

FarmShop API

Version: 1.0.0

Description: API complète pour la plateforme e-commerce FarmShop. Cette API permet de gérer les produits, commandes, locations, paiements et bien plus encore.

Base URL API: <http://127.0.0.1:8000/api>

Documentation URL: <http://127.0.0.1:8000/api/documentation>

Généré le: 14/08/2025 11:02:28

Table des matières

- [Admin](#)
- [Dashboard](#)
- [Analytics](#)
- [Statistics](#)
- [Blog](#)
- [Categories](#)
- [Comments](#)
- [Reports](#)
- [Posts](#)
- [Cart](#)
- [Checkout](#)
- [Cookies](#)
- [GDPR](#)
- [Orders](#)
- [Rentals](#)
- [Inspection](#)
- [Returns](#)
- [Products](#)
- [Stock](#)
- [Likes](#)
- [Users](#)
- [Profile](#)
- [Newsletter](#)
- [Messages](#)

Admin

GET

`/api/admin/dashboard`

Dashboard principal administrateur

Affiche le tableau de bord principal avec toutes les statistiques du site (Admin uniquement)

Réponses:

200 - Dashboard récupéré avec succès

403 - Accès non autorisé

GET

`/api/admin/dashboard/statistics`

Statistiques avancées du site

Page complète de statistiques avec données détaillées pour tous les modules (Admin uniquement)

Paramètres:

period (query) - Période d'analyse

date_from (query) - Date de début (YYYY-MM-DD)

date_to (query) - Date de fin (YYYY-MM-DD)

Réponses:

200 - Statistiques avancées récupérées avec succès

403 - Accès non autorisé

POST`/api/admin/blog/categories`

Créer une catégorie de blog

Crée une nouvelle catégorie de blog (Admin uniquement)

Réponses:

201 - Catégorie créée avec succès

422 - Erreurs de validation

403 - Accès non autorisé

PUT`/api/admin/blog/categories/{blogCategory}`

Modifier une catégorie de blog

Met à jour une catégorie de blog existante (Admin uniquement)

Paramètres:

blogCategory (path) *requis* - ID de la catégorie à modifier

Réponses:

200 - Catégorie modifiée avec succès

404 - Catégorie non trouvée

422 - Erreurs de validation

DELETE`/api/admin/blog/categories/{blogCategory}`

Supprimer une catégorie de blog

Supprime une catégorie de blog si elle ne contient pas d'articles (Admin uniquement)

Paramètres:

blogCategory (path) *requis* - ID de la catégorie à supprimer

Réponses:

200 - Catégorie supprimée avec succès

404 - Catégorie non trouvée

422 - Impossible de supprimer (catégorie contient des articles)

PATCH

`/api/admin/blog/categories/{blogCategory}/toggle-status`

Activer/Désactiver une catégorie

Change le statut actif/inactif d'une catégorie de blog (Admin uniquement)

Paramètres:

blogCategory (path) *requis* - ID de la catégorie

Réponses:

200 - Statut modifié avec succès

404 - Catégorie non trouvée

GET

`/api/admin/blog/categories/statistics`

Statistiques des catégories de blog

Récupère les statistiques complètes des catégories de blog (Admin uniquement)

Réponses:

200 - Statistiques récupérées avec succès

403 - Accès non autorisé

POST

`/api/admin/blog/categories/update-order`

Mettre à jour l'ordre des catégories

Met à jour l'ordre d'affichage des catégories de blog (Admin uniquement)

Réponses:

200 - Ordre mis à jour avec succès

422 - Erreurs de validation

403 - Accès non autorisé

GET

`/api/admin/blog/comments`

Liste des commentaires de blog

Récupère la liste des commentaires avec filtrage et modération (Admin uniquement)

Paramètres:

search (query) - Terme de recherche dans le contenu ou auteur

status (query) - Filtrer par statut de modération

post_id (query) - Filtrer par article de blog

user_id (query) - Filtrer par utilisateur

sort_by (query) - Critère de tri

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des commentaires récupérée avec succès

403 - Accès non autorisé

GET

`/api/admin/blog/comments/{blogComment}`

Détails d'un commentaire de blog

Récupère les détails complets d'un commentaire avec signalements (Admin uniquement)

Paramètres:

blogComment (path) *requis* - ID du commentaire

Réponses:

200 - Détails du commentaire récupérés avec succès

404 - Commentaire non trouvé

403 - Accès non autorisé

DELETE

`/api/admin/blog/comments/{blogComment}`

Supprimer un commentaire

Supprime définitivement un commentaire de blog (Admin uniquement)

Paramètres:

blogComment (path) *requis* - ID du commentaire à supprimer

Réponses:

200 - Commentaire supprimé avec succès

403 - Pas d'autorisation pour supprimer ce commentaire

404 - Commentaire non trouvé

POST

`/api/admin/blog/comments/{blogComment}/moderate`

Modérer un commentaire

Approuve, rejette ou marque comme spam un commentaire (Admin uniquement)

Paramètres:

blogComment (path) *requis* - ID du commentaire à modérer

Réponses:

200 - Commentaire modéré avec succès

404 - Commentaire non trouvé

422 - Erreurs de validation

PATCH

`/api/admin/blog/comments/{blogComment}/toggle-pin`

Épingler/Désépingler un commentaire

Épingle ou désépingle un commentaire (le met en évidence) (Admin uniquement)

Paramètres:

blogComment (path) *requis* - ID du commentaire

Réponses:

200 - Statut d'épingleage modifié avec succès

404 - Commentaire non trouvé

403 - Accès non autorisé

GET

`/api/admin/blog/comments/statistics`

Statistiques des commentaires

Récupère les statistiques complètes des commentaires de blog (Admin uniquement)

Réponses:

200 - Statistiques récupérées avec succès

403 - Accès non autorisé

GET

`/api/admin/blog/comment-reports`

Liste des signalements de commentaires

Récupère la liste des signalements de commentaires avec filtrage (Admin uniquement)

Paramètres:

search (query) - Terme de recherche dans le signalement ou commentaire

status (query) - Filtrer par statut

reason (query) - Filtrer par raison du signalement

priority (query) - Filtrer par priorité

high_priority (query) - Afficher uniquement les signalements haute priorité

sort_by (query) - Critère de tri

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des signalements récupérée avec succès

403 - Accès non autorisé

GET `/api/admin/blog/comment-reports/{blogCommentReport}`

Détails d'un signalement

Récupère les détails complets d'un signalement de commentaire (Admin uniquement)

Paramètres:

blogCommentReport (path) *requis* - ID du signalement

Réponses:

200 - Détails du signalement récupérés avec succès

404 - Signalement non trouvé

403 - Accès non autorisé

GET `/api/admin/blog/comment-reports/statistics`

Statistiques des signalements

Récupère les statistiques complètes des signalements de commentaires (Admin uniquement)

Réponses:

200 - Statistiques récupérées avec succès

403 - Accès non autorisé

PATCH `/api/admin/blog/comment-reports/{blogCommentReport}/process`

Traiter un signalement

Traite un signalement en le résolvant ou le rejetant avec action sur le commentaire (Admin uniquement)

Paramètres:

blogCommentReport (path) *requis* - ID du signalement à traiter

Réponses:

200 - Signalement traité avec succès

422 - Erreurs de validation

404 - Signalement non trouvé

POST `/api/admin/blog/posts`

Créer un article de blog

Crée un nouvel article de blog avec toutes ses propriétés

Réponses:

201 - Article créé avec succès

403 - Accès non autorisé

422 - Données de validation invalides

PUT

`/api/admin/blog/posts/{blogPost}`

Modifier un article de blog

Met à jour un article de blog existant (Admin uniquement)

Paramètres:

blogPost (path) *requis* - ID de l'article à modifier

Réponses:

200 - Article modifié avec succès

403 - Accès non autorisé

404 - Article non trouvé

422 - Erreurs de validation

DELETE

`/api/admin/blog/posts/{blogPost}`

Supprimer un article de blog

Supprime définitivement un article de blog (Admin uniquement)

Paramètres:

blogPost (path) *requis* - ID de l'article à supprimer

Réponses:

200 - Article supprimé avec succès

403 - Accès non autorisé

404 - Article non trouvé

POST

`/api/admin/blog/posts/{blogPost}/publish`

Publier un article de blog

Publie un article de blog (passe le statut à 'published') (Admin uniquement)

Paramètres:

blogPost (path) *requis* - ID de l'article à publier

Réponses:

200 - Article publié avec succès

403 - Accès non autorisé

404 - Article non trouvé

POST

`/api/admin/blog/posts/{blogPost}/unpublish`

Dépublier un article de blog

Dépublie un article de blog (passe le statut à 'draft') (Admin uniquement)

Paramètres:

blogPost (path) *requis* - ID de l'article à dépublier

Réponses:

200 - Article dépublié avec succès

403 - Accès non autorisé

404 - Article non trouvé

POST

`/api/admin/blog/posts/{blogPost}/schedule`

Programmer la publication d'un article

Programme la publication automatique d'un article à une date donnée (Admin uniquement)

Paramètres:

blogPost (path) *requis* - ID de l'article à programmer

Réponses:

200 - Article programmé avec succès

403 - Accès non autorisé

422 - Erreurs de validation

GET

`/api/admin/blog/posts/statistics`

Statistiques des articles de blog

Récupère les statistiques complètes des articles de blog (Admin uniquement)

Réponses:

200 - Statistiques récupérées avec succès

403 - Accès non autorisé

GET

`/api/admin/carts/stats`

Statistiques des paniers

Récupère les statistiques détaillées des paniers pour l'administration

Réponses:

200 - Statistiques des paniers récupérées avec succès

403 - Accès refusé

401 - Non authentifié

GET

`/api/admin/carts`

Liste des paniers avec filtres

Récupère une liste paginée de tous les paniers avec options de filtrage

Paramètres:

status (query) - Filtrer par statut du panier

user_id (query) - Filtrer par ID utilisateur

min_total (query) - Montant minimum du panier

max_total (query) - Montant maximum du panier

date_from (query) - Date de création à partir de

date_to (query) - Date de création jusqu'à

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des paniers récupérée avec succès

403 - Accès refusé

401 - Non authentifié

POST

`/api/admin/categories`

Créer une nouvelle catégorie

Crée une nouvelle catégorie. Accès réservé aux administrateurs.

Réponses:

201 - Catégorie créée avec succès

403 - Accès refusé

422 - Erreurs de validation

POST

`/api/admin/rentals/{id}/start-inspection`

Démarrer l'inspection d'une location

Initie le processus d'inspection d'un matériel retourné (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la commande de location

Réponses:

200 - Inspection démarrée avec succès

400 - Commande pas prête pour inspection

403 - Accès non autorisé

POST

`/api/admin/rentals/{id}/finish-inspection`

Finaliser l'inspection d'une location

Termine le processus d'inspection avec évaluation des conditions des équipements retournés (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la commande de location

Réponses:

200 - Inspection finalisée avec succès

400 - Commande pas en cours d'inspection ou données invalides

403 - Accès non autorisé

GET

`/api/admin/returns`

Liste des demandes de retour

Récupère la liste complète des demandes de retour avec système d'inspection (Admin uniquement)

Paramètres:

search (query) - Recherche par numéro de retour, commande, utilisateur

status (query) - Filtrer par statut

return_type (query) - Filtrer par type de retour

date_from (query) - Date de début (YYYY-MM-DD)

date_to (query) - Date de fin (YYYY-MM-DD)

sort_by (query) - Critère de tri

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des retours récupérée avec succès

403 - Accès non autorisé

GET

`/api/admin/returns/{id}`

Détails d'une demande de retour

Affiche les détails complets d'une demande de retour avec informations d'inspection (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la demande de retour

Réponses:

200 - Détails de la demande de retour récupérés avec succès

404 - Demande de retour non trouvée

403 - Accès non autorisé

POST

`/api/admin/returns/{id}/approve`

Approuver une demande de retour

Approuve une demande de retour et initie le processus d'inspection (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la demande de retour

Réponses:

200 - Demande de retour approuvée avec succès

422 - Demande ne peut pas être approuvée

403 - Accès non autorisé

POST

`/api/admin/returns/{id}/reject`

Rejeter une demande de retour

Rejette une demande de retour avec justification (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la demande de retour

Réponses:

200 - Demande de retour rejetée avec succès

422 - Demande ne peut pas être rejetée ou données invalides

403 - Accès non autorisé

POST`/api/admin/returns/{id}/mark-received`

Marquer un retour comme reçu

Marque un retour comme reçu et évalue l'état du produit pour finaliser le remboursement (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la demande de retour

Réponses:

200 - Retour marqué comme reçu avec succès

422 - Retour n'est pas dans le bon statut ou données invalides

403 - Accès non autorisé

POST`/api/admin/returns/{id}/process-refund`

Finaliser le remboursement

Traite le remboursement après réception et inspection du produit retourné (Admin uniquement)

Paramètres:

id (path) *requis* - ID de la demande de retour

Réponses:

200 - Remboursement traité avec succès

422 - Retour n'est pas prêt pour le remboursement ou données invalides

403 - Accès non autorisé

GET`/api/admin/returns/stats`

Statistiques des retours

Récupère les statistiques complètes du système de retours et d'inspection (Admin uniquement)

Paramètres:

period (query) - Période d'analyse en jours

date_from (query) - Date de début (YYYY-MM-DD)

date_to (query) - Date de fin (YYYY-MM-DD)

Réponses:

200 - Statistiques des retours récupérées avec succès

403 - Accès non autorisé

GET`/api/admin/products`

Liste des produits (Admin)

Récupère tous les produits avec statut et filtres pour l'administration

Paramètres:

status (query) - Filtrer par statut du produit

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des produits récupérée avec succès

403 - Accès refusé

401 - Non authentifié

POST

`/api/admin/products`

Créer un produit

Crée un nouveau produit avec toutes ses propriétés

Réponses:

201 - Produit créé avec succès

422 - Données de validation invalides

403 - Accès refusé

GET

`/api/admin/products/{product}`

Détails d'un produit (Admin)

Récupère les détails complets d'un produit pour l'administration

Paramètres:

product (path) *requis* - ID du produit

Réponses:

200 - Détails du produit récupérés avec succès

404 - Produit non trouvé

403 - Accès refusé

PUT

`/api/admin/products/{product}`

Mettre à jour un produit

Met à jour les informations d'un produit existant

Paramètres:

product (path) *requis* - ID du produit

Réponses:

200 - Produit mis à jour avec succès

404 - Produit non trouvé

422 - Données de validation invalides

403 - Accès refusé

DELETE

`/api/admin/products/{product}`

Supprimer un produit

Supprime un produit (soft delete)

Paramètres:

product (path) *requis* - ID du produit

Réponses:

200 - Produit supprimé avec succès

404 - Produit non trouvé

403 - Accès refusé

POST

`/api/admin/products/{id}/restore`

Restaurer un produit

Restaure un produit supprimé

Paramètres:

id (path) *requis* - ID du produit supprimé

Réponses:

200 - Produit restauré avec succès

404 - Produit non trouvé

403 - Accès refusé

PATCH

`/api/admin/products/{product}/toggle-status`

Basculer le statut du produit

Active ou désactive un produit

Paramètres:

product (path) *requis* - ID du produit

Réponses:

200 - Statut du produit modifié avec succès

404 - Produit non trouvé

403 - Accès refusé

GET`/api/admin/products/stock/alerts`

Alertes de stock

Récupère les alertes de stock pour le dashboard admin

Réponses:

200 - Alertes de stock récupérées avec succès

403 - Accès refusé

GET`/api/admin/products/stock/dashboard`

Dashboard de stock

Récupère les statistiques globales de stock pour le dashboard admin

Réponses:

200 - Statistiques de stock récupérées avec succès

403 - Accès refusé

GET`/api/admin/users`

Liste des utilisateurs

Récupère la liste de tous les utilisateurs (y compris supprimés). Accès administrateur requis.

Paramètres:

page (query) - Numéro de page

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des utilisateurs récupérée avec succès

403 - Accès refusé

GET

`/api/admin/users/{user}`

Détails d'un utilisateur

Récupère les détails d'un utilisateur spécifique. Accès administrateur requis.

Paramètres:

user (path) *requis* - ID de l'utilisateur

Réponses:

200 - Détails de l'utilisateur récupérés avec succès

404 - Utilisateur non trouvé

403 - Accès refusé

PUT

`/api/admin/users/{user}`

Mettre à jour un utilisateur

Met à jour les informations d'un utilisateur spécifique. Accès administrateur requis.

Paramètres:

user (path) *requis* - ID de l'utilisateur

Réponses:

200 - Utilisateur mis à jour avec succès

404 - Utilisateur non trouvé

403 - Accès refusé

422 - Données de validation invalides

DELETE

`/api/admin/users/{user}`

Supprimer un utilisateur

Supprime un utilisateur (soft delete). Accès administrateur requis.

Paramètres:

user (path) *requis* - ID de l'utilisateur

Réponses:

200 - Utilisateur supprimé avec succès

403 - Accès refusé

404 - Utilisateur non trouvé

POST

`/api/admin/users/{id}/restore`

Restaurer un utilisateur

Restaure un utilisateur supprimé. Accès administrateur requis.

Paramètres:

id (path) *requis* - ID de l'utilisateur

Réponses:

200 - Utilisateur restauré avec succès

403 - Accès refusé

404 - Utilisateur non trouvé

Blog

GET

`/api/blog/categories`

Liste des catégories de blog

Récupère la liste des catégories de blog avec possibilité de filtrage et tri

Paramètres:

search (query) - Terme de recherche dans le nom ou slug

status (query) - Filtrer par statut (admin uniquement)

sort_by (query) - Critère de tri

sort_direction (query) - Direction du tri

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des catégories récupérée avec succès

GET

`/api/blog/categories/{slug}`

Détails d'une catégorie de blog

Récupère les détails d'une catégorie avec ses articles

Paramètres:

slug (path) *requis* - Slug de la catégorie

Réponses:

200 - Catégorie récupérée avec succès

404 - Catégorie non trouvée

GET

`/api/blog/posts/{blogPost}/comments`

Commentaires d'un article de blog

Récupère les commentaires approuvés d'un article de blog avec les réponses

Paramètres:

blogPost (path) *requis* - ID de l'article de blog

page (query) - Numéro de page

Réponses:

200 - Commentaires récupérés avec succès

404 - Article non trouvé

POST

`/api/blog/comments/{blogComment}/report`

Signaler un commentaire

Permet à un utilisateur de signaler un commentaire inapproprié

Paramètres:

blogComment (path) *requis* - ID du commentaire à signaler

Réponses:

201 - Signalement créé avec succès

409 - Commentaire déjà signalé par cet utilisateur

422 - Erreurs de validation

GET

`/api/blog/posts`

Liste des articles de blog

Récupère la liste des articles de blog publiés avec filtres et pagination

Paramètres:

search (query) - Recherche dans le titre et contenu

category (query) - Slug de la catégorie

author (query) - ID de l'auteur

tag (query) - Tag pour filtrer

sort (query) - Tri des résultats

per_page (query) - Nombre d'articles par page

Réponses:

200 - Liste des articles récupérée avec succès

GET`/api/blog/posts/{slug}`

Détails d'un article de blog

Récupère les détails complets d'un article de blog par son slug

Paramètres:

slug (path) *requis* - Slug de l'article

Réponses:

200 - Article récupéré avec succès

404 - Article non trouvé

Cart

GET`/api/cart`

Récupérer le panier de l'utilisateur

Affiche le contenu du panier de l'utilisateur connecté avec vérification de disponibilité

Réponses:

200 - Panier récupéré avec succès

401 - Non authentifié

POST`/api/cart/products/{product}`

Ajouter un produit au panier

Ajoute un produit au panier de l'utilisateur connecté avec vérification de stock

Paramètres:

product (path) *requis* - ID du produit à ajouter

Réponses:

201 - Produit ajouté au panier avec succès

400 - Stock insuffisant

401 - Non authentifié

422 - Données invalides

PUT

`/api/cart/update-quantity`

Mettre à jour la quantité d'un produit

Met à jour la quantité d'un produit spécifique dans le panier

Réponses:

200 - Quantité mise à jour avec succès

400 - Stock insuffisant

404 - Produit non trouvé dans le panier

422 - Données invalides

DELETE

`/api/cart/remove/{product}`

Supprimer un produit du panier

Supprime complètement un produit du panier de l'utilisateur

Paramètres:

product (path) *requis* - ID du produit à supprimer

Réponses:

200 - Produit supprimé du panier avec succès

404 - Produit non trouvé dans le panier

401 - Non authentifié

DELETE

`/api/cart/clear`

Vider le panier

Supprime tous les produits du panier de l'utilisateur

Réponses:

200 - Panier vidé avec succès

404 - Aucun panier actif trouvé

401 - Non authentifié

GET

`/api/cart/check-availability`

Vérifier la disponibilité des produits

Vérifie la disponibilité et le stock de tous les produits dans le panier

Réponses:

200 - Vérification de disponibilité effectuée

404 - Aucun panier actif trouvé

401 - Non authentifié

GET `/api/cart/summary`

Résumé du panier

Obtient un résumé détaillé du panier avec totaux et calculs

Réponses:

200 - Résumé du panier récupéré avec succès

401 - Non authentifié

POST `/api/cart/prepare-checkout`

Préparer le panier pour commande

Effectue toutes les vérifications nécessaires avant de procéder au checkout

Réponses:

200 - Panier prêt pour la commande

400 - Le panier n'est pas prêt pour la commande

404 - Aucun panier actif trouvé

401 - Non authentifié

Categories

GET`/api/categories`

Liste des catégories

Récupère la liste des catégories. Les administrateurs voient toutes les catégories, les autres utilisateurs ne voient que les catégories actives.

Paramètres:

page (query) - Numéro de page pour la pagination

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des catégories récupérée avec succès

500 - Erreur serveur

Cookies

GET`/api/cookies/preferences`

Obtenir les préférences de cookies

Récupère les préférences de cookies de l'utilisateur ou visiteur avec statut de consentement

Paramètres:

force_consent (query) - Forcer l'affichage du consentement

X-Force-Cookie-Consent (header) - Header pour forcer le consentement

Réponses:

200 - Préférences récupérées avec succès

POST

`/api/cookies/preferences`

Mettre à jour les préférences de cookies

Met à jour les préférences de consentement aux cookies pour l'utilisateur ou visiteur

Réponses:

200 - Préférences mises à jour avec succès

422 - Erreurs de validation

POST

`/api/cookies/accept-all`

Accepter tous les cookies

Accepte tous les types de cookies en une seule action

Réponses:

200 - Tous les cookies acceptés avec succès

POST

`/api/cookies/reject-all`

Rejeter tous les cookies optionnels

Rejette tous les cookies non-essentiels, garde uniquement les cookies nécessaires

Réponses:

200 - Cookies optionnels rejetés avec succès

Orders

GET

`/api/orders`

Liste des commandes utilisateur

Récupère la liste des commandes de l'utilisateur connecté avec filtres et tri

Paramètres:

status (query) - Filtrer par statut de commande

sort_by (query) - Trier les résultats

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des commandes récupérée avec succès

401 - Non authentifié

POST

`/api/orders`

Créer une nouvelle commande

Crée une nouvelle commande à partir du panier actuel avec paiement

Réponses:

201 - Commande créée avec succès

400 - Erreur de validation ou panier vide

422 - Données de validation invalides

401 - Non authentifié

GET

`/api/orders/{order}`

Détails d'une commande

Récupère les détails complets d'une commande spécifique

Paramètres:

order (path) *requis* - ID de la commande

Réponses:

200 - Détails de la commande récupérés avec succès

403 - Accès non autorisé à cette commande

404 - Commande non trouvée

401 - Non authentifié

POST

`/api/orders/{order}/cancel`

Annuler une commande

Annule une commande si elle est éligible à l'annulation

Paramètres:

order (path) *requis* - ID de la commande

Réponses:

200 - Commande annulée avec succès

403 - Accès non autorisé

422 - Commande ne peut plus être annulée

404 - Commande non trouvée

401 - Non authentifié

Products

GET

`/api/products`

Récupérer la liste des produits

Récupère une liste paginée de produits avec possibilité de filtrage et tri

Paramètres:

category_id (query) - ID de la catégorie pour filtrer

type (query) - Type de produit (sale, rental, both)

stock_status (query) - Statut du stock

search (query) - Terme de recherche

sort_by (query) - Critère de tri

sort_direction (query) - Direction du tri

per_page (query) - Nombre d'éléments par page

Réponses:

200 - Liste des produits récupérée avec succès

422 - Erreur de validation

GET

`/api/products-likes`

Produits les plus likés

Récupère la liste des produits triés par nombre de likes (accessible à tous)

Paramètres:

per_page (query) - Nombre de produits par page

page (query) - Numéro de page

Réponses:

200 - Produits récupérés avec succès

GET

`/api/products/{product}/likes`

Likes d'un produit

Récupère la liste des likes d'un produit spécifique avec détails des utilisateurs

Paramètres:

product (path) *requis* - ID du produit

per_page (query) - Nombre de likes par page

Réponses:

200 - Likes récupérés avec succès

404 - Produit non disponible

Users

GET

`/api/profile`

Profil utilisateur

Récupère les détails du profil de l'utilisateur connecté

Réponses:

200 - Profil utilisateur récupéré avec succès

401 - Non authentifié

PUT

`/api/profile`

Mettre à jour le profil

Met à jour le profil de l'utilisateur connecté

Réponses:

200 - Profil mis à jour avec succès

401 - Non authentifié

422 - Données de validation invalides

GET`/api/profile/download-data`

Télécharger les données utilisateur

Télécharge toutes les données utilisateur en format ZIP avec PDF (conformité RGPD)

Réponses:

200 - Fichier ZIP contenant les données utilisateur

401 - Non authentifié

500 - Erreur lors de la génération du fichier

Newsletter

POST`/api/newsletter/subscribe`

S'abonner à la newsletter

Abonne l'utilisateur connecté à la newsletter

Réponses:

200 - Abonnement à la newsletter réussi

401 - Non authentifié

POST`/api/newsletter/unsubscribe`

Se désabonner de la newsletter

Désabonne l'utilisateur connecté de la newsletter

Réponses:

200 - Désabonnement de la newsletter réussi

401 - Non authentifié

Messages

POST

`/api/messages/{message}/mark-read`

Marquer un message comme lu

Marque un message comme lu pour l'utilisateur connecté

Paramètres:

message (path) *requis* - ID du message

Réponses:

200 - Message marqué comme lu

403 - Accès refusé

404 - Message non trouvé

POST

`/api/messages/{message}/archive`

Archiver un message

Archive un message pour l'utilisateur connecté

Paramètres:

message (path) *requis* - ID du message

Réponses:

200 - Message archivé

403 - Accès refusé

404 - Message non trouvé

DELETE

`/api/messages/{message}`

Supprimer un message

Supprime un message pour l'utilisateur connecté

Paramètres:

message (path) *requis* - ID du message

Réponses:

200 - Message supprimé

403 - Accès refusé

404 - Message non trouvé