

Résultats, évaluations et correction de module

Génération des résultats

`byModuleGenerateTeamModuleResults()` orchestre grading + reassemble puis persiste le résultat dans plusieurs documents.

```
sequenceDiagram
    participant G as GenerateResults
    participant TM as teamsModules
    participant T as teams
    participant E as evaluationTeam
    participant RTT as RTT
    G->>TM: lire moduleTypesData
    G->>G: gradeModule()
    G->>G: reassembleAnswer()
    G->>T: teamDecisionsData.results[actModuleUid][moduleType]
    G->>E: actPerf[actModuleUid][moduleType]
    G->>RTT: execute_request GetTeamModuleTypeResults
    RTT-->>Unity: récupérer le résultat
```

Document teams

Le résultat complet est enregistré dans `teamDecisionsData.results[actModuleUid][moduleType]`. Il contient notamment `scores`, `errors`, `expectedValues` et `errorScores`.

evaluationTeam

Les scores sont aussi propagés dans `evaluationTeam.actPerf`, afin d'alimenter l'évaluation globale de l'activité.

Summative retry

Lorsque la préférence `RETRY_SUMMATIVE` est active, le code compare la nouvelle note globale à `lastSummativeGrade`. La meilleure note est conservée; sinon les anciens scores d'évaluation sont restaurés. Un message `module_graded` est envoyé pour synchroniser l'état.

Pourquoi les résultats ne sont pas envoyés directement en RTT

Le commentaire serveur indique que le message serait trop gros. Le serveur envoie donc `execute_request` avec `GetTeamModuleTypeResults`; le client fait ensuite une commande dédiée pour récupérer la payload.

Correction côté Unity

Les résultats alimentent notamment `CorrectedTeamModuleWindowControl`, `ModuleErrorsPopupControl`, `ModuleResultsReportItemControl` et l'évaluation By Module. `ignoreErrorFromTeamModuleResults` permet de modifier l'état de certaines erreurs après correction.

Debug

Si l'answer key est correcte mais la note est mauvaise, inspecter dans l'ordre : résultat brut de `gradeModule` → `reassembleAnswer` → persistance dans `teams` → `evaluationTeam` → récupération côté Unity.

Ajouter une nouvelle donnée de résultat ou de correction

1. Déterminer si la donnée appartient au **raw grading**, au **reassemble** ou au **résultat final UI**.
2. Ajouter la valeur au bon objet retourné par grading/reassemble.
3. Persister uniquement ce qui doit survivre au reload dans `teamDecisionsData.results[actModuleUid][moduleType]`.
4. Si la valeur contribue à la note globale, mettre à jour `evaluationTeam.actPerf`.
5. Ajouter la désérialisation Unity dans `ResultsByModule` ou le modèle concerné.
6. Afficher la donnée dans la correction/report sans recalculer une deuxième logique différente côté Unity.
7. Si la payload devient grosse, conserver le pattern `execute_request → GetTeamModuleTypeResults`.

Retry summative

Toute nouvelle valeur de score doit être compatible avec le mécanisme `RETRY_SUMMATIVE`. Si la nouvelle tentative est moins bonne, le code peut restaurer l'ancienne note. Vérifier que les données auxiliaires de correction restent cohérentes avec la tentative retenue.

Debug du pipeline résultat

1. inspecter la sortie `byModuleGradeModule()`;
2. inspecter `byModuleReassembleAnswer()`;
3. inspecter l'objet écrit dans `teams`;
4. inspecter `evaluationTeam`;
5. appeler manuellement `GetTeamModuleTypeResults`;
6. inspecter la désérialisation Unity.

Revision #2

Created 2026-09-18 19:16:33 UTC by Mathis

Updated 2026-09-18 20:13:46 UTC by Mathis