NON DEFINI.**

IL Y A DANS AU MOINS UN MODELE UN CHAMP TEXTUEL, UN CHAMP NUMERIQUE, UN CHAMP DATE, UN CHAMP IMAGE, UN CHAMP SELECT, UN CHAMP CALCULE ET UN CHAMP RELATED.**ERREUR ! SIGNET NON DEFINI.**

IL Y A AU MOINS DANS UN MODELE UNE METHODE QUI CONTROLE DES DONNEES.....**ERREUR ! SIGNET NON DEFINI.**

IL EXISTE AU MOINS UNE RELATION UN A PLUSIEURS ET UNE RELATION PLUSIEURS A PLUSIEURS. **ERREUR ! SIGNET NON DEFINI.**

Relation "Un à Plusieurs".....**Erreur ! Signet non défini.**

Relation "Plusieurs à Plusieurs"**Erreur ! Signet non défini.**

IL EXISTE DES DONNEES DE DEMO AU FORMAT XML. CERTAINS ENREGISTREMENTS SONT LIES PAR LES RELATIONS D'AUTRES NON.**ERREUR ! SIGNET NON DEFINI.**

UN DES MODELES ETEND UN MODELE D'ODOO.**ERREUR ! SIGNET NON DEFINI.**

VUES..... ERREUR ! SIGNET NON DEFINI.

IL EXISTE UNE VUE LISTE PERSONNALISEE.**ERREUR ! SIGNET NON DEFINI.**

IL EXISTE UNE VUE FORMULAIRE PERSONNALISEE.**ERREUR ! SIGNET NON DEFINI.**

IL EXISTE UN CHAMP RECHERCHE PERSONNALISE.**ERREUR ! SIGNET NON DEFINI.**

IL EXISTE UNE VUE CALENDRIER OU UNE VUE KANBAN.....**ERREUR ! SIGNET NON DEFINI.**

UNE PAGE WEB EST DISPONIBLE.**ERREUR ! SIGNET NON DEFINI.**

DANS UNE DES VUES ON TROUVE AU MOINS UN EXEMPLAIRE DE TOUS LES CHAMPS EXIGES DANS LES CONSIGNES SUR LES MODELES.....**ERREUR ! SIGNET NON DEFINI.**

UN BOUTON PERMET D'EXECUTER LA METHODE QUI CONTROLE LES DONNEES**ERREUR ! SIGNET NON DEFINI.**

L'HERITAGE DU MODELE ETENDU EST UTILISE DANS UNE VUE (AFFICHAGE D'UN CHAMP PARENT OU MODIFICATION D'UNE DES VUES DU MODELE PARENT)**ERREUR ! SIGNET NON DEFINI.**

UN OU PLUSIEURS CHAMPS NE SONT VISIBLES QUE POUR UN GROUPE D'UTILISATEUR OU EN FONCTION DE LA VALEUR D'UN AUTRE CHAMP.**ERREUR ! SIGNET NON DEFINI.**

Table des illustrations :

Figure 1 : Remplissage du formulaire de Base de Données 4

Figure 2 : Sélectionner la vue liste personnalisée **Erreur ! Signet non défini.**

Figure 3 : Localisation des filtres de recherche personnalisé... **Erreur ! Signet non défini.**

Figure 4 : Accéder a la vue Kanban..... **Erreur ! Signet non défini.**

Figure 5 : Localisation bouton Site Web..... **Erreur ! Signet non défini.**

Figure 6 : Consulter la fiche d'un client..... **Erreur ! Signet non défini.**

Figure 7 : Bouton pour vérifier le dossier d'un client..... **Erreur ! Signet non défini.**

Contexte

Bonjour monsieur. J'ai perdu l'intégralité de mon travail le 14/02/2026 à cause d'une mise à jour Windows qui a corrompu mon OS Arch qui contenait mon travail. J'ai du donc reprendre tout mon travail (qui était en partie stocké sur GitHub), même si tout ce que j'avais fait entre temps est perdu. Normalement, tout est de nouveau fonctionnel, mais ce rapport ne risque pas d'être aussi complet que celui que j'avais fait précédemment étant donné du manque de temps.

Je vous prie de m'en excuser.

Mise en place et Environnement

Pour lancer ce projet : Une fois le dossier du projet extrait (ce que vous avez fait étant donné que vous lisez ce rapport), ouvrez un terminal dans le dossier contenant le fichier `docker-compose.yml` ainsi que le dossier `addons`. Lancez la commande **`docker-compose up --build`**, qui va permettre de démarrer les conteneurs Docker permettant l'utilisation de l'application.

Une fois cela fait, **attendez quelques secondes** le temps que les conteneurs démarrent correctement, et rendez vous sur le lien <http://localhost:8069>.

Une fois dessus, créez la base de données en remplissant les champs comme indiqué ci-dessous :



Master Password	<input type="password" value="admin"/>	
Database Name	<input type="text" value="odoo18"/>	
Email	<input type="text" value="admin"/>	
Password	<input type="password" value="admin"/>	
Phone Number	<input type="text" value="0102030405"/>	
Language	<input type="text" value="French / Français"/>	
Country	<input type="text" value="France"/>	
Demo Data	<input checked="" type="checkbox"/>	
<div>Create database or restore a database</div>		

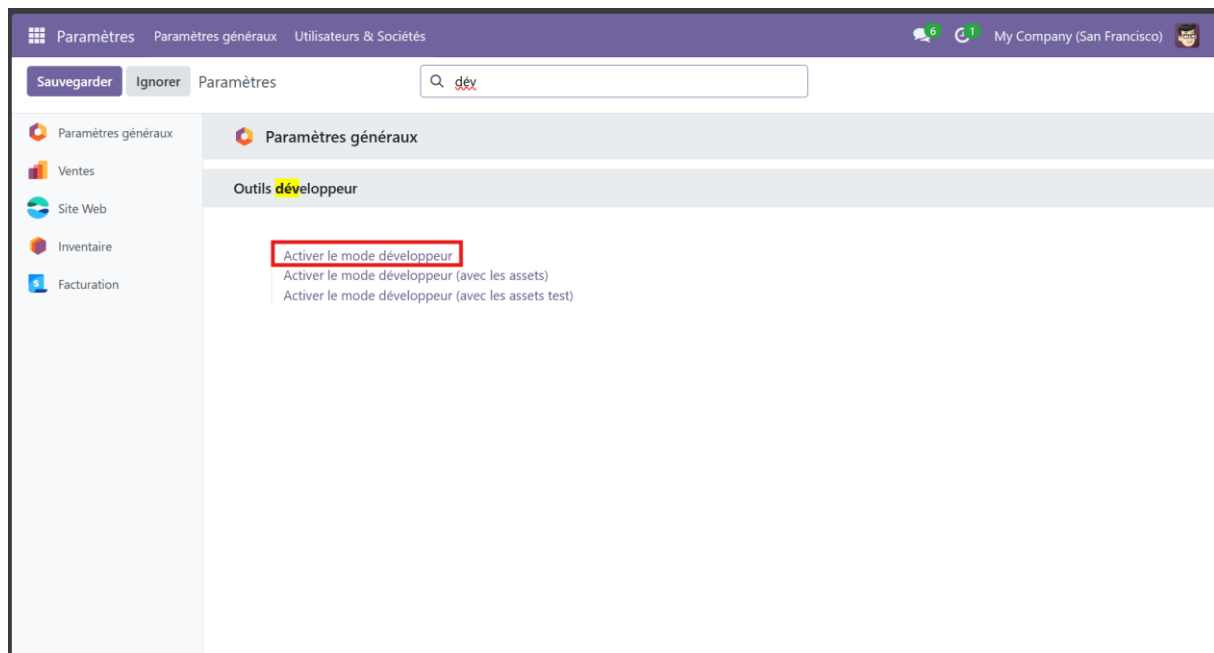
Figure 1 : Remplissage du formulaire de Base de Données

Une fois la Base de données créée, connectez-vous à votre compte avec comme utilisateur « admin » et mot de passe « admin ».

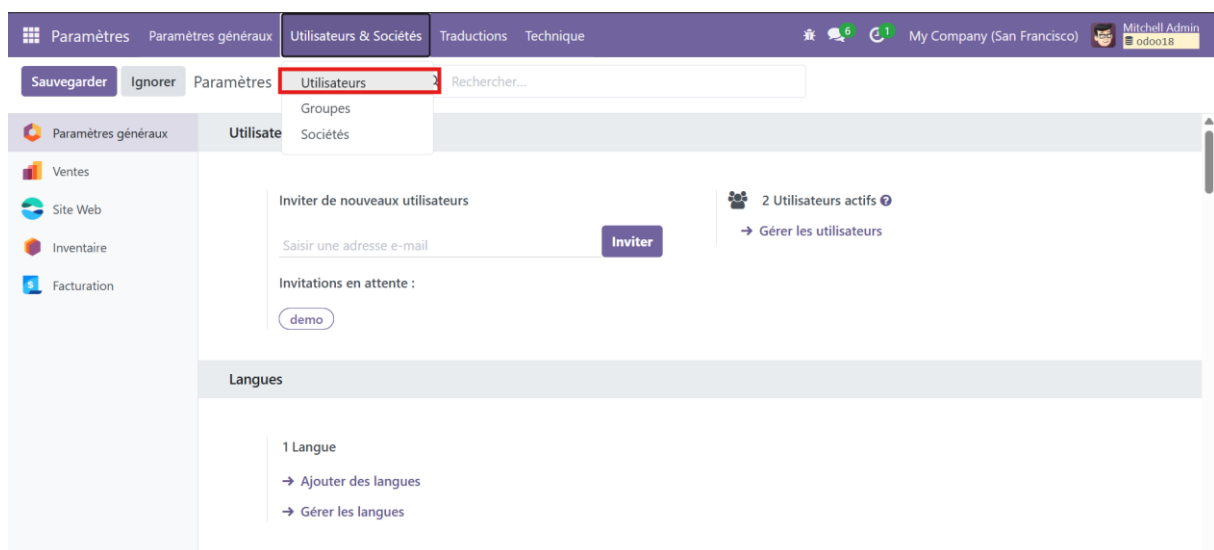
Enfin, activez les applications nécessaires au bon fonctionnement du module : Ventes, Inventaire, Site Web, Contacts, et évidemment Cyberware. (c'est très important pour les droits de l'utilisateur qui va nous permettre d'accéder à la Base de Données).

Créer le compte API

Pour avoir accès à tout les paramètres, pensez bien à activer le mode développeur :



Une fois fait, créer l'utilisateur api en allant dans utilisateurs & Société -> Utilisateurs



Ensuite, cliquer sur nouveau et entrez les informations de celui-ci (appelez le par exemple api@cyberware.local).

Nouveau Utilisateurs
api@cyberware.local

Groupes 21 Droits d'accès 459 Règles sur les enregistrements 64

Jamais connecté Confirmer

Nom ?
api@cyberware.local

Adresse e-mail ?
api@cyberware.local

Droits d'accès Préférences

MULTI-SOCIÉTÉS

Sociétés autorisées ? My Company (San Francisco) X

Société par défaut ? My Company (San Francisco)

USER TYPE

Types d'utilisateur ? ☒ Utilisateur interne
☐ Portail
☐ Publique

localhost:8069/odoo/users/new?debug=1#

Veillez à lui donner les droits de vision sur les données et fonctionnalités du module :

Tableau de bord ? Administration ?

OTHER

Réponses enregistrées ? Administrateur des réponses enregistrées

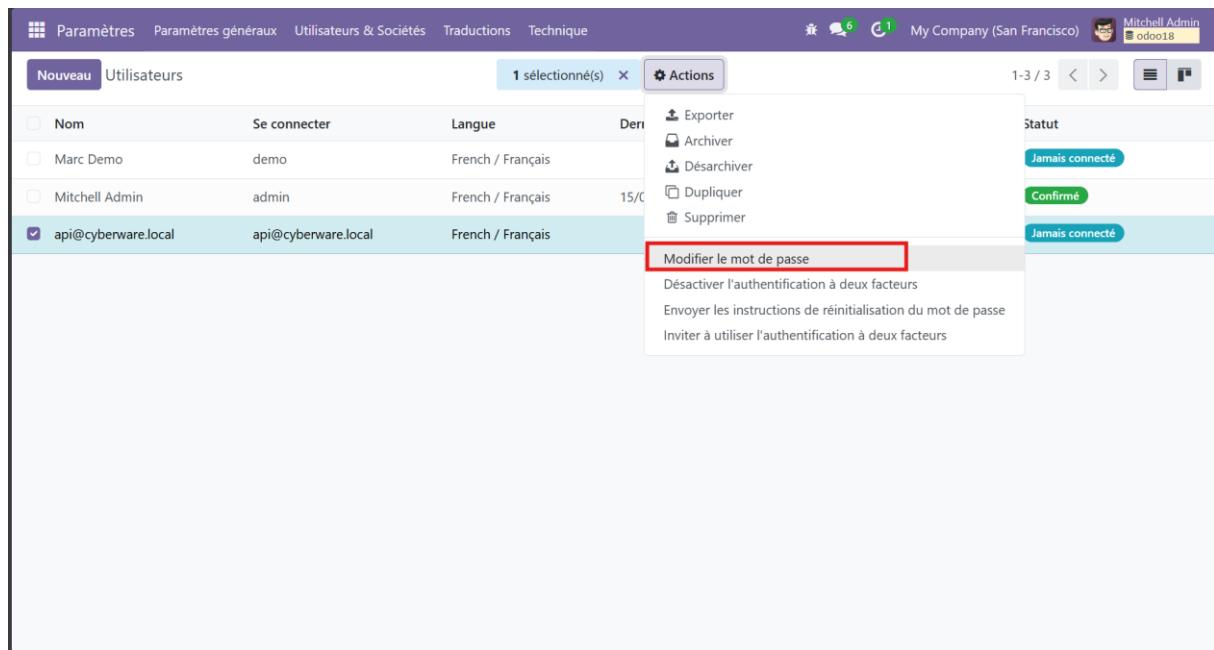
CYBERWARE

Charcudoc (Ripperdoc) ? ☒ Client Cyberware ? ☒

TECHNICAL

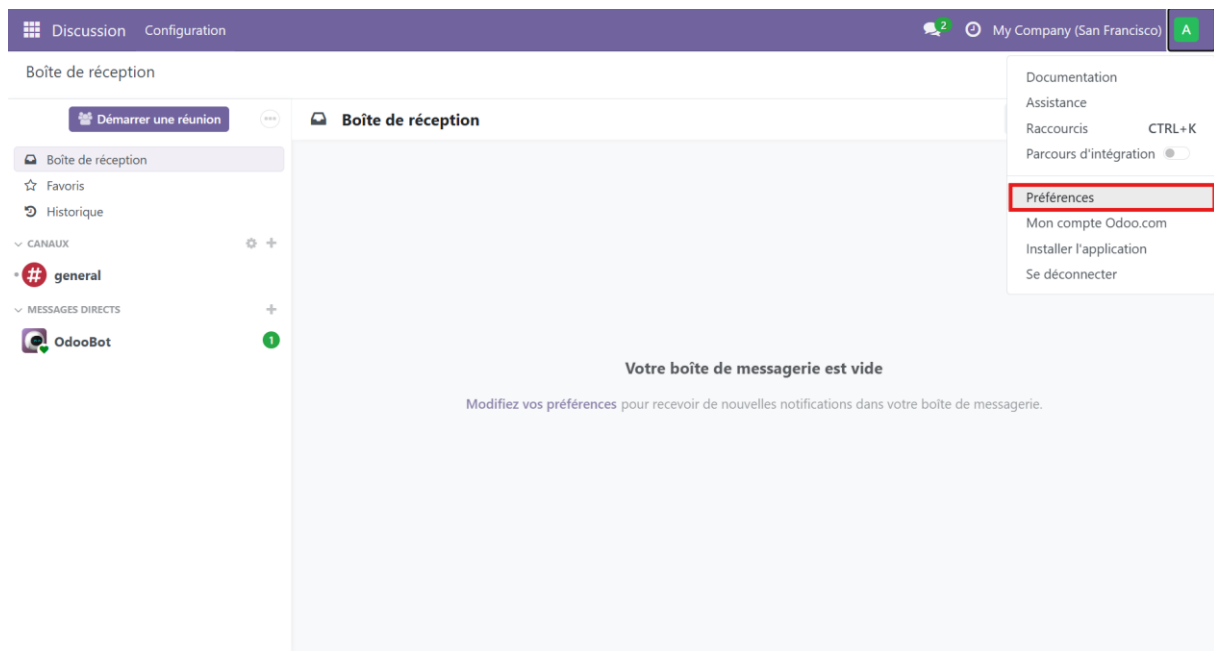
Une alerte peut être mise sur un contact (compte) ? <input type="checkbox"/>	Un avertissement peut être défini sur un contact (stock) ? <input type="checkbox"/>
Un avertissement peut être défini sur un produit ou un client (ventes) ? <input type="checkbox"/>	Accès à la fonctionnalité d'export ? <input checked="" type="checkbox"/>
Autoriser la gestion d'arrondi des paiements en espèces ? <input type="checkbox"/>	Comptabilité analytique ? <input type="checkbox"/>
Listes de prix de base ? <input type="checkbox"/>	Adresse de livraison ? <input type="checkbox"/>
Remise sur les lignes ? <input type="checkbox"/>	Afficher les lots & numéros de série sur les bons de livraisons ? <input type="checkbox"/>
Affichage du numéro de série et de lot sur les factures ? <input type="checkbox"/>	Verrouiller les ventes confirmées ? <input type="checkbox"/>
Éditeur de modèle d'e-mail ? <input type="checkbox"/>	Gérer différents propriétaires de stock ? <input type="checkbox"/>

Ensuite, revenez sur l'interface des utilisateurs, sélectionnez celui précédemment créé, et créez lui un mot de passe (par exemple : admin) :



Une fois fait, déconnectez-vous de votre utilisateur actuel pour vous connecter a celui que vous venez de créer. Nous allons lui créer une clé API.

Allez dans préférences :



Puis sécurité du compte :

Changer mes préférences

api@cyberware.local

PréférencesSécurité du comptePériphériques

GESTION DU MOT DE PASSE

Modifier le mot de passe

AUTHENTIFICATION À DEUX FACTEURS

L'authentification à deux facteurs ("2FA") est un système de double authentification. La première se fait avec votre mot de passe et la deuxième avec un code que vous obtenez depuis une application mobile dédiée. Les plus populaires sont Authy, Google Authenticator ou Microsoft Authenticator. En savoir plus

AUTRES PÉRIPHÉRIQUES

Se déconnecter de tous les appareils

EnregistrerAnnuler

Puis descendez tout en bas pour créer une nouvelle clé API :

New API Key

Nommer votre clé

Saisissez une description et la finalité de la clé.

Wordpress

Il est très important que cette description soit claire et complet, **ce sera le seul moyen d' identifier la clé une fois créée.**

Indiquer une durée de validité pour la clé

1 Month

La clé sera supprimée à l'issue de cette période.

Générer une cléAnnuler

GARDEZ LA BIEN, elle nous sera utile plus tard.

Nous allons passer ensuite a l'installation de Wordpress. Rendez vous sur <http://localhost:8000/> (redirigé vers <http://localhost:8000/wp-admin/install.php>). Entrez les informations nécessaire à la création de votre site Wordpress. ATTENTION, les informations doivent concorder avec celles de votre utilisateur API Odoo.

Informations nécessaires

Veuillez renseigner les informations suivantes. Ne vous inquiétez pas, vous pourrez les modifier plus tard.

Titre du site

Identifiant

Les identifiants ne peuvent utiliser que des caractères alphanumériques, des espaces, des tirets bas ("_"), des traits d'union ("-"), des points et le symbole @.

Mot de passe [Cacher](#)

Très faible

Important : Vous aurez besoin de ce mot de passe pour vous connecter. Pensez à le stocker dans un lieu sûr.

Confirmer le mot de passe ☒ Confirmer l'utilisation du mot de passe faible

Votre e-mail

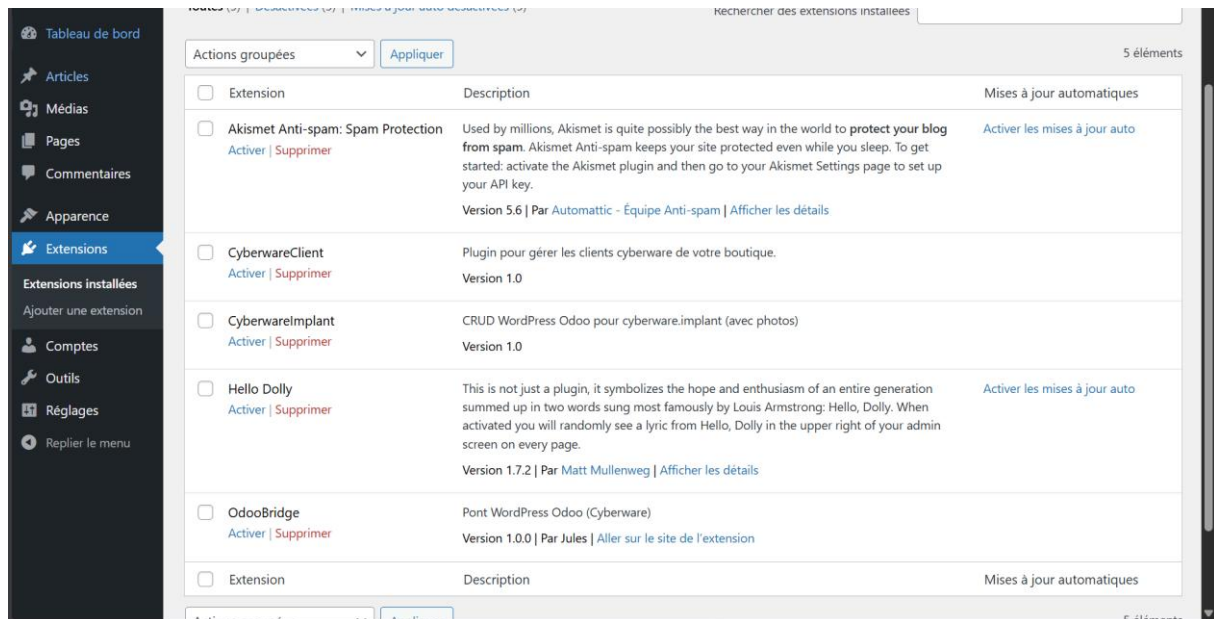
Vérifiez bien cette adresse e-mail avant de continuer.

Visibilité par les moteurs de recherche ☐ Demander aux moteurs de recherche de ne pas indexer ce site
Certains moteurs de recherche peuvent décider de l'indexer malgré tout.

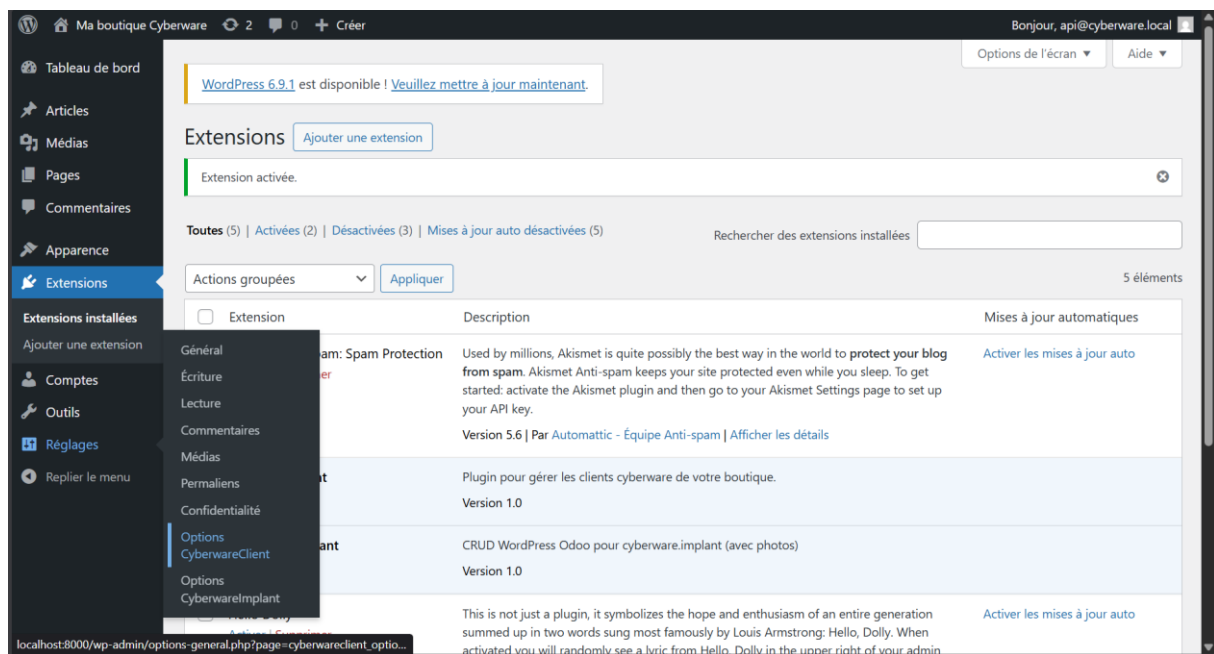
[Installer WordPress](#)

Une fois tout créé, connectez vous.

Allez dans extension, et activez les deux plugins (il font concrètement la même chose, donc au choix activez cyberwareImplant qui contient les 25 données conforme a la consigne).



Une fois activés, allez dans réglages -> OptionsCyberware[...]



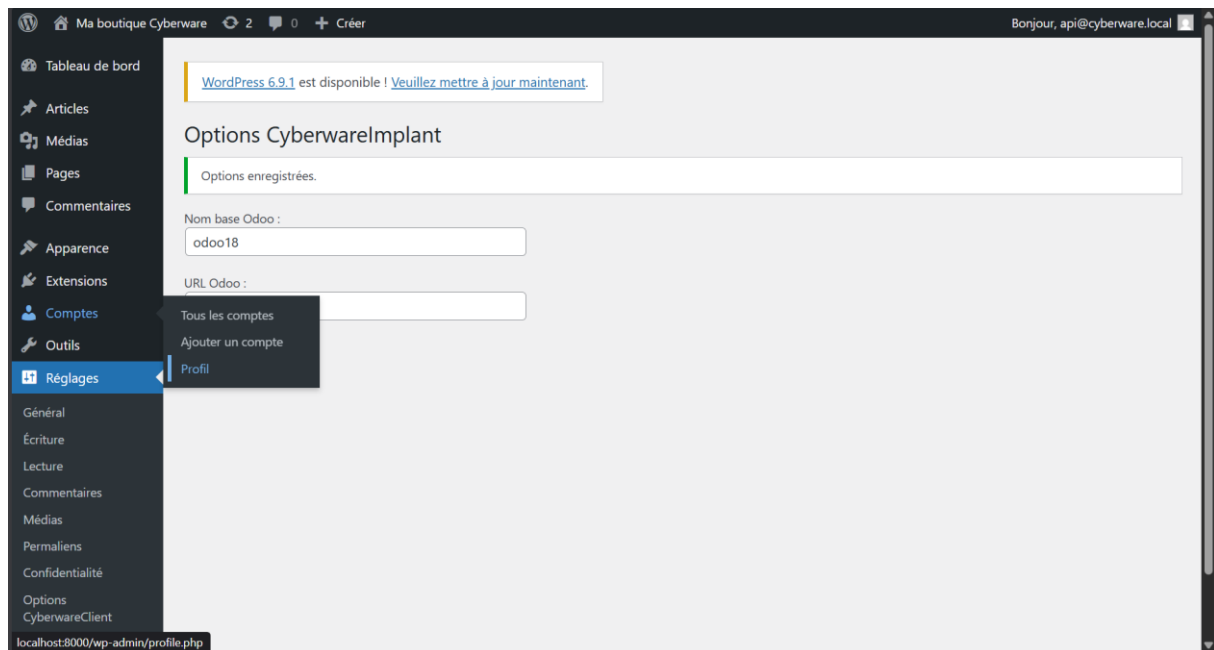
Renseigner les deux champs avec les informations ci dessous :

Nom base Odoo : odoo18 (comme fait lors de l'initialisation d'Odoo)

URL Odoo : <http://web:8069> (le port sur lequel tourne Odoo)

Cliquez sur enregistrer.

Ensuite allez dans l'espace de votre compte



Descendez tout en bas jusqu'à Cyberware[...] où l'on vous demande une api

Entrez y la clé que vous avez généré précédemment , et cliquez sur « Mettre à jour le profil ».

une fois cela fait, rendez vous sur <http://localhost:8000/>, et constatez que deux nouveau menus sont apparus



Vous pouvez ainsi modifier les données depuis ces interfaces (donc l'ergonomie laisse a désirer) :

Créer via le menu déroulant :

Cyberware Implants

Total : 27

recherche nom

Tous types

Toutes raretés

Filtrer

Ajouter un implant

Clique pour ouvrir

Nom

Kiroshi Optics Mk.3

Type

Optique

Rareté

Commun

Prix (€)

0

Coût essence

0

Emplacement

tete / bras / ...

Photo

Choisir un fichier

Aucu...hoisi

PNG/JPG/WebP • max 2MB

Description

...

Actif

☒

Créer

Reset

Modifier via le bouton modifier en bout de tableau (caché par la mise en page) :

NOM	TYPE	RARETÉ	PRIX	ESSENCE	EMPL.	ACTIONS
TEST —	Optique	Commun	10 €	0	—	Modifier

Supprimer depuis l’interface de modification :

Zetatech		
Mechatronic Core Zetatech	Neural	Commun
Pain Editor Zetatech	Armure	Légendaire
Shock-N-Awe Arasaka	Armure	Épique
Cataresist Biotechnica	Organe interne	Commun
Reflex Tuner Zetatech	Neural	Rare
Adrenaline Booster Militech	Organe interne	Rare
Heal-On-Kill	Organe	Commun

Rareté

Commun

Prix (€)

10

Coût essence

0

Emplacement

Photo (remplacer)

Choisir un fichier

Aucun fichier choisi

Description

OUI

Actif

Enregistrer

Supprimer

Exercice 19

Plugin WordPress autonome et indépendant de OdooBridge

Le plugin contient sa propre copie de la librairie XML-RPC (Ripcord). Vous pouvez le voir dans l'arborescence : `php/wp-content/plugins/CyberwareClients/ripcord/`

Il charge donc sa propre instance de ripcord (fichier `cyberwareImplants/OdooPrimitive.php`) :

```
if (!class_exists('ripcord')) {
    require_once __DIR__ . '/ripcord/ripcord.php';
}
```

Fonctions et variables préfixées avec le nom du plugin

Pour ne pas empiéter sur les autres plugins, celui-ci contient des fonctions avec des noms qui lui sont propres, comme par exemple : `cyberwareimplant_odoo_connect`.

Connexion à un modèle Odoo via API XML-RPC

Le fichier `php/wp-content/plugins/CyberwareClients/OdooPrimitive.php` contient la classe qui gère la communication brute avec cette ligne : `$common = ripcord::client($url . "/xmlrpc/2/common");` pour l'authentification et cette ligne : `return ripcord::client($url . "/xmlrpc/2/object");` pour la récupération des objets.

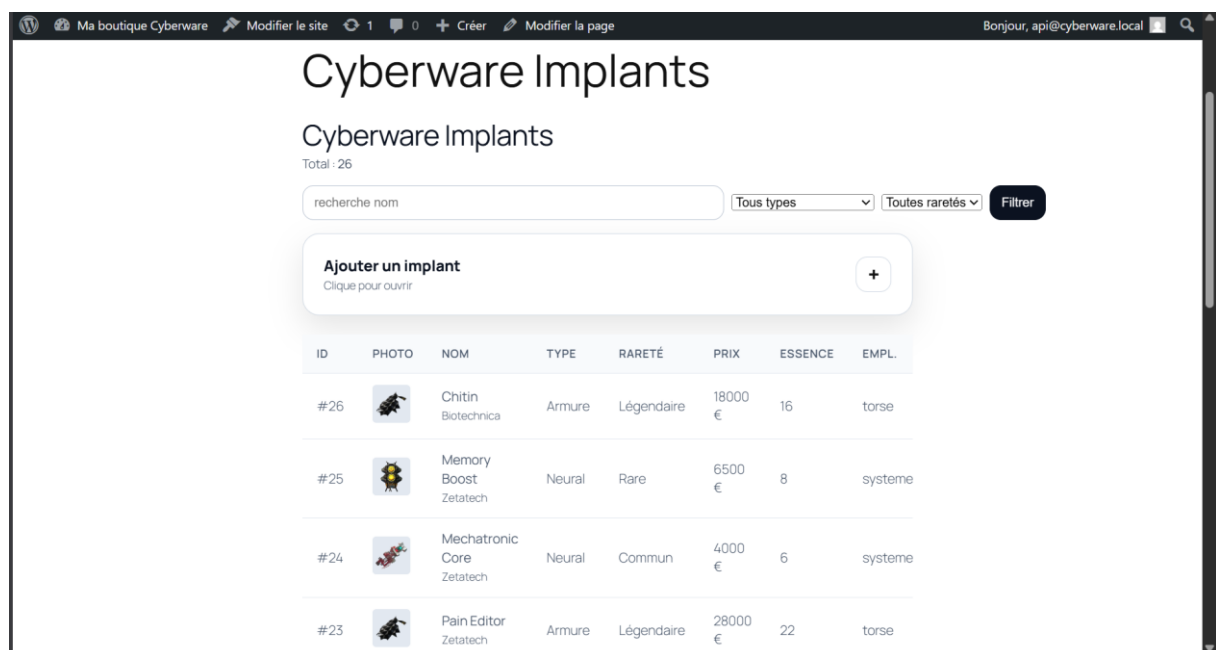
Lister tous les éléments

Dans le fichier `php/wp-content/plugins/CyberwareImplants/front.php`, le code effectue un appel à la classe `OdooPrimitive` pour récupérer les enregistrements.

```
return $obj->execute_kw($db, $uid, $apikey, 'cyberware.implant', 'search_read',  
[$domain], $kwargs);
```

Cet appel demande à Odoo de lister les éléments en respectant les filtres de recherche (`$domain`) et la pagination (`$limit`, `$offset`).

C'est aussi visible sur l'interface :



Ajouter un élément

Formulaire de création (HTML)

Dans le fichier `php/wp-content/plugins/CyberwareClients/front.php`, section `<div class='create-body'>` :

- **Select (Many2one)** : La balise `<select name='cyberwareclient_user_id'>` récupère la liste `$users_list` et affiche les options.
- **Checkboxes (Many2many)** : La boucle `foreach ($implants as $imp)` génère les `<input type='checkbox' name='cyberwareclient_implants[]'>` pour sélectionner plusieurs implants.
- **Action** : Le champ caché `<input type='hidden' name='cyberwareclient_action' value='create'>` est là pour router la demande d'ajout.

2. Traitement de la requête

Dans le fichier `php/wp-content/plugins/CyberwareClients/back.php` Localisation : Dans la fonction `cyberwareclient_handle_posts()`. On récupère l'ID utilisateur via `$user_id_form = absint($_POST['cyberwareclient_user_id'] ?? 0);`. On récupère les implants via `$implants = ... array_map('absint', ...)`. Et avec toutes les informations récupérées on passe tout à la fonction de création :

```
$id = cyberwareclient_odoo_create_client($nom_client, $pseudo, $implants, $image_b64, $user_id_form);
```

Envoi à l'API Odoo

Dans le fichier `php/wp-content/plugins/CyberwareClients/OdooPrimitive.php`, dans la fonction `cyberwareclient_odoo_create_client`.

La fonction construit un tableau `val` avec les éléments qu'elle reçoit (en incluant les relations `many2one` avec `'user_id' => $final_user_id` ET `many2many` avec `'implant_ids' => [6, 0, array_values(array_filter($implant_ids))]`). Et enfin elle crée tout avec `execute_kw`.

Et sur l'interface :

Ajouter un client
Clique pour ouvrir

Nom: Jules VINET LATRILLE | Pseudo: Juju | Utilisateur lié: api@cyberware.local

Photo: Choisir un fichier 3BABA7F2-FD64...92628AC7F67.jpg
PNG/JPG/WebP • max 2MB

Implants:

- ☒ Adrenaline Booster
- ☒ Bioconductor
- ☒ Biomonitor
- ☐ Blood Pump
- ☒ Cataresist
- ☐ Chitin
- ☐ Gorilla Arms
- ☐ Heal-On-Kill
- ☐ ImplantDeTest
- ☐ Kerenzikov
- ☐ Kiroshi Optics Mk3
- ☐ Lynx Paws
- ☐ Mantis Blades
- ☐ Mechatronic Core
- ☐ Memory Boost
- ☐ Microrotors
- ☐ Neural Mesh
- ☐ Neural Net
- ☐ Optic Cam
- ☐ Pain Editor
- ☐ Reflex Tense
- ☐ Reflexed Tendon
- ☐ Speed Link

Créer | Reset

Cyberware Clients

Total : 4

recherche nom / pseudo | Rechercher

Ajouter un client
Clique pour ouvrir

Âge: — | Essence: 0 / 100 | Restante: 100

Implants: Biomonitor, Cataresist, Bioconductor, Adrenaline Booster

Modifier / Supprimer

Âge: 80 | Essence: 0 / 10 | Restante: 10

Implants: aucun

Modifier / Supprimer

Âge: 18 | Essence: 0 / 80 | Restante: 80

Implants: aucun

Modifier / Supprimer

Modifier un élément

Dans le fichier php/wp-content/plugins/CyberwareClients/front.php :

- Le formulaire de modification est généré à l'intérieur de la boucle foreach (\$clients as \$c).
- Il est caché dans un élément `<details class='drawer'>` pour chaque carte, ce qui permet l'édition "in-place".
- Les champs simples sont pré-remplis via des codes de ce genre : `<input required name='cyberwareclient_nom_client' value='' . esc_attr($nom) . ''>`
- Pour les many-to-many :
 - D'abord le code vérifie si l'implant est posédé : `$checked = in_array($iid, $selected_implants, true) ? "checked" : "";`

- Ensuite l'attribut est ajouté à la balise : `<input type='checkbox' ... {$checked}>`.
- Pour le many-to-one :
 - Le code compare chaque option avec l'utilisateur actuel : `$selected = ($current_val === $uid_opt) ? "selected" : "";`
 - L'option correspondante est ainsi pré-sélectionnée : `<option ... {$selected}>...</option>`.

Supprimer un élément

Bouton de suppression

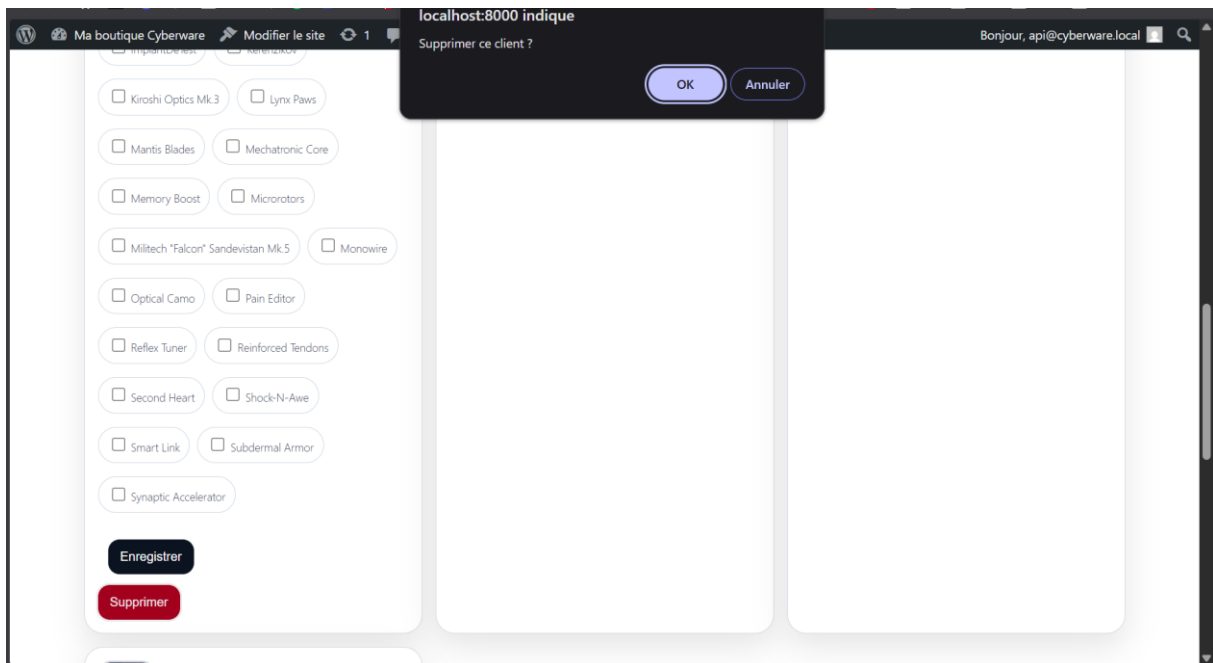
Dans le fichier front.php (à l'intérieur de la boucle foreach des clients, section `<details class='drawer'>`) :

- Le formulaire est présent avec un bouton "Supprimer".
- Il inclut une sécurité JavaScript native pour éviter les clics accidentels : `onsubmit="return confirm('Supprimer ce client ?');"`.
- Il envoie l'ID du client à supprimer via un champ caché :


```
<form method='post' class='delete' onsubmit="return confirm('Supprimer ce client ?');">
  <input type='hidden' name='cyberwareclient_action' value='delete'>
  <input type='hidden' name='cyberwareclient_client_id' value=' . $id . "'>
  <button class='btn danger' type='submit'>Supprimer</button>
</form>
```

Traitement de la suppression

Dans le fichier back.php, le bloc if (\$action === 'delete') récupère l'ID, le sécurise avec absint(), et appelle la fonction de suppression.

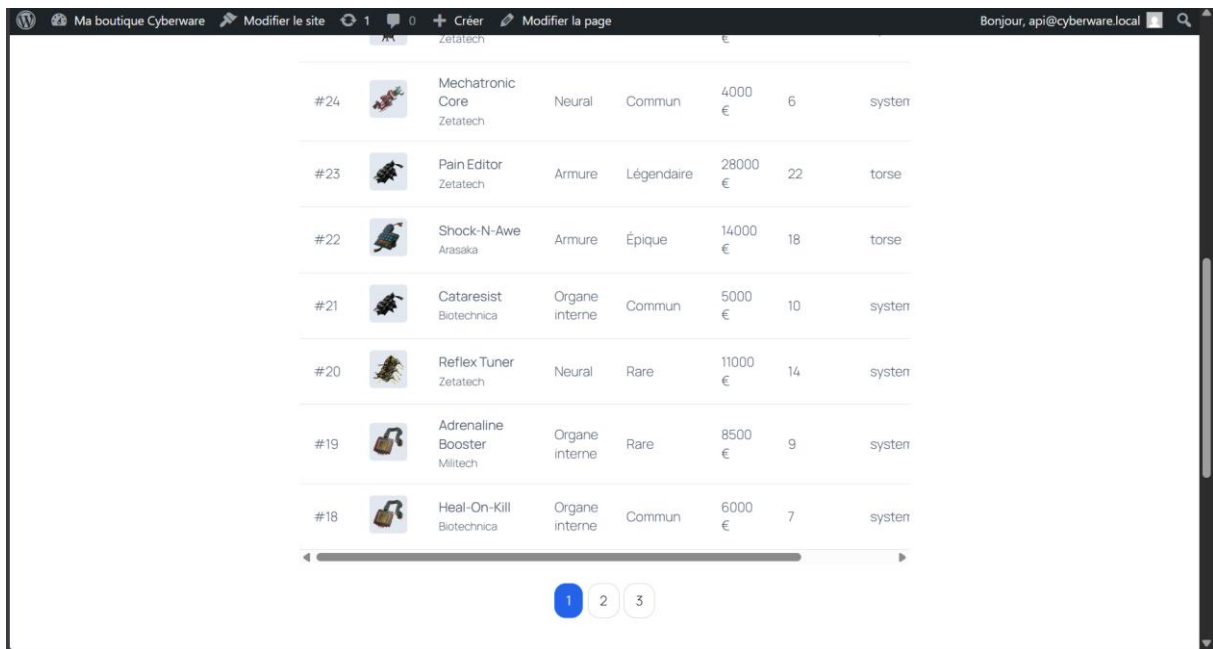


Appel API Odoo

Dans le fichier OdooPrimitive.php :

- La fonction `cyberwareclient_odoo_delete_client` utilise la méthode standard Odoo `unlink` pour supprimer l'enregistrement définitivement.
- L'appel est envoyé : `execute_kw(..., 'unlink', [[[int] $client_id]])`.

Tableau ou affichage paginé avec pagination de 10 éléments par page



#	Image	Nom	Catégorie	Rareté	Prix (€)	Quantité	Localisation
#24		Mechatronic Core Zetatech	Neural	Commun	4000 €	6	system
#23		Pain Editor Zetatech	Armure	Légendaire	28000 €	22	torse
#22		Shock-N-Awe Arasaka	Armure	Épique	14000 €	18	torse
#21		Cataresist Biotechnica	Organe interne	Commun	5000 €	10	system
#20		Reflex Tuner Zetatech	Neural	Rare	11000 €	14	system
#19		Adrenaline Booster Militech	Organe interne	Rare	8500 €	9	system
#18		Heal-On-Kill Biotechnica	Organe interne	Commun	6000 €	7	system

Pagination de 10 éléments par page

Dans `cyberwareclient/OdooPrimitive.php`, la fonction `cyberwareimplant_odoo_implants_page` définit `$limit = 10`.

Boutons de pagination numérotés et dynamiques

Dans le fichier `cyberwareImplants/front.php`, le bloc de pagination se situe à la fin de la fonction d'affichage (`cyberwareimplant_render_implants_page`).

On calcule le nombre de pages avec `$nb_pages`. On boucle dessus `for ($i = 1; $i <= $nb_pages; $i++)`. On génère les liens `{$i}` avec la classe active si c'est la page courante.

Et on conserve bien les filtres de recherche s'il y en a (`q`, `type`, `rarete`) dans les liens via `add_query_arg`.

Pagination réalisée côté API Odoo

Dans le fichier `cyberwareImplants/OdooPrimitive.php`, la fonction `cyberwareimplant_odoo_implants_page` calcule l'offset : `$offset = ($page - 1) * $limit`. Ensuite, on passe `'limit' => $limit` et `'offset' => $offset` dans le tableau `$kwargs` envoyé à `$obj->execute_kw(..., 'search_read', ...)` pour que ce soit Odoo qui filtre.

Champ de recherche relié à l'API

Front

Le fichier `cyberwareImplants/front.php` contient un champ de texte : `<input type='text' name='q' ...>`.

Il contient également une balise `<input type='hidden' name='pagename' value='cyberwareimplants'>` qui assure que l'utilisateur reste sur la bonne page lors de la soumission.

Le formulaire utilise la méthode GET (method='get'), ce qui permet de transmettre les critères de recherche (q, type, rarete) directement dans l'URL.

Front vers back

Dès le début du chargement de la page (cyberwareImplants/front.php), le code récupère et sécurise la valeur saisie : `$q = sanitize_text_field($_GET['q'] ?? '');`

Cette variable `$q` est ensuite transmise comme argument aux fonctions `cyberwareimplant_odoo_implants_count($q, ...)` et `cyberwareimplant_odoo_implants_page(..., $q, ...)`.

Back

Dans le fichier cyberwareImplants/OdooPrimitive.php, la fonction `cyberwareimplant_odoo_implants_page` traite cette variable.

Le code construit dynamiquement le "domaine" de recherche Odoo :

```
$q = trim((string) $q);
if ($q !== "") {
    "nom_implant"
    $domain[] = ['nom_implant', 'ilike', $q];
}
```

Ce domaine est ensuite envoyé dans l'appel XML-RPC `search_read`, garantissant que le filtrage est effectué par le serveur Odoo et non par le navigateur.

