

Module « Études » – Spécifications fonctionnelles & techniques (v 1.1 – brouillon)

Mise à jour après revue du 19 mai 2025 : intégration de la maquette UX, amélioration de la lisibilité des tableaux, retrait des replays vidéo et des tags WordPress.

1. Parcours fonctionnels

Acteur	Parcours résumé	Traces & notifications
Visiteur (anonyme)	<ol style="list-style-type: none">1. Accède à l'archive des études (seules celles > 1 an sont listées).2. Lance une recherche (plein-texte + filtres).3. Télécharge le rapport ou la synthèse (FR/EN).	–
Membre authentifié	<ol style="list-style-type: none">1-2. Parcours identique au visiteur mais voit <i>toutes</i> les études.3. Télécharge rapport / synthèse (FR/EN).	–
Manager (rôle unique)	<ol style="list-style-type: none">1. Crée une étude en brouillon (CPT).2. Renseigne les métadonnées (N° d'étude, axes, auteurs, année...).3. Clique « Publier (membres) » → l'étude est visible côté membres.4. Cron quotidien : à J+365, bascule automatique en public (ou via bouton « Rendre publique »).5. Suit les téléchargements directement dans la fiche BO (compteur visiteurs + liste des membres). EN ATTENTE	–

2. Architecture logique

Bloc	Description	Notes
CPT : <i>etude</i>	1 post = 1 étude terminée.	WPML - Multiple champs FR et EN dans la page de la langue par défaut.

Taxonomies	<ul style="list-style-type: none"> • <code>axe_recherche</code> (hiérarchique) • <code>auteur_etude</code> (non hiérarchique) 	Utilisées dans les filtres.
Champs ACF	<ul style="list-style-type: none"> • N° d'étude (texte) • Date de finalisation (date) • Mots-clés libres (champ texte) • Rapport PDF FR / EN (fichier) • Synthèse PDF FR / EN (fichier) • Statut interne (lecture seule – « membres » / « public ») 	Fichiers stockés hors /uploads (protégés par <code>.htaccess deny</code>).
Table SQL : <code>wp_studies_downloads</code>	<code>id, study_id, user_id (0 = visiteur), timestamp, file_type</code>	Alimente les statistiques.
Options ACF (page « Études »)	<ul style="list-style-type: none"> • Poids max des fichiers • Types MIME autorisés 	Alignées avec celles du module AO.

3. Comportement détaillé

Fonction	Mécanisme
Publication initiale	Le bouton « Publier (membres) » ajoute automatiquement <code>etude_visibilite = membres</code> .
Bascule « public »	Cron quotidien : si <code>date_finalisation ≤ today - 365 jours</code> ⇒ change <code>etude_visibilite</code> en <code>public</code> .
Moteur de recherche	<ul style="list-style-type: none"> • Formulaire en GET avec : <code>s</code> (plein-texte), <code>axe</code>, <code>auteur</code>, <code>annee</code>, <code>numero</code>. • <code>WP_Query</code> avec <code>tax_query</code> + <code>meta_query</code>. • Option AJAX pour rechargement asynchrone.
Téléchargement sécurisé	<ol style="list-style-type: none"> Endpoint <code>/download-study/?id=123&type=rapport-fr</code> :1. Vérifie la capacité (public vs membre). Log dans <code>wp_studies_downloads</code>. Force-download (headers <code>Content-Disposition</code>).
Statistiques BO	Dans l'éditeur de chaque étude : <ul style="list-style-type: none"> • Compteur visiteurs (total global). • Liste des membres ayant téléchargé (login, date / heure, type de fichier). • Bouton « Exporter CSV ». EN ATTENTE

RGPD / purge

Cron annuel : anonymise ou supprime les logs > 3 ans (durée à confirmer).

4. Gabarits front-end (sobriété et accessibilité)

Fichier	Rôle	Particularités
<code>archive-etude.php</code>	Liste paginée avec formulaire de recherche + filtres visibles.	Charge CSS/JS uniquement ici.
<code>single-etude.php</code>	Détail de l'étude : méta, lien synthèse / rapport.	Bloc « Accès réservé » si visibilité = membres.
<code>taxonomy-axe_recherche.php</code>	Vue filtrée par axe (optionnelle).	Génère moins de requêtes.

5. Rôles & capacités

Rôle	Capabilités clés
Manager	<code>manage_etudes</code> , <code>publish_etude</code> , <code>edit_download_stats</code> , <code>manage_options</code>
Subscriber (WP)	Aucune capacité supplémentaire ; accès front-end réservé.

6. Sécurité & éco-conception

- Stockage des PDF **hors webroot** + règle `.htaccess`.
 - Nonces + contrôle `current_user_can()` sur chaque endpoint.
 - reCAPTCHA v3 prêt pour tout futur formulaire (ex. « Proposer une étude »).
 - **Eco-Index B** minimum : CSS critiques inlinés, JS chargé différé, poids page < 200 Ko.
-

7. Étapes de développement (ordre conseillé)

1. **Structure thème** : /études/ + autoloader.
 2. **Déclaration CPT, taxonomies, capabilities.**
 3. **ACF Field Groups** (export JSON pour versionning).
 4. **Options page** « Études ».
 5. **Cron “bascule public”** + wrapper mail (actuellement désactivé mais prêt).
 6. **Endpoint téléchargement sécurisé** + table SQL + fonctions de log.
 7. **Templates front** + formulaire de recherche (Ajax facultatif).
 8. ~~**Bloc statistiques BO** (WP_List_Table + export CSV).~~
 9. **Tests accessibilité / performance** (Lighthouse ≥ 90 / EcoIndex $\geq B$).
 10. **Documentation** : README + quick-start.
-

8. Résultats attendus

- **Recherche** : réponse < 300 ms sur 1 000 études.
 - **Pages** : < 200 Ko, aucune librairie superflue.
 - **Administration** : cycle « brouillon → membres → public » sans double saisie.
 - **Traçabilité** : compteur visiteurs fiable, liste membres accessible + [exportable.EN](#)
- ATTENTE**
-

Prochaines actions :

1. Valider la liste définitive des champs ACF (nommage + aides de saisie).

2. Confirmer la durée de conservation RGPD pour les logs de téléchargement.
3. Choisir la stack de recherche front (simple [WP_Query](#) ou ElasticPress si index externe).

Merci de relire ! Tes retours seront intégrés à la v 1.2.