

Formule — Modification groupée

Documentation technique du contrôleur `formule_modification_groupee_controller` — Smart Central — 01/10/2026

Contrôleur	<code>app/Http/Controllers/formule_modification_groupee_controller.php</code>
Vue	<code>resources/views/formule_modification_groupee/index.blade.php</code>
Routes	<code>routes/web.php</code> (groupe <code>formule_modification_groupee</code>)
Migration droit d'ajout	<code>2026_09_29_1200_V905000_NB_t_profil_formule_modification_groupee_ajout.php</code>

1. But

Écran qui permet de modifier **en une seule opération la composition (`t_composition`) de plusieurs formules** d'une même centrale :

- remplacer un produit par un autre (avec ou sans changement de quantité) ;
- ajouter un produit + un dosage à des formules ;
- supprimer un produit de formules.

Jusqu'à **5 paires de produits** peuvent être traitées en même temps, sur toutes les formules cochées dans le tableau.

2. Routes

Méthode	URL	Contrôleur	Rôle
GET	<code>/formule_modification_groupee</code>	<code>index()</code>	Affiche la page (middleware <code>auth:sanctum</code> , interne)
GET	<code>/liste_produit</code>	<code>liste_produit()</code>	Options des produits « à remplacer »
GET	<code>/liste_produit_substitution</code>	<code>liste_produit_substitution()</code>	Options des produits « de remplacement / à ajouter »
GET	<code>/table</code>	<code>data_table()</code>	Données JSON du DataTable des formules
POST	<code>/valider</code>	<code>valider()</code>	Applique les modifications

3. Droits (`t_profil`)

Champ	Effet
<code>profil_utilil_formule</code>	Si <code>0</code> → <code>index()</code> renvoie 401 (pas d'accès à la page).
<code>profil_formule_modification_groupee_ajout</code>	Booléen (défaut <code>0</code>). Si <code>0</code> : l'option « Ajouter un produit et un dosage » n'apparaît pas, <code>data_table()</code> renvoie 401 pour <code>action=3</code> , et <code>valider()</code> ignore silencieusement les paires d'ajout. Case à cocher dans la fiche profil « Modification groupée formule : ajout produit ».

4. Les 4 actions (action)

Val.	Libellé	Champs visibles	Effet sur la quantité
0	Ne remplacer que les codes produits	produit à remplacer / de remplacement	Quantité d'origine conservée
1	Remplacer sans vérifier les quantités d'origine	+ qté de remplacement	Nouvelle qté si saisie (> 0), sinon conservée
2	Remplacer en vérifiant les quantités d'origine	+ qté à remplacer + qté de remplacement	Ne touche que les compositions dont la qté = qté à remplacer ; nouvelle qté si saisie, sinon conservée
3	Ajouter un produit et un dosage (droit dédié)	produit à ajouter + dosage	Crée la composition avec le dosage (0 si vide)

Types de produit (type_produit / composition_type) : 0 Gra, 1 Liant, 2 EAU, 3 ADJ, 5 Ajout (table t_ajout), 9 Aucun.

5. Détail des méthodes

5.1 index()

- Vérifie profil_outil_formule.
- Charge les listes : centrales, variantes, désignation béton, classes consistance / résistance / chlorure / exposition, type de béton.
- Passe peut_ajouter à la vue (affiche ou non l'action 3).

5.2 liste_produit() — produits à remplacer

- type_produit numérique → tous les produit de ce type (**toutes centrales**, pas de filtre).
- type_ajout numérique → tous les ajout.
- Retourne { produit: "<option>..." } au format code -> libellé.

5.3 liste_produit_substitution() — produits de remplacement

Même chose, mais **filtré sur la centrale** via t_produit_centrale / t_ajout_centrale : seuls les produits rattachés à la centrale choisie sont proposés.

5.4 data_table() — liste des formules

Entrées : centrale_id, produit_type, action, formule_variant_id, produits_a_remplacer[], qtes_a_remplacer[] + 6 filtres de classification.

1. **Contrôles** : action 3 sans droit → 401 ; action 3 avec type invalide → tableau vide ; aucun produit à remplacer (hors action 3) → vide ; action 2 avec un produit sans quantité → vide.
2. **Requête de base** sur composition : centrale, formule_id <> 0, et composition_type = produit_type **sauf en action 3** (une formule peut ne pas encore posséder ce type).
3. **Pour chaque produit renseigné** : la formule (clé formule_id_variante) doit contenir ce produit (et cette quantité en action 2) → filtres cumulés en **ET**.
4. **Filtres optionnels** : variante, puis désignation / consistance / résistance / type béton / chlorure / exposition.
5. **Regroupement** par formule + variante, puis remplissage des 5 colonnes produit : d'abord les produits recherchés dans leur slot, puis les autres compositions du même type dans les slots libres.
6. **Ligne renvoyée** : case à cocher, code formule, app. commerciale, variante, 5 × (produit, qté), 6 libellés de classification. recordsTotal alimente le badge « Nombre de formule(s) ».

Le tableau se recharge à chaque changement d'action, de produit, de quantité, de variante, de filtre ou de centrale.

5.5 valider() — application

Entrées : `action`, `formule[]` (ids cochés), `centrale_id`, `type_produit`, `formule_variant_id`, `pairs[]` = `{produit_id, qte_produit, produit_substitution, qte_substitution}`.

Pour **chaque formule × chaque paire**, le mode est déduit du contenu de la paire :

Paire	Mode	Traitement
produit et remplacement	Modification	Recherche des compositions (formule, centrale, type, produit, variante ; + qté en action 2). Met à jour produit, code, libellé et quantité (voir §4). Traça « Modification ».
remplacement seul	Ajout (si droit)	Variante cible = variante choisie, sinon 1. Si le produit existe déjà : mise à jour de la quantité si différente (traça « Modification »), sinon création d'une composition (traça « Ajout »).
produit seul	Suppression	Supprime les compositions correspondantes (+ filtre qté en action 2). Traça « Suppression » avec qté 0.

- Code/libellé du nouveau produit lus dans `ajout` (type 5) ou `produit` ; si introuvable, la paire est ignorée.
- Chaque écriture passe par `ecrire_traca_composition()` (avant modification/suppression, après création).
- Réponse : « Enregistrement validé (N ligne(s) modifiée(s)) ».

6. Déroulé côté écran

1. Choisir l'action, éventuellement les filtres de classification.
2. Choisir centrale, variante (STD par défaut), type de produit → chargement des listes produits.
3. Renseigner jusqu'à 5 paires (produit à remplacer → produit de remplacement, + quantités selon l'action).
4. Le tableau affiche les formules qui contiennent **tous** les produits à remplacer.
5. Cocher les formules (ou « tout cocher »), cliquer **Valider**, confirmer.
6. Message de résultat puis rechargement du tableau après 3 s.

7. Points d'attention

- **Suppression implicite** : en actions 0/1/2, une paire avec un produit à remplacer mais **sans produit de remplacement** supprime ce produit des formules cochées. L'écran ne le signale pas.
- **Ajout implicite** : en actions 0/1/2, une paire avec uniquement un produit de remplacement déclenche un ajout (si droit d'ajout).
- **Pas de transaction** : en cas d'erreur en cours de traitement, les modifications déjà faites restent.
- Seule la route `index` porte le middleware `auth:sanctum / interne` et seul `index()` contrôle `profil_outil_formule` (même convention que les autres modules) ; `valider()` ne revérifie que le droit d'ajout.
- Le produit de remplacement n'est pas revérifié côté serveur comme rattaché à la centrale (seule la liste de l'écran est filtrée).
- La vue teste encore `type_produit == 4` alors que cette option n'existe pas dans la liste.
- Un ancien écran PHP procédural existe encore dans `public/formule_modification_groupee_laravel_default/` (non utilisé par la version Laravel).