

Websource Social Login : guide utilisateur

Version 1.0.0 : module PrestaShop gratuit, licence AFL-3.0, par [Websource](#).

1. Présentation

Websource Social Login permet à vos clients de se connecter et de créer leur compte en un clic avec **Google, Facebook, X (Twitter), LinkedIn, Microsoft, Apple et GitHub**.

- Connexion **et inscription automatique** : le compte client est créé à partir du profil du réseau (identifiant, prénom, nom, e-mail).
- Aucune bibliothèque externe : cURL et PHP natif (OAuth 2.0 / OpenID Connect, PKCE, validation de l'id_token).
- **Anti-prise de compte** : jamais de fusion automatique de comptes ; un e-mail non vérifié par le réseau est confirmé par lien.
- Un écran de configuration par réseau avec un **guide pas à pas intégré** (français et anglais), l'URL de rappel à copier et un test de configuration.
- Boutons personnalisables (3 styles, grille d'icônes, couleurs, formes) et placés où vous voulez : connexion, inscription, tunnel de commande, en-tête, liste de souhaits.
- RGPD : données minimales, export et suppression avec le compte client.

Connexion avec les réseaux sociaux

Connexion et inscription en un clic : Google, Facebook, X, LinkedIn, Microsoft, Apple, GitHub.

Boutique : **Ma boutique**

Réglages enregistrés.

Réseaux Apparence et emplacements Textes et RGPD Aide

Réseaux sociaux

Pour chaque réseau : créez l'application chez lui en suivant le guide, collez l'URL de rappel indiquée, puis recopiez l'identifiant et le secret. Un réseau n'apparaît côté boutique que s'il est activé ET complet.

**Google** Non configuré

Client ID *

Client Secret *

Collez le secret ici

Scopes demandés

openid email profile

Par défaut : openid email profile

Position (1 = premier)

1

URL de rappel à coller chez le réseau

https://ma-boutique.example/fr/module/websourcesociallogin/callback

Copier

Exacte, en https, pour cette boutique. À recopier telle quelle.

PKCE S256 activé OpenID Connect (id_token vérifié)

Tester la configuration

✖ Champ renseigné : Client ID

✖ Champ renseigné : Client Secret

✔ URL de rappel en https

✔ Scopes définis

✖ Réseau activé

**Facebook** Non configuré

Comment se passe une connexion

1. Le client clique sur « Continuer avec Google » (ou un autre réseau).
2. Au retour du réseau :
 - compte social déjà lié : le client est **connecté** ;
 - e-mail **vérifié** par le réseau et client existant : le client **saisit son mot de passe** pour lier les deux comptes (jamais de

fusion automatique) ;

- e-mail vérifié et inconnu : le **client est créé** (mot de passe aléatoire fort, groupe par défaut, newsletter et offres partenaires non cochées) puis connecté ;
- e-mail **non vérifié** (ex. Microsoft) ou **absent** (ex. X) : le client saisit son e-mail et reçoit un **lien de confirmation** (valable 1 heure, à ouvrir dans le même navigateur).

3. Dans « Mon compte > Connexions sociales », le client lie ou délie un réseau.

2. Prérequis

Élément	Exigence
PrestaShop	8.x (compatible à partir de 1.7.7)
PHP	7.3 ou supérieur (extensions <code>curl</code> , <code>json</code> , <code>openssl</code>)
HTTPS	obligatoire : les réseaux refusent une URL de rappel en <code>http://</code> (Paramètres de la boutique > Trafic et SEO > Activer le SSL sur toutes les pages)
Comptes développeur	un compte gratuit chez chaque réseau que vous activez
Pare-feu	votre serveur doit pouvoir joindre les réseaux en HTTPS sortant (cURL)

Vous n'avez pas besoin d'activer les sept réseaux : activez ceux que vos clients utilisent.

3. Installation

Depuis le zip (recommandé)

1. Téléchargez `websourcesociallogin-1.0.0.zip` (page du module sur webservice.fr ou page « Releases » du dépôt GitHub). Vérifiez au besoin l'empreinte SHA-256 publiée avec le fichier : `sha256sum websourcesociallogin-1.0.0.zip` .
2. Back-office > **Modules** > **Gestionnaire de modules** > **Installer un module**, choisissez le zip.
3. Cliquez sur **Configurer** une fois l'installation terminée.

Depuis GitHub

```
cd modules/  
git clone https://github.com/Webservice-fr/websourcesociallogin.git  
cd ..  
php bin/console prestashop:module install websourcesociallogin
```

Le dépôt contient aussi `docs/` et `tests/` (inutiles en production). Le dossier `vendor/` est fourni : aucune commande Composer n'est nécessaire.

Mise à jour

Remplacez le dossier du module (ou ré-importez le zip de la nouvelle version) : PrestaShop exécute les scripts de mise à jour. Vos réglages sont conservés.

4. Configuration générale

Back-office > Modules > Webservice Social Login > **Configurer**. L'écran est organisé en quatre parties : **Réseaux**, **Apparence et**

emplacements, Textes et RGPD, Aide. Les réglages sont **par boutique** (multiboutique : choisissez le contexte de boutique).

Pour chaque réseau (voir la section 5) :

Champ	Rôle
Activer	interrupteur du réseau
Client ID / Client Secret	fournis par le réseau ; le secret est chiffré en base et jamais renvoyé dans la page (« •••• enregistré ») ; laissé vide, il est conservé
Scopes demandés	autorisations demandées ; la valeur par défaut convient dans presque tous les cas
Position	ordre d'affichage (1 = premier)
URL de rappel	adresse exacte à coller chez le réseau, avec bouton Copier
Tester la configuration	vérifie les champs renseignés, l'URL de rappel en https, les scopes et l'activation
État	<i>Actif, Configuré, Incomplet, Non configuré</i>

Un réseau n'apparaît côté boutique que s'il est activé ET complet.

réseaux

Apparence et emplacements

Textes et RGPD

Aide

Client ID *

Client Secret *

Collez le secret ici

Scopes demandés

openid email profile

Par défaut : openid email profile

Position (1 = premier)

1

URL de rappel à coller chez le réseau

https://ma-boutique.example/fr/module/websourcesociallogin/callback

Copier

Exacte, en https, pour cette boutique. À recopier telle quelle.

PKCE S256 activé

OpenID Connect (id_token vérifié)

Tester la configuration

✖ Champ renseigné : Client ID

✖ Champ renseigné : Client Secret

✓ URL de rappel en https

✓ Scopes définis

✖ Réseau activé

▼ Guide pas à pas : Google

Google utilise OpenID Connect : le module valide la signature de l'id_token et n'accepte l'e-mail que s'il est vérifié par Google. Comptez environ 10 minutes. Aucun coût.

Ouvrir la console : Google Cloud Console

1

Créer (ou choisir) un projet Google Cloud

console.cloud.google.com > sélecteur de projet (en haut) > Nouveau projet

Connectez-vous avec le compte Google de votre entreprise. Cliquez sur le sélecteur de projet, puis **Nouveau projet**. Donnez-lui le nom de votre boutique et validez.

2

Configurer l'écran de consentement OAuth

3

Déclarer les autorisations (scopes)

Websource Social Login 1.0.0 - 3/19

4 Créer l'ID client OAuth

▸ Menu ≡ > API et services > Identifiants > + Créer des identifiants > ID client OAuth

Type d'application : **Application Web**. Dans **URI de redirection autorisés**, collez exactement l'adresse ci-dessous (https, sans espace, sans « / » ajouté).

Type d'application	Application Web
Nom	Connexion boutique
Origines JavaScript autorisées	laisser vide (inutile)
URI de redirection autorisés	<code>https://ma-boutique.example/fr/module/websourcesociallogin/callback</code> Copier

5 Copier l'ID client et le code secret dans le module

6 Passer en production

Scopes demandés : `openid email profile` . Le module reçoit : identifiant Google (sub), e-mail, état de vérification de l'e-mail, prénom, nom. Rien d'autre.

Erreurs fréquentes et solutions

Erreur 400 : `redirect_uri_mismatch`

L'URI de redirection saisie dans Google n'est pas strictement identique à

`https://ma-boutique.example/fr/module/websourcesociallogin/callback` Copier . Vérifiez https, le nom de domaine (avec ou sans www), l'absence de « / » final en trop. La modification peut mettre quelques minutes à être prise en compte.

Erreur 403 : `access_denied` / « L'application n'a pas terminé le processus de validation »

L'application est encore en mode Test et votre compte n'est pas dans les utilisateurs tests : ajoutez-le, ou publiez l'application (étape 6).

Erreur 401 : `invalid_client`

L'ID client ou le code secret est incorrect ou a été régénéré. Recollez-les depuis la page Identifiants.

« Cette application n'est pas validée par Google »

Normal si vous demandez des scopes sensibles. Limitez-vous aux trois scopes de l'étape 3.

Erreur 400 : `invalid_request` (« Les règles de Google interdisent... »)

Une URI de redirection n'est pas en https, ou le domaine n'est pas déclaré dans l'écran de consentement.

Liens officiels : [Console des identifiants Google](#) · [Documentation OpenID Connect de Google](#)

5. Guides par réseau

Principe commun : (1) créer une application chez le réseau, (2) y enregistrer l'**URL de rappel** affichée par le module (elle est propre à votre boutique ; dans les exemples ci-dessous : `https://votre-boutique.example/fr/module/websourcesociallogin/callback`), (3) copier l'identifiant et le secret dans le module, (4) activer, enregistrer, tester, (5) **publier l'application** chez le réseau : tant qu'elle est en mode test ou développement, seuls vos comptes de test peuvent se connecter.

Ces guides reprennent ceux de l'écran de configuration (onglet Réseaux, « Guide pas à pas »), qui affichent directement votre URL de rappel avec un bouton Copier. Les interfaces des réseaux évoluent : en cas d'écart, suivez l'intitulé des étapes et les liens officiels.

1. Google

Google utilise OpenID Connect : le module valide la signature de l'id_token et n'accepte l'e-mail que s'il est vérifié par Google. Comptez environ 10 minutes. Aucun coût.

- **Console** : [Google Cloud Console](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example/fr/module/websourcesociallogin/callback`

Étape 1. Créer (ou choisir) un projet Google Cloud

Où : *console.cloud.google.com* > sélecteur de projet (en haut) > Nouveau projet

Connectez-vous avec le compte Google de votre entreprise. Cliquez sur le sélecteur de projet, puis **Nouveau projet**. Donnez-lui le nom de votre boutique et validez.

Étape 2. Configurer l'écran de consentement OAuth

Où : Menu ☰ > API et services > Écran de consentement OAuth (ou « Google Auth Platform > Branding »)

C'est la fenêtre que vos clients voient en se connectant. Choisissez le type d'utilisateur **Externe**.

Champs à renseigner	
Nom de l'application	le nom de votre boutique
E-mail d'assistance utilisateur	une adresse que vous lisez
Domaines autorisés	votre-boutique.example
Lien vers les règles de confidentialité	https://votre-boutique.example/ (votre page « politique de confidentialité »)
E-mail du développeur	votre e-mail

Étape 3. Déclarer les autorisations (scopes)

Où : Écran de consentement > Accès aux données (Scopes) > Ajouter ou supprimer des niveaux d'accès

Ajoutez uniquement les trois niveaux d'accès « non sensibles » : openid , .../auth/userinfo.email et .../auth/userinfo.profile . Ils ne déclenchent pas de vérification approfondie par Google.

Étape 4. Créer l'ID client OAuth

Où : Menu ☰ > API et services > Identifiants > + Créer des identifiants > ID client OAuth

Type d'application : **Application Web**. Dans **URI de redirection autorisés**, collez exactement l'adresse ci-dessous (https, sans espace, sans « / » ajouté).

Champs à renseigner	
Type d'application	Application Web
Nom	Connexion boutique
Origines JavaScript autorisées	laisser vide (inutile)
URI de redirection autorisés	https://votre-boutique.example/fr/module/websourcesociallogin/callback

Étape 5. Copier l'ID client et le code secret dans le module

Où : Fenêtre « Client OAuth créé » > ID client + Code secret du client

Copiez l'**ID client** (se termine par .apps.googleusercontent.com) et le **Code secret du client**, collez-les dans les champs de ce module, activez Google puis enregistrez. Le secret n'est affiché en clair nulle part ensuite.

Étape 6. Passer en production

Où : Écran de consentement > État de publication > Publier l'application

Tant que l'application est en **Test**, seuls les « utilisateurs tests » que vous listez (100 maximum) peuvent se connecter, et leur autorisation expire au bout de 7 jours. Cliquez sur **Publier l'application** (statut « En production »). Avec uniquement les scopes non sensibles ci-dessus, aucune vérification par Google n'est nécessaire.

Autorisations (scopes) : Scopes demandés : ```. Le module reçoit : identifiant Google (sub), e-mail, état de vérification de l'e-mail, prénom, nom. Rien d'autre.

Erreurs fréquentes

- **Erreur 400 : `redirect_uri_mismatch`** : L'URI de redirection saisie dans Google n'est pas strictement identique à `https://votre-boutique.example.fr/module/websourcesociallogin/callback`. Vérifiez `https`, le nom de domaine (avec ou sans `www`), l'absence de `< / >` final en trop. La modification peut mettre quelques minutes à être prise en compte.
- **Erreur 403 : `access_denied` / « L'application n'a pas terminé le processus de validation »** : L'application est encore en mode Test et votre compte n'est pas dans les utilisateurs tests : ajoutez-le, ou publiez l'application (étape 6).
- **Erreur 401 : `invalid_client`** : L'ID client ou le code secret est incorrect ou a été régénéré. Recollez-les depuis la page Identifiants.
- **« Cette application n'est pas validée par Google »** : Normal si vous demandez des scopes sensibles. Limitez-vous aux trois scopes de l'étape 3.
- **Erreur 400 : `invalid_request` (« Les règles de Google interdisent... »)** : Une URI de redirection n'est pas en `https`, ou le domaine n'est pas déclaré dans l'écran de consentement.

Liens officiels : [Console des identifiants Google](#) · [Documentation OpenID Connect de Google](#)

2. Facebook

Facebook Login (Meta). Les permissions `public_profile` et `email` sont accordées par défaut, sans passer par la revue d'application. Comptez environ 15 minutes. Gratuit.

- **Console :** [Meta for Developers](#)
- **URL de rappel (callback) à enregistrer :** `https://votre-boutique.example.fr/module/websourcesociallogin/callback`

Étape 1. Créer une app Meta

Où : *developers.facebook.com > Mes apps > Créer une app*

Connectez-vous avec un compte Facebook (devenu « compte développeur »). Cliquez sur **Créer une app**, saisissez le nom de votre boutique et votre e-mail. Pour le cas d'usage, choisissez **Authentifier et demander des données aux utilisateurs avec Facebook Login**. Type d'app : « Consommateur » ou « Autre », selon l'assistant.

Étape 2. Renseigner les informations de base

Où : *Paramètres de l'app > Général*

Renseignez ce qui est obligatoire pour passer en mode Live, puis copiez **ID de l'app** et la **Clé secrète** (bouton « Afficher », votre mot de passe Facebook est demandé).

Champs à renseigner	
Domaines de l'app	votre-boutique.example
URL de la politique de confidentialité	votre page de politique de confidentialité (obligatoire)
URL des conditions d'utilisation	votre page de CGV (recommandé)
Icône de l'app (1024×1024)	votre logo
Catégorie	Achats / Commerce

Étape 3. Configurer Facebook Login

Où : *Tableau de bord > Cas d'usage > Facebook Login > Personnaliser > Paramètres (ou Produits > Facebook Login > Paramètres)*

Activez **Connexion OAuth client** et **Connexion OAuth Web, Forcer HTTPS** et le **mode strict** pour les URI de redirection, puis

collez l'adresse ci-dessous.

Champs à renseigner	
URI de redirection OAuth valides	<code>https://votre-boutique.example/fr/module/websourcesociallogin/callback</code>

Étape 4. Vérifier les permissions

Où : *Cas d'usage > Facebook Login > Permissions et fonctionnalités*

Assurez-vous que `public_profile` et `email` sont sur « Accès standard » (ils le sont par défaut). Aucune revue d'application n'est nécessaire pour ces deux-là.

Étape 5. Coller l'ID et la clé secrète dans le module

Où : *Ce module > Facebook*

ID de l'app = « Client ID », Clé secrète = « Client Secret ». Activez Facebook et enregistrez.

Étape 6. Passer l'app en mode Live

Où : *Bandeau en haut du tableau de bord > « Mode de l'app » : Développement → Live*

En mode **Développement**, seuls les administrateurs, développeurs et testeurs de l'app peuvent se connecter. Basculez sur **Live** une fois la politique de confidentialité et la catégorie renseignées. La « vérification de l'entreprise » n'est demandée que pour des accès avancés, pas pour `public_profile` + `email`.

Autorisations (scopes) : Permissions demandées : ```. Le module reçoit : identifiant Facebook (id), prénom, nom et e-mail s'il existe. Facebook ne renvoie que des e-mails confirmés.

Erreurs fréquentes

- « **URL bloquée : la redirection a échoué** » / **redirect_uri is not whitelisted** : L'URI saisie dans « URI de redirection OAuth valides » diffère de `https://votre-boutique.example/fr/module/websourcesociallogin/callback`. Copiez-la avec le bouton, activez le mode strict et réessayez.
- « **Fonctionnalité indisponible** » / « **L'app n'est pas active** » : L'app est en mode Développement et votre compte n'a pas de rôle dessus. Ajoutez-le dans Rôles de l'app, ou passez en Live (étape 6).
- « **Can't load URL : le domaine de cette URL n'est pas inclus dans les domaines de l'app** » : Ajoutez votre-boutique.example dans Paramètres > Général > Domaines de l'app.
- **Aucun e-mail reçu** : Le client n'a pas d'e-mail confirmé sur Facebook, ou il a décoché l'autorisation. Le module lui demande alors de saisir son e-mail, qu'il confirme par un lien.
- **Invalid client_secret / Error validating client secret** : La clé secrète a été régénérée ou mal copiée : recollez-la.

Liens officiels : [Mes apps Meta](#) · [Documentation Facebook Login](#)

3. X (Twitter)

X (anciennement Twitter) : OAuth 2.0 avec PKCE, application de type « client confidentiel ». L'e-mail n'est fourni que si votre application y est autorisée (scope `users.email`) ; sinon le module le demandera au client. Les offres et quotas de l'API X évoluent : vérifiez-les sur le portail.

- **Console** : [X Developer Portal](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example/fr/module/websourcesociallogin/callback`

Étape 1. Ouvrir un compte développeur X

Où : *developer.x.com > Developer Portal*

Connectez-vous avec le compte X de votre entreprise et acceptez le contrat développeur (décrivez l'usage : « connexion des clients de ma boutique avec leur compte X »).

Étape 2. Créer un Project et une App

Où : *Developer Portal > Projects & Apps > + Add Project (puis Add App)*

L'application doit obligatoirement être rattachée à un **Project** (sinon l'API répond « client-not-enrolled »).

Étape 3. Activer l'authentification utilisateur (OAuth 2.0)

Où : *Votre App > Settings > User authentication settings > Set up*

C'est ici que se règle la connexion.

Champs à renseigner	
App permissions	Read (lecture seule suffit)
Request email from users	à cocher si disponible (nécessite les deux URL ci-dessous)
Type of App	Web App, Automated App or Bot (= Confidential client)
Callback URI / Redirect URL	<code>https://votre-boutique.example/fr/module/websourcesociallogin/callback</code>
Website URL	<code>https://votre-boutique.example</code>
Terms of service / Privacy policy	URL de vos CGV et de votre politique de confidentialité

Étape 4. Récupérer le Client ID et le Client Secret

Où : *Votre App > Keys and tokens > OAuth 2.0 Client ID and Client Secret*

Cliquez sur **Generate** (ou Regenerate). Le **Client Secret n'est affiché qu'une seule fois** : copiez-le immédiatement. N'utilisez pas « API Key / API Secret » ni le « Bearer Token » : ce sont d'autres mécanismes (OAuth 1.0a).

Étape 5. Choisir les scopes

Où : *Ce module > X > Scopes*

Par défaut : `users.read tweet.read users.email`. Si X répond « invalid_scope », supprimez `users.email` : la connexion fonctionnera, et le module demandera l'e-mail au client.

Étape 6. Coller les identifiants dans le module

Où : *Ce module > X*

Client ID, Client Secret, activez X puis enregistrez.

Étape 7. Mise en production

Où : *Developer Portal > votre App > Settings*

X n'a pas de mode « développement / live » : l'application est utilisable dès sa création. Vérifiez cependant le niveau d'accès (Free, Basic...) et ses quotas de requêtes : une boutique à fort trafic peut dépasser les limites d'un niveau gratuit.

Autorisations (scopes) : Scopes demandés : ```. Le module reçoit : identifiant X (id), nom affiché et, si autorisé, l'e-mail confirmé (`confirmed_email`).

Erreurs fréquentes

- « **Something went wrong. You weren't able to give access to the App** » : L'URL de rappel enregistrée dans X diffère de `https://votre-boutique.example/fr/module/websourcesociallogin/callback` (https, domaine, « / » final). Corrigez-la dans User authentication settings.
- **invalid_scope** : Le scope `users.email` n'est pas autorisé pour votre application : retirez-le dans les scopes du module.
- **client-not-enrolled / 403** : L'App n'est pas dans un Project, ou votre niveau d'accès n'inclut pas l'endpoint `/2/users/me`.

- **429 Too Many Requests** : Quota de l'offre X dépassé : patientez ou changez d'offre.
- **Aucun e-mail reçu** : Cas normal si l'e-mail n'est pas autorisé ou n'est pas confirmé sur X : le client saisit son e-mail et le confirme par un lien.

Liens officiels : [X Developer Portal](#) · [Documentation OAuth 2.0 \(X\)](#)

4. LinkedIn

LinkedIn via le produit « Sign In with LinkedIn using OpenID Connect » (OpenID Connect standard). Une Page LinkedIn d'entreprise est obligatoire pour créer l'application. Comptez environ 10 minutes. Gratuit.

- **Console** : [LinkedIn Developer Portal](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example/fr/module/websourcesociallogin/callback`

Étape 1. Créer l'application

Où : [linkedin.com/developers/apps](#) > *Create app*

Renseignez les champs ; la Page LinkedIn doit appartenir à votre entreprise (vous devez en être administrateur).

Champs à renseigner	
App name	le nom de votre boutique
LinkedIn Page	la Page LinkedIn de votre entreprise
Privacy policy URL	votre politique de confidentialité
App logo	votre logo (minimum 100×100 px)

Étape 2. Ajouter le produit « Sign In with LinkedIn using OpenID Connect »

Où : *Votre app* > onglet *Products*

Cliquez sur **Request access** en face de « Sign In with LinkedIn using OpenID Connect » (accès accordé immédiatement). Sans ce produit, LinkedIn répond « `unauthorized_scope_error` ».

Étape 3. Déclarer l'URL de redirection

Où : *Votre app* > onglet *Auth* > *OAuth 2.0 settings* > *Authorized redirect URLs for your app*

Cliquez sur le crayon, **Add redirect URL**, collez l'adresse ci-dessous puis Update.

Champs à renseigner	
Authorized redirect URLs	<code>https://votre-boutique.example/fr/module/websourcesociallogin/callback</code>

Étape 4. Copier le Client ID et le Client Secret

Où : *Votre app* > onglet *Auth* > *Application credentials*

Copiez le **Client ID** et le **Primary Client Secret**, collez-les dans ce module, activez LinkedIn puis enregistrez.

Étape 5. Vérifier l'application

Où : *Votre app* > onglet *Settings* > *Verify*

Cliquez sur **Verify** : LinkedIn génère un lien que l'administrateur de la Page doit approuver. Une application non vérifiée peut afficher un avertissement aux clients. Il n'existe pas de mode « live » séparé : l'app fonctionne dès l'ajout du produit.

Autorisations (scopes) : Scopes demandés : ` (openid, profile, email). Le module reçoit : identifiant LinkedIn (sub), prénom, nom,

e-mail et son état de vérification.

Erreurs fréquentes

- **redirect_uri doesn't match / Bummer, something went wrong** : L'URL enregistrée dans l'onglet Auth diffère de `https://votre-boutique.example.fr/module/websourcesociallogin/callback`. Elle doit être identique au caractère près.
- **unauthorized_scope_error / scope not authorized** : Le produit « Sign In with LinkedIn using OpenID Connect » n'a pas été ajouté (onglet Products) : ajoutez-le.
- **invalid_client_id / invalid_client_secret** : Recopiez le Client ID et le Primary Client Secret depuis l'onglet Auth (pas l'identifiant de la Page).
- **Aucun e-mail (ou e-mail non vérifié)** : Le membre n'a pas d'e-mail confirmé : le module lui demande son e-mail et le confirme par un lien.

Liens officiels : [Mes applications LinkedIn](#) · [Documentation Sign In with LinkedIn \(OpenID Connect\)](#)

5. Microsoft

Microsoft Entra ID (comptes professionnels/scolaires, et comptes personnels Outlook, Hotmail, Xbox). Important : Microsoft ne garantit pas que l'e-mail d'un compte professionnel appartient à son titulaire ; le module ne le considère donc comme vérifié que si Microsoft l'atteste (revendication `xms_edov`) et demande sinon une confirmation par e-mail.

- **Console** : [Microsoft Entra admin center](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example.fr/module/websourcesociallogin/callback`

Étape 1. Inscrire une application

Où : *entra.microsoft.com > Identité > Applications > Inscriptions d'applications > + Nouvelle inscription*

Il faut un compte Azure/Entra (un compte Microsoft gratuit suffit pour créer le locataire).

Champs à renseigner	
Nom	le nom de votre boutique
Types de comptes pris en charge	Comptes dans un annuaire d'organisation et comptes Microsoft personnels (multilocataire et personnels)
URI de redirection - plateforme	Web
URI de redirection	<code>https://votre-boutique.example.fr/module/websourcesociallogin/callback</code>

Étape 2. Noter l'ID d'application (client)

Où : *Votre inscription > Vue d'ensemble*

Copiez l'**ID d'application (client)** : c'est le Client ID. Le champ « Locataire » du module reste sur `common` (comptes professionnels et personnels). Utilisez `consumers` pour les seuls comptes personnels, `organizations` pour les seuls comptes professionnels, ou l'ID de répertoire (locataire) pour un seul annuaire.

Étape 3. Créer un secret client

Où : *Certificats et secrets > Secrets client > + Nouveau secret client*

Choisissez une durée (24 mois maximum), puis copiez immédiatement la colonne **Valeur** (pas l'« ID de secret »). Elle ne sera plus affichée. **Notez la date d'expiration** : à l'échéance, la connexion cessera de fonctionner tant que vous n'aurez pas créé un nouveau secret.

Étape 4. Vérifier les autorisations

Où : *Autorisations de l'API*

Microsoft Graph, autorisations déléguées : `openid` , `profile` , `email` (et `User.Read` par défaut). Aucun consentement administrateur n'est requis pour ces droits.

Étape 5. Ajouter les revendications facultatives (recommandé)

Où : *Configuration des jetons > + Ajouter une revendication facultative > ID*

Ajoutez `email` , `given_name` , `family_name` et, si proposée, `xms_edov` (domaine de l'e-mail vérifié). Acceptez d'activer l'autorisation Graph « email » si Entra le propose.

Étape 6. Marque et mise en production

Où : *Image de marque et propriétés*

Renseignez le logo, les URL des conditions d'utilisation et de la politique de confidentialité, et si possible la **vérification de l'éditeur** (supprime la mention « non vérifié » à l'écran de consentement). Il n'y a pas de mode « test » à quitter : l'application est utilisable dès son inscription. Dans certaines organisations, un administrateur doit autoriser l'application pour ses utilisateurs.

Autorisations (scopes) : Scopes demandés : ```. Le module reçoit : identifiant (locataire + objet), nom, e-mail (préféré : revendication email).

Erreurs fréquentes

- **AADSTS50011 : l'URL de réponse ne correspond pas** : L'URI de redirection de l'inscription (plateforme « Web ») diffère de `https://votre-boutique.example.fr/module/websourcesociallogin/callback` . Corrigez-la dans Authentification.
- **AADSTS7000215 : secret client invalide** : Vous avez copié l'« ID de secret » au lieu de la **Valeur**, ou le secret a expiré : créez-en un nouveau.
- **AADSTS700016 / AADSTS50020 : application introuvable dans le locataire** : Le type de comptes de l'inscription ou le champ « Locataire » du module ne correspondent pas (ex. compte personnel avec locataire « organizations »).
- **AADSTS65001 : consentement requis** : Un administrateur de l'organisation du client doit autoriser l'application, ou le client doit accepter la demande.
- **Un e-mail est redemandé au client** : Normal : Microsoft n'atteste pas la propriété de l'e-mail (protection contre l'attaque nOAuth). Ajouter la revendication `xms_edov` (étape 5) évite cette étape lorsque Microsoft la fournit.

Liens officiels : [Inscriptions d'applications Entra](#) · [Documentation du flux de code d'autorisation](#)

6. Apple

Sign in with Apple : le plus technique des réseaux. Il exige le programme Apple Developer (99 USD/an), un Services ID, une clé privée .p8 et une adresse en HTTPS. Le module génère lui-même le « client secret » (JWT signé ES256) à chaque connexion à partir de votre clé : vous n'avez pas à le renouveler tous les 6 mois. Apple peut masquer l'e-mail du client (adresse relais @privaterelay.appleid.com).

- **Console** : [Apple Developer](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example.fr/module/websourcesociallogin/callback`

Étape 1. Rejoindre l'Apple Developer Program

Où : *developer.apple.com/programs > Enroll*

L'adhésion est payante (99 USD/an) et peut prendre 1 à 2 jours pour une entreprise (numéro D-U-N-S).

Étape 2. Créer un App ID avec « Sign in with Apple »

Où : *Certificates, Identifiers & Profiles > Identifiers > + > App IDs > App*

Description et Bundle ID (ex. `fr.votreboutique.web`). Dans Capabilities, cochez **Sign in with Apple**, puis Continue > Register. Cet App ID sert de « Primary App ID ».

Étape 3. Créer un Services ID (= Client ID)

Où : *Identifiers > + > Services IDs*

Description et Identifier (ex. fr.votreboutique.signin) : cet **Identifiant est le Client ID**. Cochez **Sign in with Apple** puis cliquez sur **Configure**.

Champs à renseigner	
Primary App ID	l'App ID créé à l'étape 2
Domains and Subdomains	votre-boutique.example (sans https://)
Return URLs	https://votre-boutique.example/fr/module/websourcesociallogin/callback

Étape 4. Créer une clé privée

Où : *Keys > + > Key Name > Sign in with Apple > Configure*

Choisissez le Primary App ID, validez, puis **téléchargez le fichier AuthKey_XXXXXXXXXX.p8** : Apple ne le propose qu'une seule fois, gardez-le en lieu sûr. Notez le **Key ID** (10 caractères).

Étape 5. Relever le Team ID

Où : *developer.apple.com/account > Membership details > Team ID*

Identifiant de 10 caractères.

Étape 6. Remplir le module

Où : *Ce module > Apple*

Services ID, Team ID, Key ID, puis collez **tout le contenu** du fichier .p8 (de -----BEGIN PRIVATE KEY----- à -----END PRIVATE KEY-----) dans le champ « Clé privée ». Elle est chiffrée en base et jamais renvoyée au navigateur. Activez Apple et enregistrez.

Étape 7. Mise en production et e-mails relais

Où : *Certificates, Identifiers & Profiles > More > Configure Sign in with Apple for Email Communication*

Apple n'a pas de mode « test » : tout fonctionne dès la configuration. Si un client choisit « Masquer mon e-mail », vos e-mails transactionnels passent par un relais Apple : enregistrez là votre domaine d'envoi et vos adresses d'expédition (SPF et DKIM valides) sinon ils ne seront pas livrés. Apple peut aussi demander de vérifier le domaine en déposant un fichier dans /.well-known/.

Autorisations (scopes) : Scopes demandés : ` . Le module reçoit : identifiant Apple (sub), e-mail (réel ou relais), état de vérification, et le nom uniquement lors de la toute première autorisation.

Erreurs fréquentes

- **invalid_client** : Le Services ID, le Team ID, le Key ID ou la clé .p8 ne correspondent pas, ou la clé n'est pas associée au bon Primary App ID. Recréez la clé si besoin.
- **invalid_request : Invalid web redirect url / redirect_uri mismatch** : Les « Return URLs » du Services ID diffèrent de `https://votre-boutique.example/fr/module/websourcesociallogin/callback`, ou le domaine n'est pas listé dans « Domains and Subdomains ».
- **Retour sur le site « session expirée » après Apple** : Apple répond en POST depuis un autre domaine : le site doit être en HTTPS (cookie SameSite=None; Secure). Vérifiez le certificat et que le cookie de liaison n'est pas bloqué.
- **Le prénom et le nom sont vides** : Apple n'envoie le nom qu'à la première autorisation. Le client peut révoquer l'accès sur `appleid.apple.com > Connexion et sécurité > Se connecter avec Apple`, puis recommencer.
- **E-mails qui ne partent pas vers @privaterelay.appleid.com** : Enregistrez votre domaine d'envoi dans « Configure Sign in with Apple for Email Communication » (étape 7).

Liens officiels : [Services IDs Apple Developer](#) · [Documentation Sign in with Apple \(web\)](#)

7. GitHub

GitHub (OAuth App). Idéal pour une clientèle de développeurs. Le module n'utilise que les e-mails marqués « vérifiés » par GitHub. Comptez 5 minutes. Gratuit.

- **Console** : [GitHub Developer settings](#)
- **URL de rappel (callback) à enregistrer** : `https://votre-boutique.example/fr/module/websourcesociallogin/callback`

Étape 1. Ouvrir les réglages développeur

Où : `github.com` > votre avatar > Settings > Developer settings > OAuth Apps > New OAuth App

Pour une application appartenant à une organisation : Organization > Settings > Developer settings > OAuth Apps.

Étape 2. Remplir le formulaire

Où : Register a new OAuth application

GitHub n'accepte **qu'une seule** URL de rappel par OAuth App : créez une application par domaine de boutique.

Champs à renseigner	
Application name	le nom de votre boutique
Homepage URL	<code>https://votre-boutique.example</code>
Authorization callback URL	<code>https://votre-boutique.example/fr/module/websourcesociallogin/callback</code>
Enable Device Flow	laisser décoché

Étape 3. Récupérer le Client ID et générer le secret

Où : Après « Register application » > Client ID, puis « Generate a new client secret »

Copiez le **Client ID** puis cliquez sur **Generate a new client secret** : il n'est affiché qu'une seule fois, copiez-le immédiatement dans ce module.

Étape 4. Scopes

Où : Ce module > GitHub > Scopes

Par défaut `read:user user:email` : `user:email` permet de lire les e-mails du compte, dont le principal vérifié.

Étape 5. Mise en production

Où : Application > Advanced / logo

Les OAuth Apps n'ont ni revue ni mode test : elles sont utilisables immédiatement par tous les comptes GitHub. Ajoutez un logo (Upload new logo). Si l'organisation d'un client restreint l'accès aux applications tierces, il verra « Request » : c'est à l'administrateur de son organisation d'approuver.

Autorisations (scopes) : Scopes demandés : ```. Le module reçoit : identifiant GitHub (id), nom, e-mail principal vérifié.

Erreurs fréquentes

- **redirect_uri_mismatch** : L'URL de rappel de l'OAuth App diffère de `https://votre-boutique.example/fr/module/websourcesociallogin/callback` (une seule URL possible par application, et son chemin doit commencer par celui enregistré).
- **incorrect_client_credentials** : Client ID ou secret erroné : régénérez un secret.
- **bad_verification_code** : Le code a déjà été utilisé ou a expiré (10 minutes) : recommencez la connexion.
- **Aucun e-mail reçu** : Le compte n'a aucune adresse vérifiée (ou le scope `user:email` a été retiré) : le client saisit son e-mail

et le confirme par un lien.

Liens officiels : [OAuth Apps GitHub](#) · [Documentation GitHub OAuth](#)

Problèmes courants (tous réseaux)

- **Aucun bouton n'apparaît** : Un réseau n'est visible que s'il est **activé ET complet** (tous les champs obligatoires renseignés) et que l'emplacement est coché dans « Où afficher les boutons ». Les réseaux incomplets restent invisibles pour vos clients.
- **« Session expirée » après le retour du réseau** : Le cookie de liaison a été bloqué (navigateur en navigation privée stricte, domaine différent entre www et sans www) ou la demande a duré plus de 10 minutes. Réessayez ; utilisez toujours la même adresse (l'URL de rappel ci-dessus est celle de la boutique).
- **Le réseau refuse une URL « http:// »** : Passez la boutique en HTTPS (Paramètres de la boutique > Trafic & SEO > Activer SSL sur toute la boutique). L'URL de rappel est affichée en https.
- **Boutique multilingue ou plusieurs domaines** : L'URL de rappel est la même pour toutes les langues d'un domaine (elle utilise la langue par défaut). Avec plusieurs domaines/boutiques, enregistrez l'URL de chaque boutique chez le réseau (affichée dans chaque boutique).

6. Personnalisation des boutons

Onglet **Apparence et emplacements** : l'aperçu se met à jour en direct.

Réglage	Options
Style	<i>Contour</i> (fond blanc, bordure du thème), <i>Plein</i> (couleur officielle du réseau), <i>Neutre</i> (aux couleurs du thème)
Disposition	colonne pleine largeur, ou grille d'icônes
Forme	carrée, arrondie (thème), pilule
Icône seule	masque le texte (conservé pour les lecteurs d'écran)
Rayon, bordure	en pixels (0 à 32 et 0 à 4)
Couleurs	fond, texte, bordure : <code>#rgb</code> , <code>#rrggbb</code> , <code>rgb()</code> , <code>rgba()</code> , <code>transparent</code> ou <code>var(--variable-du-thème)</code> ; toute autre valeur est refusée
Séparateur « ou »	affiché ou non
Dernier réseau utilisé	mis en premier (cookie local sans donnée personnelle)

Les logos Google et Microsoft gardent leurs couleurs officielles, comme l'exigent leurs chartes.

Contour	Plein	Neutre
OU Continuer avec Google Continuer avec Facebook Continuer avec X Continuer avec LinkedIn Continuer avec Microsoft Continuer avec Apple Continuer avec GitHub En continuant, vous acceptez nos conditions générales et notre politique de confidentialité. Nous recevons de votre compte son identifiant, votre nom et votre adresse e-mail pour créer ou retrouver votre compte client. Rien n'est publié en votre nom.	OU Continuer avec Google Continuer avec Facebook Continuer avec X Continuer avec LinkedIn Continuer avec Microsoft Continuer avec Apple Continuer avec GitHub En continuant, vous acceptez nos conditions générales et notre politique de confidentialité. Nous recevons de votre compte son identifiant, votre nom et votre adresse e-mail pour créer ou retrouver votre compte client. Rien n'est publié en votre nom.	OU Continuer avec Google Continuer avec Facebook Continuer avec X Continuer avec LinkedIn Continuer avec Microsoft Continuer avec Apple Continuer avec GitHub En continuant, vous acceptez nos conditions générales et notre politique de confidentialité. Nous recevons de votre compte son identifiant, votre nom et votre adresse e-mail pour créer ou retrouver votre compte client. Rien n'est publié en votre nom.

Emplacements

Cochez où afficher les boutons : **page de connexion**, **page de création de compte**, **tunnel de commande** (informations personnelles), **en-tête** (menu « Mon compte », visiteurs), **fenêtre de la liste de souhaits**. Les boutons ne s'affichent jamais à un client déjà connecté.

Réseaux
Apparence et emplacements
Textes et RGPD
Aide

Apparence des réseaux

Style
Contour (fond blanc, bordure du thème)

Disposition
Colonne, pleine largeur

Forme
Arrondie (thème)

☐ Icône seule (texte masqué, conservé pour les lecteurs d'écran)

Rayon (px, 0-32)
Bordure (px, 0-4)

auto
1

Couleurs facultatives (remplacent le style) : #rgb, #rrggbb, #rrggbbaa, rgb(), rgba(), transparent ou var(--couleur-du-theme). Toute autre valeur est refusée.

Fond
Texte
Bordure

#ffffff
#1a1a1a
rgba(0,0,0,.2)

☒ Afficher le séparateur « ou »

☒ Mettre en avant le dernier réseau utilisé (cookie local sans donnée personnelle)

Aperçu

OU

Continuer avec Google
Continuer avec Facebook
Continuer avec X
Continuer avec LinkedIn
Continuer avec Microsoft
Continuer avec Apple
Continuer avec GitHub

En continuant, vous acceptez nos conditions générales et notre politique de confidentialité. Nous recevons de votre compte son identifiant, votre nom et votre adresse e-mail pour créer ou retrouver votre compte client. Rien n'est publié en votre nom.

Aperçu avec tous les réseaux ; côté boutique seuls les réseaux actifs et complets s'affichent.

Où afficher les boutons

La connexion automatique s'affiche partout où une connexion ou une inscription est nécessaire.

☒ Page de connexion
☒ Page de création de compte
☒ Tunnel de commande (informations personnelles)
☒ En-tête : menu « Mon compte » (visiteur)
☒ Fenêtre « connectez-vous » de la liste de souhaits

Textes et RGPD

Onglet **Textes et RGPD** : texte des boutons (par langue), séparateur et mention d'information affichée sous les boutons (à adapter à votre politique de confidentialité).

Textes

Texte des boutons

Utilisez {network} pour insérer le nom du réseau. Vide = « Continuer avec {network} ».

FR
Continuer avec {network}

EN
Continue with {network}

Mention RGPD

Affichée sous les boutons. Vide = texte par défaut.

FR

En continuant, vous acceptez nos conditions générales et notre politique de confidentialité. Nous recevons de votre compte son identifiant, votre nom et votre adresse e-mail pour créer ou retrouver votre compte client. Rien

EN

By continuing, you accept our terms and conditions and privacy policy. We receive your account ID, name and e-mail address from the network to create or find your customer account. Nothing is posted on your behalf.

7. Compatibilité thème

Le module s'appuie sur deux hooks :

- `displayCustomerLoginFormAfter` : hook natif du formulaire de connexion (thèmes Classic, Hummingbird et la plupart des thèmes tiers). Aucune action requise : les boutons apparaissent sur la page de connexion.
- `displayWebsourceSocialLogin` : hook du module, à ajouter dans votre thème pour les autres emplacements, avec le paramètre `place` : `register` , `checkout` , `header` , `wishlist` .

Si un emplacement n'apparaît pas, ajoutez l'appel dans le gabarit du thème (copie dans votre thème enfant si besoin), puis videz le cache Smarty :

```
{* Page de création de compte : themes/<theme>/templates/customer/registration.tpl, après le formulaire *}
{hook h='displayWebsourceSocialLogin' place='register'}

{* Tunnel de commande : templates/checkout/_partials/steps/personal-information.tpl *}
{hook h='displayWebsourceSocialLogin' place='checkout'}

{* Menu « Mon compte » de l'en-tête (visiteur), dans le gabarit de votre bloc compte *}
{hook h='displayWebsourceSocialLogin' place='header'}

{* Fenêtre de la liste de souhaits (modals/login.tpl du module blockwishlist, surcharge du thème) *}
{hook h='displayWebsourceSocialLogin' place='wishlist'}

{* Page de connexion si votre thème n'appelle pas le hook natif : templates/customer/authentication.tpl *}
{hook h='displayCustomerLoginFormAfter'}
```

La page « Mon compte » reçoit une tuile **Connexions sociales** via le hook natif `displayCustomerAccount` .

Style : `views/css/front.css` utilise les variables CSS du thème (`--color-brand-*` , `--color-line` , `--radius-xl` ...) avec des valeurs de repli. Pour un thème tiers, ajustez le style, les couleurs et le rayon dans l'onglet Apparence, ou surchargez les classes `.wssl` , `.wssl-btn` , `.wssl__sep` dans le CSS de votre thème.

Après un changement de gabarit : Paramètres avancés > Performances > Vider le cache.

8. RGPD et sécurité

Données stockées

Table `ps_websourcesociallogin_link` (une ligne par réseau lié) : client, réseau, identifiant du client chez le réseau, e-mail transmis, date de consentement, date de création, date de dernière connexion. Table `ps_websourcesociallogin_flow` : états temporaires de connexion (hachés, supprimés à expiration, 10 minutes à 1 heure). **Aucun jeton d'accès n'est stocké ni journalisé.** Le journal PrestaShop ne contient que le réseau, l'événement et un code court (sans e-mail, jeton ni secret) ; l'adresse IP n'est conservée que sous forme de hachage pour la limitation du taux.

Informez vos clients dans votre politique de confidentialité : le module reçoit du réseau l'identifiant, le prénom, le nom et l'e-mail (rien n'est publié sur le réseau). Une mention configurable s'affiche sous les boutons.

Export et suppression

Les liens sont inclus dans l'export RGPD du client (`actionExportGDPRData`) et supprimés avec le compte (`actionDeleteGDPRCustomer` , suppression du client). Le client peut délier un réseau lui-même dans Mon compte > Connexions sociales.

Protections

- **Anti-prise de compte** : jamais de fusion automatique. E-mail vérifié correspondant à un client existant : mot de passe exigé

(5 essais maximum). E-mail non vérifié ou absent : lien de confirmation, dans le même navigateur. Microsoft : e-mail considéré non vérifié sauf revendication `xms_edov` (protection `nOAuth`).

- `state` aléatoire (256 bits), à usage unique, 10 minutes, lié au navigateur par un cookie `HttpOnly` ; `nonce` OpenID Connect ; PKCE S256 là où le réseau le gère.
- `id_token` : RS256 uniquement (`none` et HS256 refusés), signature vérifiée par JWKS, contrôle de `iss` , `aud` , `exp` , `iat` , `nbf` .
- Redirections sur liste blanche (aucune redirection ouverte).
- Secrets chiffrés au repos (chiffrement PrestaShop), jamais dans le HTML ni le JavaScript.
- Limitation du taux : 30 démarrages par 10 minutes et par IP.
- Liaison et déliaison en POST avec jeton CSRF.

9. Dépannage et FAQ

Aucun bouton n'apparaît. Le réseau doit être activé et complet, et l'emplacement coché. Vérifiez le hook du thème (section 7), videz le cache. Un client connecté ne voit jamais les boutons.

« **redirect_uri_mismatch** » **ou équivalent.** L'URL enregistrée chez le réseau diffère de celle du module : copiez-la avec le bouton Copier (https, même nom de domaine avec ou sans www, pas de « / » final). Une modification peut mettre quelques minutes à être prise en compte.

Seuls mes comptes peuvent se connecter. L'application est en mode test/développement chez le réseau : publiez-la (voir la fin de chaque guide).

« **Session expirée** » **au retour.** Cookie bloqué, domaine différent (www ou non), ou délai de plus de 10 minutes : recommencez en utilisant toujours la même adresse.

Le client n'a pas d'e-mail. Il le saisit puis le confirme par lien (X, Microsoft, Apple avec relais...).

Où voir les erreurs ? Paramètres avancés > Informations > Logs : filtrez sur `[websourcesociallogin]` .

Boutique multilingue ou multiboutique ? L'URL de rappel est la même pour toutes les langues d'un domaine. Avec plusieurs domaines ou boutiques, enregistrez l'URL de chaque boutique chez le réseau (affichée dans chaque contexte).

Mon thème est tiers et un emplacement manque. Voir la section 7.

Y a-t-il un coût ? Le module est gratuit ; les réseaux listés n'ont pas de coût pour la connexion.

10. Désinstallation

Modules > Webservice Social Login > Désinstaller : supprime les deux tables du module (liens et états temporaires) et tous ses réglages. **Les liens réseau-client sont alors perdus** : les clients créés via un réseau conservent leur compte, mais n'ont pas de mot de passe connu et devront utiliser « Mot de passe oublié ». Pensez à exporter ce dont vous avez besoin avant.

11. FAQ développeur : ajouter un réseau

Les réseaux sont des classes `Webservice\SocialLogin\Provider\ProviderInterface` (dossier `src/Provider/`). Pour un fournisseur OpenID Connect, étendez `OidcProvider` ; sinon `AbstractProvider` .

1. Créez `src/Provider/MonReseauProvider.php` implémentant : `id()` , `label()` , `fields()` (champs de configuration : `client_id` , `client_secret` ...), `defaultScopes()` , `usesPkce()` , `usesFormPost()` , `buildAuthorizationUrl(...)` et `fetchProfile(...)` qui renvoie un objet `Profile` (identifiant, e-mail, e-mail vérifié, prénom, nom).
2. Enregistrez-le dans `ProviderRegistry::default()` .
3. Ajoutez son guide dans `src/Guide/data_fr.php` et `data_en.php` (clés `intro` , `console` , `steps` , `scopes` , `errors` , `links`) et son logo dans `src/Render/IconLibrary.php` .
4. Lancez les tests : `php tests/run.php` (sans PrestaShop ni réseau ; fournisseurs testés avec un client HTTP factice).

L'anti-prise de compte et le flux de liaison sont gérés par le noyau (`Linking\LinkingService` , `Flow\FlowService`) : un fournisseur doit seulement renvoyer un `Profile` fiable, en indiquant honnêtement si l'e-mail est vérifié.

12. Changelog

Voir [CHANGELOG.md](#). Version 1.0.0 : première version publique.

13. Licence et support

Académique Free License 3.0 (AFL-3.0), voir `LICENSE.md` . Bibliothèques tierces : voir `THIRD_PARTY.md` . Signalez un problème sur [GitHub](#) ou via [webservice.fr](#).