Installer et configurer Kinimi

I) Prérequis

Pour installer une instance de Kinimi sur votre serveur personnel, il faut respecter les prérequis suivants :

- Système d'exploitation : Nous vous recommandons un système à base Linux, mais vous pouvez choisir un autre système selon vos habitudes et votre expérience.

- Serveur : Nous recommandons Apache2, un serveur web aisément configurable. Nginx est une alternative envisageable.

- Langage CGI : Kinimi est écrit en PHP 7, et est compatible PHP 7.1 et supérieures. Si votre serveur n'est pas équipé de PHP 7 en CGI, Kinimi ne fonctionnera pas !

- Langage CLI : PHP 7 doit aussi être installé en mode CLI afin de pouvoir utiliser l'outil artisan (si cette condition n'est pas remplie, vous ne pourrez pas déployer la base de données simplement)

- Extensions PHP : Votre serveur doit obligatoirement remplir les pré-requis suivants : Avoir installé et activé OpenSSL PHP Extension, PDO PHP Extension, Mbstring PHP Extension, Tokenizer PHP Extension, XML PHP Extension. L'absence d'une ou plusieurs de ces extensions générera des erreurs majeures lors de l'exécution du code.

- Serveur de bases de données (SGBD) : Kinimi est compatible avec tous les SGBD du marché. Nous conseillons PostgreSQL aux utilisateurs ayant une expérience sur cette plateforme. MySQL est une alternative fiable et simple à mettre en place pour les utilisateurs débutants.

- Serveur SMTP : Vous devez avoir un serveur SMTP sur votre machine, ou un serveur distant configuré de manière à ce que Kinimi puisse interagir avec lui. Ceci est nécessaire pour les fonctionnalités de récupération de mot de passe et pour contacter les administrateurs.

II) Configurer votre SGBD

Vous devrez ensuite configurer votre SGBD afin de créer une base de données du nom de votre choix et d'interclassement utf8_unicode_ci.

Enfin, créez un utilisateur sur votre SGBD ayant les permissions en écriture et en lecture sur votre base de données nouvellement créé. Mettez un mot de passe assez long et diversifié afin de sécuriser les accès à ce compte.

Pour effectuer ces étapes, nous vous invitons à consulter la documentation de votre SGBD.

III) Télécharger Kinimi

Une fois ces actions effectuées, téléchargez la dernière version de kinimi sur Chamilo si cela n'est pas encore fait.

IV) Installer et configurer Kinimi

Maintenant que vous avez récupéré l'archive, décompressez la dans le dossier de votre serveur web (sur une installation Apache standard sur Linux, ce serait le répertoire /var/www/html par exemple, référez-vous à la configuration de votre serveur).

Il faudra ensuite copier le fichier .env.example en .env dans le même dossier, puis modifier le .env nouvellement crée avec vos informations :

APP_NAME=Kinimi (NE PAS MODIFIER)
APP_ENV=local (NE PAS MODIFIER)
APP_KEY=base64:xo4Hv8t+9YLPAJyxQKN+2s94nLY6dCnNw0jAb8+VO9s= (NE PAS MODIFIER)
APP_DEBUG=false (Mettez cette variable à false si du public peut accéder au site)
APP_LOG_LEVEL=debug (NE PAS MODIFIER)
APP_URL=http://votrenomdedomaine.com (Mettez ici l'url de votre site suivi d'un éventuel chemin d'accès au dossier public de kinimi)

DB_CONNECTION=mysql (NE PAS MODIFIER)
DB_HOST=127.0.0.1 (L'adresse de votre serveur SQL)
DB_PORT=3306 (Le port de votre serveur SQL)
DB_DATABASE=kinimi (Le nom de la base de données préalablement créé)
DB_USERNAME=username (Le nom d'utilisateur dédié au site)

DB_PASSWORD=password (Le mot de passe associé)

BROADCAST_DRIVER=log (NE PAS MODIFIER)
CACHE_DRIVER=file (NE PAS MODIFIER)
SESSION_DRIVER=file (NE PAS MODIFIER)
SESSION_LIFETIME=120 (NE PAS MODIFIER)
QUEUE_DRIVER=sync (NE PAS MODIFIER)

REDIS_HOST=127.0.0.1 (NE PAS MODIFIER)
REDIS_PASSWORD=null (NE PAS MODIFIER)
REDIS_PORT=6379 (NE PAS MODIFIER)

MAIL_DRIVER=smtp (Insérez ici les informations de votre serveur SMTP)
MAIL_HOST=smtp.mailtrap.io (Adresse du serveur SMTP)
MAIL_PORT=2525 (Port du serveur SMTP)
MAIL_USERNAME=username (Nom d'utilisateur du serveur SMTP)
MAIL_PASSWORD=password (Mot de passe du serveur SMTP)
MAIL_ENCRYPTION=null (NE PAS MODIFIER)
MAIL_FROM_ADDRESS=no-reply@kinimi.local (Adresse mail indiquée dans le champ From des mails de notification)
MAIL_ADMIN_ADDRESS=admins@kinimi.local (Adresse mail indiquée dans le champ To des mails de contact, les utilisateurs vous contacteront et répondront en utilisant cette adresse.)
MAIL_FROM_NAME=Kinimi (Nom indiquée dans le champ From des mails de notification)

Une fois ces étapes effectuées, ouvrez un terminal dans le dossier racine de Kinimi, puis exécutez la commande suivante afin de créer la structure des bases de données :

php artisan migrate:fresh

Une fois cette commande exécutée, accédez à Kinimi (Par exemple : ip_serveur/public si kinimi est situé à la racine de votre site), puis connectez vous sur Kinimi avec le compte suivant :

Adresse mail : admin@kinimi.local
Mot de passe : ki-admin-ni-local-mi

Puis modifiez l'adresse mail et le mot de passe par défaut afin de sécuriser l'accès à ce compte.

V) En cas de déploiement à destination du public

Si vous prévoyez de déployer ce jeu à destination du public sur un serveur de production, nous vous invitons à respecter les conseils suivants afin de sécuriser un peu plus l'application :

- Pointer votre nom de domaine vers le dossier /public : En l'état, vous devez taper adresse_ip_ou_ndd/public pour accéder à Kinimi sur votre serveur personnel. Afin d'obtenir un résultat propre et d'empêcher à un tiers l'accès au code source de manière intempestive, vous pouvez configurer votre serveur pour qu'il vous redirige automatiquement dans ce dossier. Nous vous invitons à consulter la documentation de votre serveur web pour plus d'informations.

- Répliquer votre serveur SQL : Cette manipulation permet de conserver des copies identiques de votre base de données sur plusieurs serveurs SQL (distants ou non), afin de prévenir tout problème de disparition ou rollback des données enregistrées.

- Installer un pare-feu et un anti-ddos : Les attaques par déni de service sont omniprésentes ces dernières années. Chaque jour, ce sont des milliers de tentatives qui sont dénombrées, et souvent plus qui sont passées sous silence, et bon nombre d'entre elles arrivent à leurs objectifs initiaux. Un simple système Anti-DDOS couplé à un pare-feu logiciel fait souvent l'affaire pour protéger le serveur de la majorité des attaques.