Écart : Méthodologie Intersel vs livrables projet (bmad)

Application : Suivi FTI EMGI
Date : 2026-02-09
Référence : Méthodologie de développement Intersel (wiki – cycle en V, PAQ, phases, gestion de projet).
Objectif : Identifier ce qui **manque** dans la méthodologie par rapport aux besoins du projet (chiffrage, client, technique, POC/Agile) et ce qui manque au projet par rapport à la méthodologie.

---

1. Ce qui manque DANS la méthodologie (par rapport au projet bmad et aux bonnes pratiques)

1.1 Livrables de conception / chiffrage non cités explicitement

Manque	Où c'est utile	Référence projet bmad
Schéma relationnel / modèle de données	Chiffrage back, migrations, performance.	*schema-relationnel-et-tables-donnees.md*
Description détaillée des tables (Nom, Type, Description, Clé)	Spécification données, recette, onboarding dev.	Idem, format « tables »
Schéma des workflows métier (états, transitions, acteur)	Chiffrage des validations, statuts, règles métier.	*schema-workflows.md*
Schéma des intégrations et flux (endpoints, points complexes)	Chiffrage API, intégrations externes (ex. Pennylane).	*schema-chiffrage-projet.md* § 5
Schéma d'architecture applicative (composants)	Vue système, répartition des tâches.	*schema-chiffrage-projet.md* § 6
Schéma d'infrastructure et déploiement	Chiffrage déploiement, hébergement.	*schema-chiffrage-projet.md* § 7
WBS / découpage en lots chiffrables (backlog structuré par modules)	Estimation, planning, lots livrables.	*schema-chiffrage-projet.md* § 1
Matrice rôles et permissions (RBAC)	Éviter l'oubli des droits dans le chiffrage.	*schema-chiffrage-projet.md* § 2

La méthodologie mentionne « Document de conception préliminaire / détaillée » et « Document de conception système » (DCG) mais ne liste pas ces **schémas** comme livrables nommés. Ils peuvent être inclus dans le DCP/DCG ; pour un chiffrage et une recette clairs, les nommer explicitement (dans le PAQ ou la liste des types de documents) est utile.

1.2 Chiffrage et décisions client

Manque	Comment le combler
Document de chiffrage (ou fiche de chiffrage)	Non cité comme livrable. À prévoir en amont (grille de complexité, j/h par lot, contingence).
Liste des questions en suspens / décisions client	Pour figer le périmètre avant chiffrage (auth, rôles, intégrations, hébergement). Référence : *questions-en-suspens.md*.
Synthèse projet pour lecture client	Permettre au client de relire le périmètre (résumé, glossaire, écrans, hors scope). Référence : *synthèse-projet-lecture-client.md*. Non prévu dans la liste des DOC / livrables.

1.3 Onboarding et équipe

Manque	Comment le combler
Guide stagiaire / onboarding	Pas dans la méthodologie. Utile pour nouveaux devs / stagiaires (où sont les docs, comment lancer, où modifier quoi). Référence : *app/ONBOARDING-STAGIAIRE.md*.
Référence explicite au glossaire métier	Glossaire cité dans le menu du wiki mais pas dans les livrables par phase. Pour un projet métier (alevins, larves, encours, Pennylane), l'inclure dans les « éléments d'entrée » ou la spécification.

1.4 POC / mode Agile

Manque	Comment le combler
Livrables spécifiques POC	La méthodologie reste centrée cycle en V. Pour un POC : préciser livrables allégés (ex. schémas de chiffrage, cahier recettes, pas forcément MUL complet ni tous les dossiers de tests).
Definition of Done POC	Définir ce qui suffit pour « POC livrable » (recette sur cahier, pas de tests auto obligatoires, etc.).
Adaptation des phases pour sprints	« Sprints » et « Agile » sont mentionnés mais les tableaux de phases et livrables restent en V ; pas de liste type « livrables par sprint ».

1.5 Outils et références datées

Manque	Remarque
Réunions prévues (4.3.3)	Les dates sont en **2015** (template non mis à jour). À remplacer par un modèle « à compléter par projet » ou des dates réelles.
Twitter pour suivi projet (4.4)	Outil possiblement obsolète ou peu utilisé. Vérifier si à conserver ou remplacer (Slack, wiki, autre).
SVN vs Git	La méthodologie mentionne SVN (intégration) et 4.11 cite GOGS (Git). Clarifier l'outil de référence (Git/GOGS) et les procédures (branches, revue).
Gestion de configuration – liste des types de documents (4.6.1.1)	Liste riche (BL, CPRL, CRR, DSL, DCG, DCP, DML, etc.) mais sans « Schéma de données », « Schéma des workflows », « Document de chiffrage », « Synthèse client ». Possibilité d'étendre la liste ou de les ranger dans un type existant (ex. DCG).

1.6 Sécurité et qualité

Présent dans la méthodologie	Manque éventuel
Sécurité des accès, données, attaques (4.5.2)	**RGPD / données personnelles** : pas de mention explicite (durée de conservation, droit d'accès, etc.) si l'application traite des données personnelles.
Tests de robustesse MD10 (4.7)	Référence MD10 non détaillée dans l'extrait ; à s'assurer que le projet sait où trouver la procédure.
FCEA / Mantis (4.6.3)	Bien décrit. —

---

2. Ce qui manque AU PROJET bmad par rapport à la méthodologie

Pour que le projet bmad soit **aligné** avec la méthodologie Intersel, il faudrait au moins :

Exigence méthodologie	État projet bmad	Action
PAQ (Plan d'Assurance Qualité)	Non produit dans le repo bmad	Rédiger un PAQ projet (ou décliner le PAQ type Intersel pour bmad).
Document de spécifications	PRD + product brief + schémas ; pas un « DSL » au format Intersel	Soit déclarer le PRD + schémas comme équivalent spécifications, soit produire un document de spécifications formel (DSL) dérivé du PRD.
Document de conception préliminaire (DCP)	Architecture + schema-chiffrage + schema-relationnel, etc.	Regrouper sous un intitulé DCP ou référencer ces docs comme composants du DCP.
Conception détaillée (phpDoc/JsDoc)	Pas de génération phpDoc/JsDoc systématique	Si exigé pour le projet : ajouter les blocs et générer les dossiers.
Plan de tests	Cahier de recettes présent ; plan de tests (PT) non formalisé	Rédiger un plan de tests (couverture, types de tests, responsabilités) ou lier le cahier recettes au PT.
Dossier de tests unitaires / d'intégration	Pas de tests auto	Décision : soit dispense pour le POC (à acter dans le PAQ), soit mise en place de tests (PHPUnit, etc.).
Manuel utilisateur (MUL)	Absent	À prévoir en phase validation / livraison si contractuel.
Manuel d'installation / administration (MIL)	README + INSTALL partiels	Compléter ou rédiger un MIL formel (installation, config, sauvegardes).
CPRL (Cahier de procédure de recette)	Cahier recettes livraison (32 critères) présent	Aligner le nom / format avec le CPRL si exigé par la méthodologie.
Dossier de gestion de projet (wiki)	Docs dans `_bmad-output/planning-artifacts`	S'assurer que le dossier projet (wiki ou équivalent) contient ou pointe vers ces artefacts.
Calendrier et fournitures (lots)	Schema-chiffrage propose des lots	Compléter le calendrier (dates, livrables par lot, « Livré? », « Client OK/NOK ») comme en 4.2.1.
Réunions prévues	Non défini pour bmad	Renseigner les réunions (lancement, validation specs, recette, livraison, etc.).
Environnements (dev, intégration, validation)	Développement local + mode mock	Documenter les environnements (local, intégration si existant, validation, prod).
Gestion des anomalies (Mantis/FCEA)	Non mentionné pour bmad	Utiliser Mantis (ou outil équivalent) et lier les FCEA au planning comme en 4.6.3.
Versionnement (indices de version)	Git en place, nomenclature de version non précisée	Appliquer le versionnement défini dans le guide Intersel (numéro de version, tags).

3. Synthèse : quoi faire en priorité

3.1 Pour la méthodologie (côté Intersel)

- **Compléter la liste des livrables** : ajouter explicitement (dans le PAQ ou la liste des types de documents) les schémas utiles au chiffrage et à la conception : schéma relationnel / tables, schéma des workflows, schéma des intégrations, architecture applicative, infrastructure, WBS/lots, matrice RBAC.
- **Chiffrage et client** : nommer un « document de chiffrage » (ou fiche) et une « liste des questions en suspens / décisions client », et un livrable « synthèse projet pour lecture client » (ou l'inclure dans les DOC).
- **POC / Agile** : préciser les livrables allégés pour un POC et, si besoin, une Definition of Done POC.
- **Mise à jour** : remplacer les exemples de dates (2015) par un modèle « à compléter par projet », et clarifier les outils (Git vs SVN, Twitter vs Slack/wiki).

3.2 Pour le projet bmad (alignement méthodologie)

- **PAQ projet** : rédiger ou adapter le PAQ pour bmad (référence aux livrables réels : PRD, schémas, cahier recettes).
- **Dossier de gestion de projet** : s'assurer que le wiki (ou le repo) contient : planning, lots avec dates et livrés, réunions prévues, lien vers Mantis/FCEA.
- **Livrables manquants** : MUL et MIL si contractuels ; plan de tests formel ; conception détaillée (phpDoc/JsDoc) si exigée.
- **Recette** : garder le cahier recettes actuel et le faire correspondre au CPRL (nom et structure) si nécessaire.

Document à mettre à jour lorsque la méthodologie ou le périmètre du projet évoluent.