

RAPPORT DE STAGE EN MILIEU PROFESSIONNEL

Dans le cadre du BTS SIO SLAM dispensé par l'ORT Lyon

Stage effectué du 05/01/26 au 13/02/26 au sein de la société

Web-Konseil - 7 Rue Amédée Bonnet, 69006 Lyon



REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude à Web-Konseil pour m'avoir accueilli au sein de son équipe durant cette période de stage. Cette expérience m'a permis d'acquérir de précieuses compétences qui me seront utiles dans la suite de mon parcours.

Je remercie tout particulièrement M. Salim Korrichi, mon maître de stage, pour son accompagnement, ses conseils et sa disponibilité. Il m'a guidé tout au long de la conception du projet tout en me laissant trouver par moi-même les solutions, ce qui m'a permis de développer autonomie, prises d'initiatives et montée en compétences. Ses explications, ses encouragements et sa bonne humeur ont été précieux pour moi.

Le fait de retrouver les locaux et l'équipe de Web-Konseil m'a permis de me sentir à l'aise bien plus vite, et le partage de connaissances avec les membres de l'équipe a été très formateur.

Je les remercie tous pour cette nouvelle expérience enrichissante.

Table des matières

I.	Introduction.....	4
II.	Présentation de l'entreprise.....	5
1)	A propos de Web-Konseil	5
2)	Services proposés.....	5
a)	Création et refonte de site e-commerce.....	5
b)	Webmarketing et référencement.....	5
c)	Maintenance et support technique.....	6
d)	Hébergement Web.....	6
3)	L'organisation.....	7
4)	Localisation géographique	7
III.	Présentation du projet	8
1)	Contexte	8
2)	Objectif	8
IV.	Les composants techniques.....	9
1)	Modélisation de la base de données	9
2)	Création d'un plugin Wordpress.....	10
3)	Synchronisation des plugins.....	11
4)	Algorithme de génération d'identifiant.....	12
5)	Chiffrement d'un mot de passe	14
V.	Les interfaces d'administration.....	16
1)	Interfaces de gestion des sites	16
2)	Méthodes de synchronisation	18
3)	Interface de recherche	21
VI.	Automatisation avec n8n.....	23
VII.	Déploiement de la solution	24
VIII.	Conclusion.....	26

I. Introduction

Actuellement en deuxième année de BTS SIO au lycée ORT Lyon, je devais réaliser un stage de 6 semaines en milieu professionnel.

Ce rapport détaille les missions que j'ai effectuées durant ce stage.

Mon choix de rejoindre à nouveau Web-Konseil s'est fait naturellement. Avant tout car les activités de l'entreprise s'inscrivent totalement dans la continuité de mon cursus en développement informatique. Mais aussi pour la qualité des relations humaines et l'excellente entente que j'ai avec l'équipe.

Ce stage est parfaitement aligné à ma trajectoire professionnelle. En effet, grâce au BTS SIO option SLAM, je me spécialise dans le développement et la conception d'applications tout en étant capable de maintenir la solution créée.

Le projet que me proposait Web-Konseil était parfaitement en adéquation avec cela puisque je devais améliorer une solution en concevant cette année un outil capable de récupérer les plugins WordPress installés sur les sites clients. Cela me permettrait de mettre en pratique des notions vues en formation comme la base de données, le maquettage, le développement en PHP ou la cybersécurité.

II. Présentation de l'entreprise

1) A propos de Web-Konseil

Web-Konseil est une agence web et digitale lyonnaise spécialisée dans l'accompagnement des entreprises pour leur présence en ligne.

Leur cœur de métier réside dans la création de solutions e-commerce. En tant qu'expert PrestaShop, l'agence met un point d'honneur à fournir des outils qualitatifs avec un excellent suivi, que ce soit pendant la création ou après le déploiement du site.

Que ce soit la création ou la refonte de sites, le référencement, la maintenance ou l'hébergement, Web-Konseil propose des solutions sur mesure et une relation de proximité à ses clients, grâce aux profils complémentaires de ses experts.

Le savoir-faire et le professionnalisme de l'agence est reconnue par plusieurs certifications majeures :

- Google Analytics et Google Ads
- Prestashop Expert
- Oracle

2) Services proposés

a) Création et refonte de site e-commerce

Web-Konseil accompagne les entreprises dans la création de sites e-commerce, principalement via PrestaShop. L'agence intervient également sur des projets complexes de migrations de données ce qui permet de passer d'une ancienne version de PrestaShop vers une version plus récente ou de transférer un site déjà existant vers PrestaShop.

b) Webmarketing et référencement

Pour garantir la visibilité de ses clients, Web-Konseil utilise différentes stratégies de marketing.

Le SEO est un référencement naturel pour le positionnement du site dans les résultats des moteurs de recherche, visant à améliorer sa visibilité auprès des utilisateurs. Le SEA est un référencement payant permettant de positionner rapidement un site en tête des résultats des moteurs de recherche grâce à des publicités ciblées. Le SMO regroupe l'ensemble des actions visant à optimiser la visibilité d'un site à travers les réseaux sociaux.

c) Maintenance et support technique

L'agence assure un suivi constant des plateformes qu'elle héberge. Cela comprend la correction de bugs, l'optimisation des performances et la mise en place de correctifs de sécurité, permettant de garantir une disponibilité maximale pour les sites e-commerces de leurs clients.

d) Hébergement Web

Web-Konseil propose des solutions d'hébergement web adaptées aux besoins de ses clients. L'entreprise assure la sécurité, la sauvegarde des données, la maintenance et l'assistance des hébergements mutualisés ou dédiés. Les Datacenters sont localisés en France et en Europe, dans un souci de souveraineté des données et d'engagement environnemental.

3) L'organisation

L'agence repose sur une structure de 4 collaborateurs. L'équipe est principalement basée à Lyon à l'exception d'un collaborateur qui travaille depuis Paris.

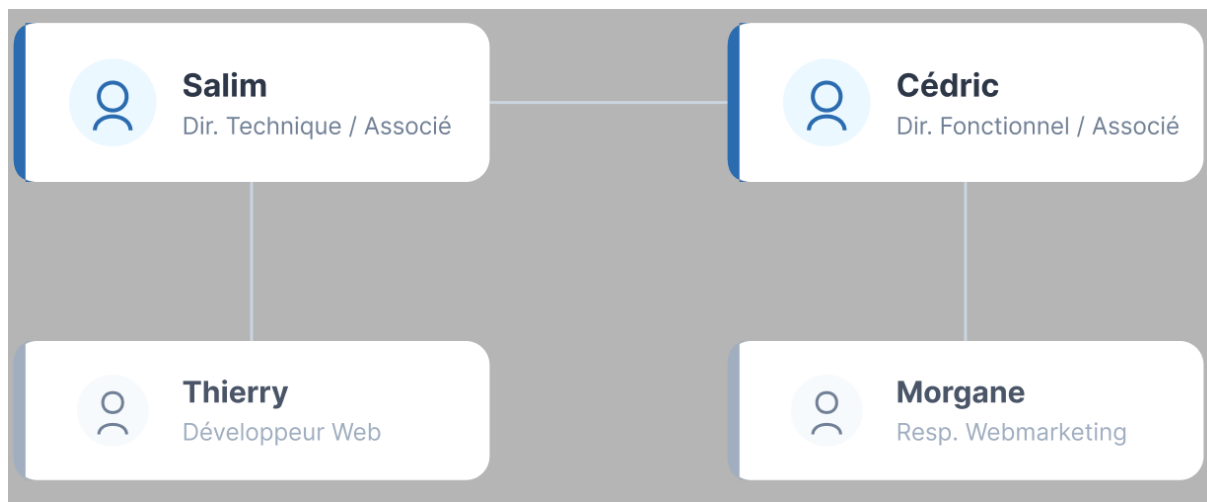


Figure 1 Organigramme de l'entreprise

4) Localisation géographique

L'agence est située au cœur du 6^{ème} arrondissement de Lyon, au 7 Rue Amédée Bonnet.



Figure 2 Localisation de l'entreprise

III. Présentation du projet

1) Contexte

L'une des missions majeures de Web-Konseil consiste à assurer la maintenance technique de plus d'une quarantaine de sites développés sous Wordpress. Chaque site intègre de nombreux plugins qui enrichissent le site de fonctionnalités ou d'améliorations graphiques. Jusqu'à présent la gestion de ces extensions était complexe car, dès que l'agence était informée de l'existence d'une faille de sécurité sur un plugin, l'équipe devait alors inspecter manuellement chaque site pour vérifier si le plugin vulnérable y était installé. Cette opération était particulièrement chronophage.

Pour répondre à cette problématique, j'ai consacré ce stage au développement d'un module de récupération dynamique des plugins. Ce projet s'inscrit dans la continuité de l'outil que j'ai développé et fait évoluer lors de mes précédents stages dans l'entreprise.

Pour mémoire, lors de ma première année, j'avais conçu la solution « SSLManager », un outil permettant de centraliser et d'alerter Web-Konseil sur les dates d'expirations des certificats SSL. Durant mon deuxième stage, j'ai fait évoluer cette solution en modernisant son interface visuelle et en ajoutant diverses fonctionnalités mineures. J'y avais surtout intégré une fonctionnalité majeure : la récupération automatique en temps réel de la date d'expiration réelle des certificats SSL.

2) Objectif

Cette année, l'objectif était de faire évoluer cette solution existante en y intégrant une fonctionnalité capable d'extraire la liste des plugins installés sur chaque Wordpress.

Grâce à cette centralisation, lorsque l'équipe aura connaissance d'une faille sur un plugin, elle n'aura plus besoin de parcourir l'ensemble des sites qu'elle gère pour localiser les sites impactés. Il suffira alors de consulter l'outil pour identifier immédiatement les sites utilisant le plugin concerné.

IV. Les composants techniques

Le projet repose sur une architecture client-serveur. J'ai développé l'application centrale avec le framework PHP Symfony. Toutes les données sont stockées et administrées via une base de données MySQL. L'interface utilisateur utilise un template bootstrap, qui permet d'avoir un affichage moderne. Les vues sont créées par le moteur Twig. J'ai aussi intégré JavaScript pour les interactions dynamiques.

Tout au long du projet, j'ai été amené à travailler sur différents composants techniques spécifiques au projet.

1) Modélisation de la base de données

La première étape du développement a été la conception de la base de données.

Après étude du besoin, j'ai identifié les champs nécessaires à la réalisation du projet : données liées aux sites Wordpress et aux plugins. J'ai donc créé une table pour les sites contenant l'URL, le nom et les identifiants du htaccess/htpasswd. Une seconde table stocke les plugins récupérés. J'ai mis en place une relation « Many-to-Many » entre ces deux tables. Un site possède plusieurs plugins et un plugin peut être installé sur plusieurs sites.

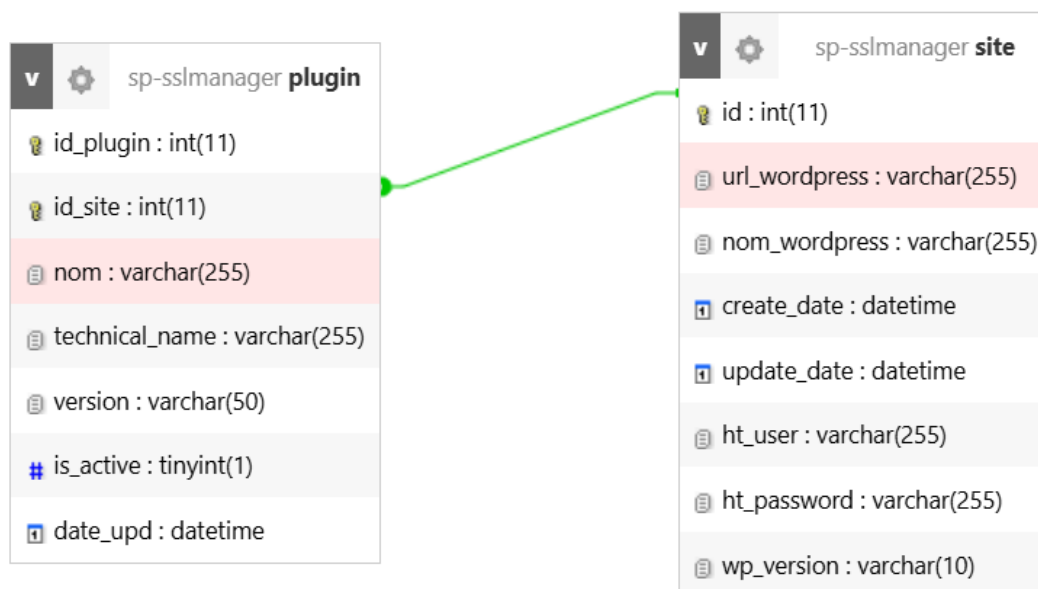


Figure 3 Modélisation des tables "plugin" et "site" dans la base de données

2) Création d'un plugin Wordpress

Pour assurer l'échange de données, j'ai développé un plugin Wordpress qui crée une route API personnalisée sur le site client. J'ai utilisé l'infrastructure native de Wordpress pour enregistrer un endpoint. C'est un mécanisme qui permet à des applications de communiquer et d'échanger des données accessibles via une URL spécifique. Ce choix technique, recommandé par mon maître de stage, permet un échange de données au format JSON.

Mon application Symfony pourra donc interroger le site Wordpress distant et récupérer les plugins installés.

A noter la mise place d'un token afin de sécuriser le endpoint et éviter qu'une personne non autorisée appelle l'URL permettant la récupération des données, ici la liste des plugins. Ce plugin sera déployé sur chaque site Wordpress.

```
// Retourne la liste des plugins installés
function wk_get_plugins_list() {

    // Charge la fonction si elle n'est pas dispo
    if (!function_exists('get_plugins')) {
        require_once ABSPATH . 'wp-admin/includes/plugin.php';
    }

    $all_plugins = get_plugins();
    $formatted_list = [];

    //Boucle sur chaque plugin pour extraire les infos
    foreach ($all_plugins as $path => $data) {

        // Détermination du slug du plugin
        $slug = dirname($path);
        if ($slug === '.') {
            $slug = basename($path, '.php');
        }

        // Construis un tableau de données
        $formatted_list[] = [
            'name' => $data['Name'],
            'technical_name' => $slug,
            'version' => $data['Version'],
            'is_active' => is_plugin_active($path),
        ];
    }

    // Récupère la version du site Wordpress
    global $wp_version ;

    // Renvoie la réponse au format JSON
    return new \WP_REST_Response([
        'site_url' => get_site_url(),
        'count' => count($formatted_list),
        'wp_version' => $wp_version,
        'plugins' => $formatted_list
    ], 200);
}
```

Figure 4 Code du plugin Wordpress

3) Synchronisation des plugins

Maintenant que le plugin est installé, je passe à l'étape de synchronisation.

L'application repose sur un service que j'ai créé. Salim, mon maître de stage, m'a demandé d'utiliser des requêtes cURL, bien que je ne connaisse pas ces commandes. Mes recherches m'ont permis de comprendre le fonctionnement et l'utilisation de ces requêtes et d'en discuter avec mon maître de stage. J'ai ainsi pu développer le code.

Le détail ci-dessous montre la communication avec les sites distants via la bibliothèque cURL. J'ai configuré la requête pour simuler le comportement d'un véritable navigateur.

```

40 $ch = curl_init();
41 curl_setopt($ch, option: CURLOPT_URL, $endpoint); // URL de l'API à appeler
42 curl_setopt($ch, option: CURLOPT_RETURNTRANSFER, value: true); // Récupérer la réponse
43 curl_setopt($ch, option: CURLOPT_FOLLOWLOCATION, value: true); // Autorise les redirections
44 curl_setopt($ch, option: CURLOPT_SSL_VERIFYPEER, value: false); // Désactive la vérification SSL
45 curl_setopt($ch, option: CURLOPT_TIMEOUT, value: 10); // Temps max d'attente de 10 secondes
46 // On se fait passer pour un navigateur
47 curl_setopt($ch, option: CURLOPT_USERAGENT,
48     value: 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36');
49 // Authentification HTTP Basic
50
51 if(!empty(trim($htUser)) && !empty(trim($htPassword))) {
52     curl_setopt($ch, option: CURLOPT_HTTPAUTH, value: CURLAUTH_BASIC);
53     curl_setopt($ch, option: CURLOPT_USERPWD, value: "$htUser:$htPassword");
54 }
55 curl_setopt($ch, option: CURLOPT_HTTPHEADER, [ // Headers envoyés à l'API
56     'Content-Type: application/json',
57     'X-WK-Token: ' . $apiToken
58 ]);
59
60 $response = curl_exec($ch); // Exécute la requête et récupère la réponse
61 $curlError = curl_error($ch); // Récupère une éventuelle erreur CURL
62 curl_close($ch); // Ferme la session CURL
63
64 // echo '<pre>';
65 // var_dump(json_decode($response, true));
66 // die();
67
68
69 $response = curl_exec($ch);
70 $httpCode = curl_getinfo($ch, option: CURLINFO_HTTP_CODE);
71 $curlError = curl_error($ch);
72 curl_close($ch);
73

```

Figure 5 Code des requêtes cURL

Une fois la réponse reçue au format JSON, le système transforme le flux en un tableau PHP. Le service vérifie d'abord que les données sont valides. Ensuite, le script compare les extensions présentes en base de données avec celles reçues dans le JSON. Si un plugin n'apparaît plus dans la réponse du site, il est alors supprimé de la base de données. A l'inverse si une différence de version est détectée, le système met à jour les enregistrements en base de données.

4) Algorithme de génération d'identifiant

Lors de l'enregistrement en base de données, chaque plugin reçoit un identifiant numérique calculé à partir de son nom technique. L'objectif est de faciliter l'indexation en base de données. Cet identifiant est généré par une méthode que j'ai développée, qui garantit qu'un même plugin possédera toujours une signature unique et constante.

Le processus débute par la création d'une empreinte sécurisée du « nom technique » du plugin avec l'algorithme de hachage SHA512.

Je découpe ensuite cette chaîne de caractères pour traiter chaque élément individuellement via une boucle. L'algorithme distingue donc les chiffres des lettres. Les chiffres sont d'abord additionnés entre eux pour former un premier total. Pour les lettres, j'utilise une fonction native de PHP qui permet de récupérer le code ASCII et le convertir en une valeur numérique de 1 à 26.

Une fois les lettres converties en chiffres, elles sont additionnées pour former un deuxième total. Le résultat final provient de la concaténation des deux totaux.

Le schéma réalisé ci-dessous illustre la logique appliquée.

Algorithme de Hash

PLUGIN **elementor-pro**

Hash complet :

```
eab2976fcdd2e10d45e30299e3c81aca552d1c1f7af9f3f6e38c52da63c36dad5bfb001d7bedf11170a3708eb18f7e1148ef4d9bab8eeb9dd3d2259bf872ea9b
```

Pour l'exemple, nous utilisons seulement les 15 premiers caractères

E

Exemple : Hash "eab2976fcdd2e10"

eab2976fcdd2e10

e a b 2 9 7 6 f c d d 2 e 1 0

1

Additionner tous les chiffres

On prend uniquement les chiffres (0-9) :

2 + 9 + 7 + 6 + 2 + 1 + 0

$2 + 9 + 7 + 6 + 2 + 1 + 0 = 27$

2

Convertir les lettres et additionner

Chaque lettre = sa position dans l'alphabet (a=1, b=2, c=3...)

e = 5 a = 1 b = 2 f = 6 c = 3
d = 4 d = 4 e = 5

$5 + 1 + 2 + 6 + 3 + 4 + 4 + 5 = 30$

$5 + 1 + 2 + 6 + 3 + 4 + 4 + 5 = 30$

3

Coller les deux résultats

On colle la somme des chiffres avec la somme des lettres :

27 + 30

↓

RÉSULTAT FINAL

2730

Récapitulatif

Chiffres

27

+

Lettres

30

=

Résultat

2730

Figure 6 Modélisation de la logique de hachage

5) Chiffrement d'un mot de passe

Certains sites clients sont protégés par une authentification via un htaccess/htpasswd.

Pour automatiser la synchronisation, l'appel cURL doit pouvoir s'authentifier sans intervention humaine. L'objectif était de garantir la confidentialité des données en évitant le stockage des mots de passe en clair, tout en conservant une méthode de déchiffrement. J'ai conçu un service de chiffrement de mot de passe pour répondre à ce besoin. Contrairement aux mots de passe utilisateurs, ces accès ne peuvent pas être hachés car ils doivent rester lisibles par les requêtes cURL.

Le système repose sur un algorithme de chiffrement par décalage. J'ai défini un alphabet personnalisé comprenant les chiffres, les lettres minuscules et majuscules. Chaque caractère du mot de passe initial subit un décalage de 7 positions vers la droite dans cet alphabet. Pour renforcer la protection, j'ai ajouté un mécanisme de salage.

Avant chaque chiffrement, une fonction génère une chaîne aléatoire de 5 caractères. Ce sel est placé au début du mot de passe chiffré. Cette méthode garantit qu'un mot de passe aura une apparence différente à chaque enregistrement en base de données.

Le schéma réalisé ci-après décrit la méthode de chiffrement.

Lors de la synchronisation, l'application effectue l'opération inverse. La méthode de déchiffrement retire d'abord les 5 caractères du sel. Elle applique ensuite un décalage de 7 positions vers la gauche sur tous les caractères pour retrouver le mot de passe original.

Cette méthode de chiffrement/déchiffrement permet d'utiliser les identifiants pour la requête cURL, tout en garantissant que le mot de passe ne soit pas stocké en base de données de manière lisible.

Chiffrement par Décalage avec Salage

1 Alphabet personnalisé

Définition de l'alphabet utilisé :

0123456789abcdefghijklmnopqrstuvwxyzABCDEF
GHIJKLMNOPQRSTUVWXYZ

62 caractères au total : chiffres + minuscules + majuscules

2 Décalage de 7 positions

Exemple avec le caractère "a" :

Position initiale : 10

a → h

Décalage +7

Nouvelle position : 17

3 Application du chiffrement

Exemple avec le mot de passe "Pass123" :

P	a	s	s	1	2	3
↓ +7	↓ +7	↓ +7	↓ +7	↓ +7	↓ +7	↓ +7
W	h	z	z	8	9	a

Pass123 → Whzz89a

4 Ajout du sel (5 caractères aléatoires)

Un sel aléatoire est généré et placé au début du mot de passe chiffré :

Sel aléatoire généré : "aB7xK"

Résultat final en base de données :

aB7xK + whzz89a = aB7xKWhzz89a

Figure 7 Modélisation du chiffrement par décalage avec salage

V. Les interfaces d'administration

1) Interfaces de gestion des sites

La finalité technique de ce projet réside dans la mise en place d'une gestion centralisée et administrable par les équipes Web-Konseil. On y retrouve différentes interfaces :

- Un tableau de bord permettant de visualiser l'ensemble des sites Wordpress enregistrés avec la version du Wordpress et le nombre de plugins installés. On y retrouve également les boutons d'actions pour (*figure 8*) :
 - L'ajout d'un nouveau site
 - L'édition d'un site
 - La suppression d'un site
 - La synchronisation des plugins
- Une interface de création, permettant l'ajout d'un nouveau site Wordpress : Nom du site, URL du site et les informations htaccess/httpasswd (*figure 9*).
- Une interface d'édition permettant (*figure 10*) :
 - De modifier les informations renseignées lors de la création du site
 - De visualiser l'ensemble des plugins installés sur le site avec sa version et son statut (actif ou inactif)
 - De synchroniser les plugins

Wordpress > Liste des sites

[Ajouter](#) [Synchroniser tous les sites](#)

Nom du site	URL Wordpress	Version Wordpress	Nombre de plugins	Actions
AARKA	https://www.aarka.fr	6.7.4	10	↺ ✎ 🗑
AERONICS	https://www.aeronics.fr	6.9.1	9	↺ ✎ 🗑
AIGUILLE VERTE CHAMONIX	https://www.aiguilleverte-chamonix.com	4.9.28	21	↺ ✎ 🗑
APPORATIQUE	https://www.apporatique.fr	6.7.4	15	↺ ✎ 🗑
ARROGEST	https://www.arrogest.com	6.5.7	9	↺ ✎ 🗑
ATELIER MUSÉE DE LA TERRE	https://www.ateliermuseedelaterre.com	6.0.3	18	↺ ✎ 🗑
BATITEC SB2	https://www.batitec-sb2.com	6.5.7	11	↺ ✎ 🗑

Figure 8 Tableau de bord

Wordpress > Liste des sites > Ajouter un site

Nom du site :

htlogin :

URL du site :

htpasswd :

https://www.exemple.com

+ Ajouter

Retour à la liste

Figure 9 Interface de création d'un nouveau site

Wordpress > Liste des sites > Modifier

Nom du site:

AARKA

URL du site:

https://www.aarka.fr

htlogin :

htpasswd :

☐ Supprimer le mot de passe

Synchroniser les plugins

Modifier

Retour à la liste

Version du Wordpress : 6.7.4

Nom du plugin	Version	Status
All In One WP Security	5.3.8	✓
CookieYes GDPR Cookie Consent	3.2.7	✓
Elementor	3.24.4	✓
Elementor Pro	3.24.2	✓
Post SMTP	3.0.1	✓
Smart Slider 3	3.5.1.24	✓
Sticky Header Effects for Elementor	1.7.2	✓
W3 Total Cache	2.8.2	✓
WPFront Scroll Top	2.2	✓
Yoast SEO	24.2	✓

Figure 10 Interface d'édition et de visualisation

2) Méthodes de synchronisation

Il existe plusieurs méthodes pour synchroniser les plugins depuis la solution créée :

- Synchronisation manuelle unitaire
- Synchronisation manuelle globale
- Synchronisation automatique via une tâche cron

La synchronisation individuelle unitaire s'effectue de deux manières :

- Via une icône dédiée située dans la colonne des actions du tableau de bord
- Via un bouton de synchronisation dans la page d'édition d'un site Wordpress

Cette fonctionnalité permet de mettre à jour les plugins d'un site spécifique sans impacter le reste de la liste. Sur le plan technique, le système cible uniquement l'URL du site sélectionné pour en extraire les informations.

Cette approche ciblée est particulièrement utile après une intervention technique sur une seule plateforme client.

Actions

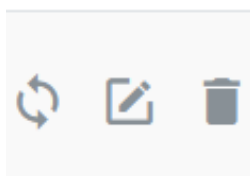


Figure 11 Icône de synchronisation (double flèches) du tableau de bord



Figure 12 Bouton de synchronisation en mode édition de site

J'ai aussi développé une fonctionnalité de synchronisation globale.

Un bouton principal situé au-dessus du tableau de bord permet de lancer la synchronisation sur tous les sites enregistrés en base de données. L'application parcourt alors chaque site et exécute automatiquement la procédure de récupération des plugins.

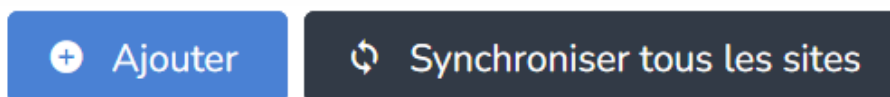


Figure 13 Bouton pour synchroniser tous les plugins

J'ai également mis en place une tâche cron appelée via une URL spécifique afin d'assurer une synchronisation totalement automatique. J'ai développé la logique de cette fonctionnalité, tandis que la configuration finale sur le serveur a été réalisée par mon maître de stage.

A noter que sur chaque synchronisation manuelle, une fenêtre modale apparaît à l'écran avec une icône de chargement. Ce visuel informe l'utilisateur que le traitement de données est en cours. Une fois la récupération terminée, la modal change pour afficher un message de validation ou d'échec de la synchronisation. Puis la page se recharge pour afficher les nouveaux plugins et les versions à jour. Ce module permet une visibilité sur le bon fonctionnement du système.

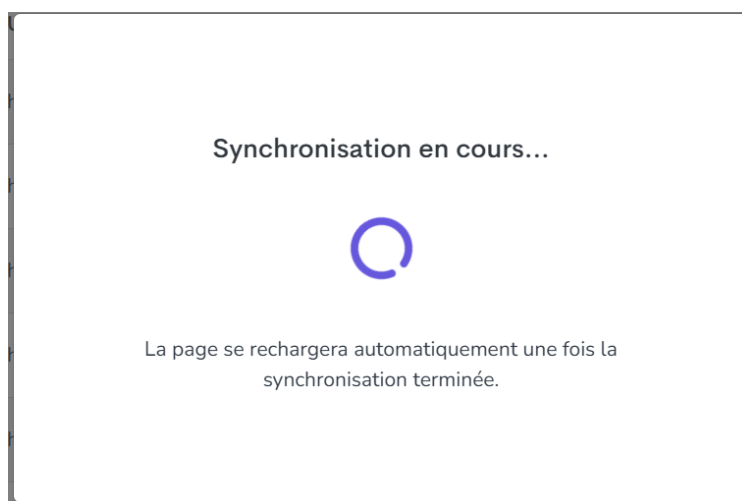


Figure 14 Popup pour signifier que la synchronisation est en cours

✔ **Synchronisation réussie !**

10 plugins synchronisés.

Figure 15 Popup pour signifier que la synchronisation est réussie

❗ **Erreur de synchronisation**

Erreur CURL : Could not resolve host: test.crypte.fr

Figure 16 Popup pour signifier que la synchronisation a échoué

❗ **Erreur de synchronisation**

18 site(s) synchronisé(s)

1 site(s) non-synchronisé(s)

La page va se recharger...

Figure 8 Popup pour signifier que la synchronisation est réussie pour certains sites mais a échoué pour d'autres

3) Interface de recherche

J'ai également développé une page de consultation globale permettant de lister l'intégralité des plugins présents sur tous les sites Wordpress enregistrés.

Cette interface centralise les données dans un tableau composé de 4 colonnes : le nom du plugin, sa version, le site Wordpress et son statut (actif ou inactif).

Il s'agit ici de l'objectif final du projet : fournir un outil de recherche centralisé pour savoir exactement où est installé chaque plugin, évitant ainsi de devoir inspecter manuellement chaque site en cas d'intervention urgente.

Rechercher

Tout afficher ▼

Nom du plugin	Version	Nom du site	Status
All In One WP Security	5.3.8	AARKA	✓
CookieYes GDPR Cookie Consent	3.2.7	AARKA	✓
Elementor	3.24.4	AARKA	✓
Elementor Pro	3.24.2	AARKA	✓
Post SMTP	3.0.1	AARKA	✓

Figure 98 Page "Liste des plugins"

La fonctionnalité majeure de cette page est le système de filtrage situé en haut de l'interface. L'utilisateur peut sélectionner un nom d'extension dans la liste déroulante ou l'écrire pour isoler les résultats.

Rechercher

Tout afficher ▲

Elem

Tous les plugins

- CWA Addons for Elementor
- Elementor
- Elementor Pro
- Image Optimizer by Elementor – Compress, Resize and Optimize Images
- Make Column Clickable Elementor

Figure 109 Page "Liste des plugins"

Le tableau se met à jour pour n'afficher que les sites possédant le plugin choisi. Cette fonction de recherche permet d'identifier en quelques secondes toutes les plateformes vulnérables lorsqu'une faille est signalée.

Rechercher

Elementor Pro ▼

Nom du plugin	Version	Nom du site	Status
Elementor Pro	3.24.2	AARKA	✓
Elementor Pro	3.33.2	AERONICS	✓
Elementor Pro	3.16.2	APPORATIQUE	✓
Elementor Pro	3.6.5	ATELIER MUSEE DE LA TERRE	✓
Elementor Pro	3.20.2	BATITEC SB2	✓

Figure 20 Page "Liste des plugins" après la recherche d'un plugin

VI. Automatisation avec n8n

En complément des missions initiales, j'ai pris l'initiative de proposer à l'entreprise la mise en place d'une surveillance automatisée des vulnérabilités. Mon objectif était de lier mon outil de supervision à une source d'informations. J'ai utilisé l'outil d'automatisation n8n pour concevoir ce workflow.

N8n est un outil d'automatisation low-code permettant de connecter divers services via une interface visuelle. J'ai proposé de créer une automatisation, et plus spécifiquement sur n8n, car cela fait déjà plusieurs mois que j'apprends en autodidacte en imaginant des scénarios différents. J'ai donc tout de suite vu la possibilité d'inclure n8n dans la continuité du projet.

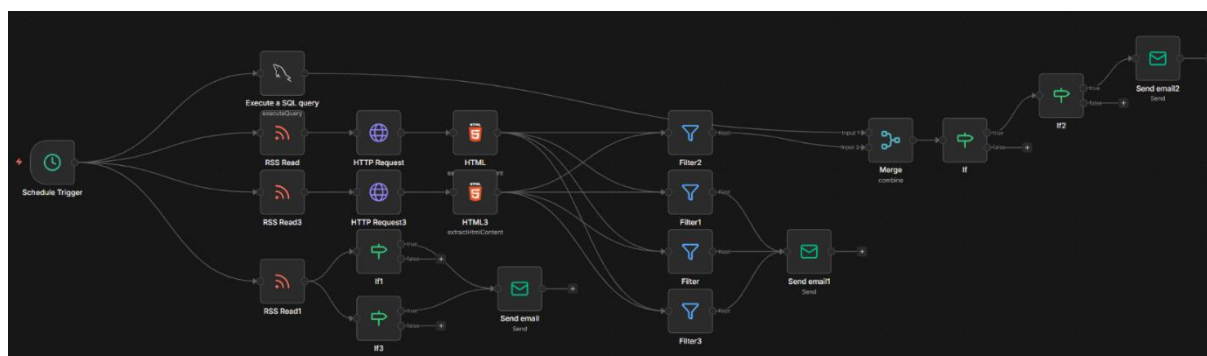


Figure 21 Automatisation complète sur n8n

Le processus démarre par un déclencheur temporel qui lance l'exécution toutes les 4 heures. Le workflow interroge d'abord la base de données « sslmanager » pour récupérer le nom de tous les plugins. En parallèle, plusieurs nœuds effectuent une lecture de flux RSS sur différents sites internet. Les flux RSS fournissent les premières lignes de l'article, j'ai donc utilisé les nœuds « HTTP Request » et « HTML Extract » pour récupérer et extraire le contenu complet des articles.

Cependant, la structure technique d'un site cible ne permettait pas d'isoler proprement le texte de l'article du reste de la page. Pour garantir la stabilité du workflow et éviter la récupération de données inutiles, j'ai dû limiter la surveillance aux seules données fournies par le flux RSS sur cette source.

Les données récoltées passent ensuite par une série de filtres dont le rôle est de comparer les mots-clés extraits des articles avec la liste des plugins récupérée précédemment dans la base de données « SSLManager ». Cette étape permet d'isoler uniquement les articles qui concernent les plugins vulnérables. Enfin, si une correspondance est trouvée, le système déclenche automatiquement l'envoi d'un e-mail d'alerte.

VII. Déploiement de la solution

Le déploiement de l'application a constitué la phase finale de mon projet. Pour réaliser cette opération, je me suis connecté au serveur de l'entreprise en utilisant les accès sécurisés fournis par l'entreprise. Cette étape de mise en ligne garantit que l'outil est désormais accessible par l'ensemble des collaborateurs.

Une phase importante de transfert de compétences a été organisée avant la livraison définitive du module. J'ai pris le temps de présenter l'interface à mes collègues et leur expliquer le fonctionnement détaillé des différentes synchronisations. Nous avons effectué plusieurs tests en conditions réelles pour valider la récupération des données sur différents sites Wordpress.

J'ai également fourni une procédure simplifiée pour l'installation du module sur de nouveaux sites clients. La manipulation nécessite le transfert du plugin agent via FTP dans le répertoire des extensions du Wordpress cible. Une fois cette étape réalisée, l'entreprise peut enregistrer l'URL dans le tableau de bord de l'application pour lancer la première synchronisation.

Je présente en page suivante l'architecture globale du projet qui reprend les étapes du fonctionnement de la solution.

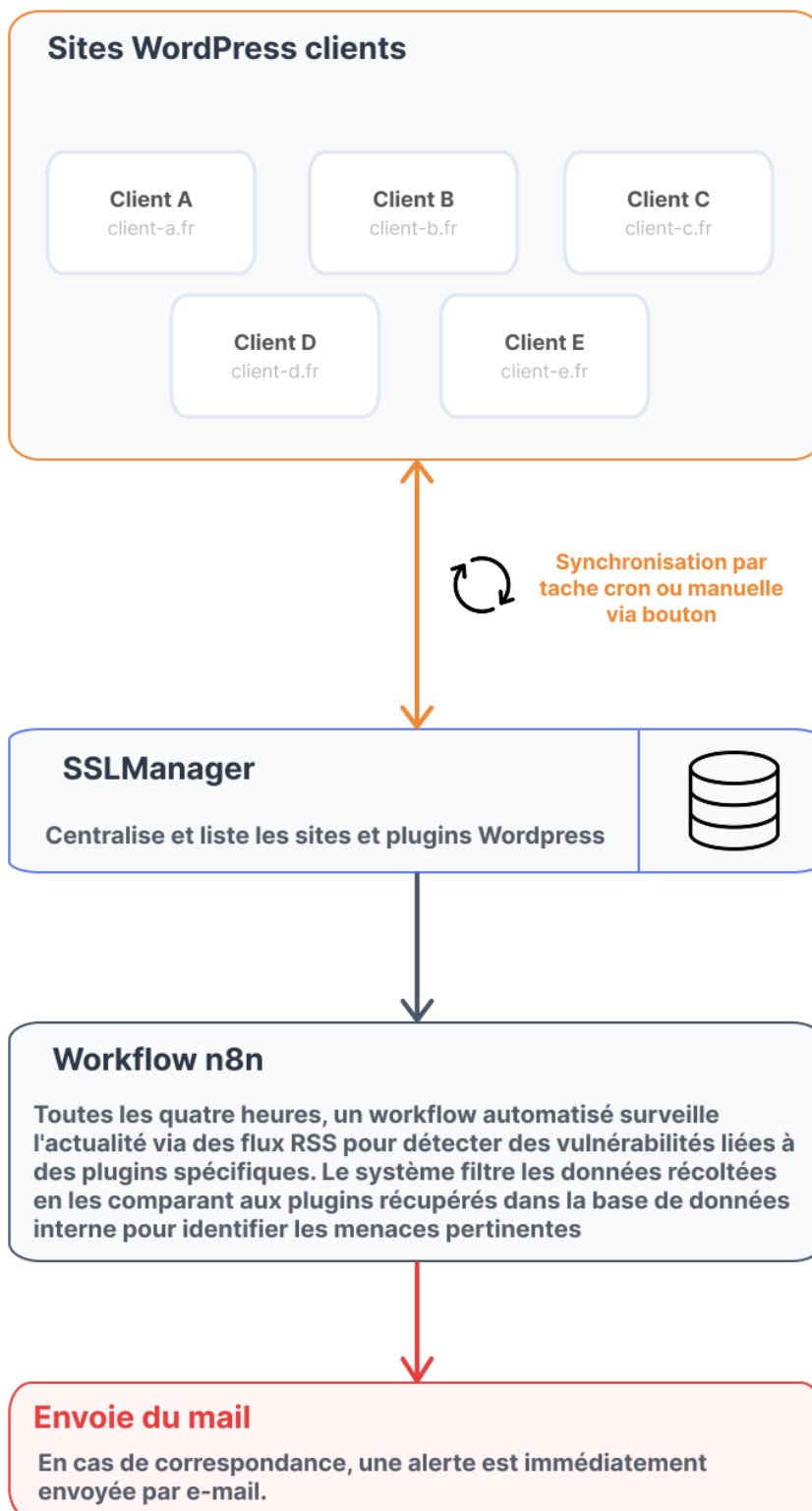


Figure 22 Schéma du projet final

VIII. Conclusion

Ce stage chez Web-Konseil a été une expérience extrêmement enrichissante et formatrice, tant sur le plan professionnel que personnel. Durant ces six semaines, j'ai eu l'opportunité de découvrir de multiples facettes du métier de développeur. Ce troisième passage au sein de l'entreprise m'as permis de mener un projet complexe de bout en bout.

Tout d'abord, ce projet m'a permis de mobiliser les compétences acquises lors de mon cursus scolaire tout en enrichissant mon bagage technique. En effet, en développant ce module de supervision, j'ai acquis des compétences opérationnelles précieuses en Framework Symfony, en base de données (notamment avec la relation « Many-to-many ») et en conception d'API REST.

De plus, et bien que je me sois appuyé sur des bases issues de mes expériences précédentes, le niveau de difficulté a été nettement supérieur, m'obligeant à mobiliser des ressources de réflexion plus poussées afin de résoudre chaque défi, et, au-delà de l'aide de mon maître de stage, d'interroger des sources d'informations variées ce qui m'a conduit à élargir ma palette technique. Par exemple, la mise en pratique d'un algorithme de cryptage et de décryptage m'a permis une mise en pratique indispensable pour mon futur professionnel et très valorisante personnellement.

Également, ce stage m'a permis de développer mon autonomie et ma prise d'initiative. En proposant moi-même l'automatisation via n8n, j'ai compris l'importance d'être à l'écoute des besoins de l'agence pour apporter des solutions innovantes. Cette expérience a renforcé ma capacité à analyser une problématique et à y répondre, en ajoutant une proposition pour apporter davantage d'efficacité. J'ai également beaucoup apprécié faire une restitution de mon projet à l'équipe, ce qui m'a demandé d'être précis et parfaitement compréhensible pour chaque professionnel présent.

Je fais ici un parallèle avec la relation client qu'il m'a été donné d'observer durant mon stage. En effet, j'ai assisté à de nombreuses reprises à des entretiens avec des clients de Web-Konseil et j'ai pu m'imprégner de la posture professionnelle adoptée par chacun : écoute active, identification des demandes, reformulation et propositions de résolutions claires et précises.

Enfin, cette nouvelle expérience chez Web-Konseil confirme mon intérêt pour la conception et le développement de solutions et applications informatiques. Ce projet m'encourage à continuer à approfondir mes compétences métiers pour être en mesure de répondre avec professionnalisme à chaque problématique.

Comme mes différents projets chez Web-Konseil, ce stage a été une formidable opportunité d'apprentissage. Je suis très reconnaissant à mon maître de stage Salim Korrichi et à toute l'équipe de Web-Konseil d'avoir participé, grâce à leur accueil bienveillant et leurs précieux conseils techniques, à forger ma future posture professionnelle.