Direction des systèmes d'information

MISE EN ŒUVRE ET MAINTENANCE DE SITES D'INTRANETS ET DE PORTAILS MÉTIER CNRS

Cahier des Clauses Techniques Particulières

Livret 2/2 du CCTP n°26463

Accord-cadre n°26.14.027

Objet : Ce document décrit les prestations attendues et les unités d'œuvre pour la mise en œuvre et la maintenance de sites d'intranets et de portails métier CNRS.

1 SOMMAIRE

2 SYNTHÈSE DES PRESTATIONS 3
3 ELÉMENTS COMMUNS À TOUTES LES UNITÉS D'ŒUVRE 7
3.1 Eléments communs à toutes les unités d'œuvre 7
3.2 Complexité d'une unité d'œuvre 7
3.3 Coefficient de majoration ou minoration 12
3.4 Niveaux de services attendus pour la correction des anomalies applicatives 12
3.5 Pénalités de retard 13
3.6 Notion de tests 13
4 DESCRIPTION DES PRESTATIONS ATTENDUES 14
4.1 Prestation 1 : Lancement de l'accord cadre 14
4.2 Prestation 2 : Développement d'une nouvelle application ou refonte d'une application existante 15
4.3 Prestation 3 : Reprise de données lors d'une refonte d'une application existante 20
4.4 Prestation 4 : Maintien en conditions opérationnelles et de sécurité des applications après mise en service 23
4.5 Prestation 5 : Développement d'évolutions d'une application existante 28
4.6 Prestation 6 : Études et expertises 39
4.7 Prestation 7 : Migrations techniques (montées de version) 44
4.8 Prestation 8 : Conduite du changement 49
4.9 Prestation 9 : Tests spécifiques supplémentaires 52
4.10 Prestation 10 : Réversibilité et transfert de compétences 58
5 ANNEXES 62
5.1 Annexe 1 : Abréviations et Glossaire 62
5.2 Annexe 2 : Description des principaux plans et livrables documentaires 65

2 SYNTHÈSE DES PRESTATIONS

Les présentes prestations sont **assorties d'une obligation de résultat,** telle qu'elle est définie dans le CCAP.

Le titulaire s'oblige à garantir au CNRS des résultats conformes aux prestations attendues, périmètres associés et obligations requises, qui sont matérialisés par le présent cahier des charges.

Toutes les actions nécessaires à la bonne exécution de l'accord-cadre et de ses marchés subséquents par le titulaire qui ne sont pas expressément mises à la charge du CNRS par le Cahier des Clauses Administratives Particulières ou par le Cahier des Clauses techniques Particulières, sont à la charge du titulaire, quand bien même elles n'auraient pas été spécifiquement listées dans les pièces de l'accord-cadre et des marchés subséquents, qu'elles émanent du CNRS ou du titulaire.

Dispositions générales applicables à toutes les prestations susceptibles d'être demandées au titulaire :

Seules peuvent être facturées au CNRS les prestations pour lesquelles le CNRS a expressément demandé au titulaire de fournir un prix.

Les prestations, ainsi que les unités d'œuvre[1] (UO) sur lesquelles elles s'appuient, se déroulent en respectant les exigences du livret 1.

Les regroupements d'unités d'œuvre dans les prestations sont présentées de manière opérationnelle mais un marché subséquent visant une prestation déterminée, peut faire appel à des UO d'une autre prestation autant que de besoin.

	Prestation	Unité d'œuvre	Complexité	Déclenchement	Modalités de commande
1- Lancement de l'accord-cadre					
	Lancement de l'accord cadre	-	-	Une fois, en début de marché	Bon de commande, forfait BPU
2- Développement d'une nouvelle application ou refonte d'une application existante					
2.1	Lancement d'un projet de réalisation d'une nouvelle application	-	-	A survenance du besoin	Bon de commande, forfait BPU pour chacune des 4 applications initiales ou Marché subséquent avec engagement forfaitaire pour une nouvelle application
2.2	Conception de la plateforme technique d'une application	-	-	A survenance du besoin	Bon de commande, forfait BPU ou Marché subséquent avec engagement forfaitaire pour une nouvelle application
2.3	Conception et réalisation de la version initiale d'une application	-	-	A survenance du besoin	Bon de commande, forfait BPU chacune des 4 applications initiales Ou Marché subséquent avec engagement forfaitaire pour une nouvelle application
3- Reprise de données					

[1] Une unité d'œuvre est un forfait pour une tâche définie, assorti d'une obligation de résultat et d'un délai d'exécution.

	Prestation	Unité d'œuvre	Complexité	Déclenchement	Modalités de commande
3.1	Analyse des données d'une application	-	-	A survenance du besoin	Bon de commande, forfait BPU
3.2	Réalisation de la reprise des données d'une application	à définir	à définir	A survenance du besoin	Bon de commande, forfait BPU pour les 4 applications initiales Ou Marché subséquent avec engagement forfaitaire pour une nouvelle application
4- Maintien en conditions opérationnelles et de sécurité des applications après mise en service					
4.1	Prise en main d'une application réalisée hors marché	MCO_INTRO (prise en main d'une application)	Complexité de l'application TF / F / M / E / TE	A survenance du besoin	Bon de commande, forfait BPU par application
4.2	Maintien en conditions opérationnelles et de sécurité d'une application après mise en service	MCO_COR (correctif))	Complexité de l'application TF / F / M / E / TE	Récurrent mensuel (à compter de la mise en service de l'application – VA)	Bon de commande, forfait mensuel BPU par application
		MCO_EVOLUTION (petites évolutions)	-	Récurrent trimestriel	Bon de commande, forfait trimestriel BPU (petites évolutions) : 15j/h
		MCO_EVOLUTION_SUPPLEMENTAIRE (petites évolutions supplémentaires)		A survenance du besoin (reportable d'une année sur l'autre)	Bon de commande, forfait BPU : 10j/h
4.3	Sortie du périmètre du MCO d'une application	MCO_SORTIE (sortie d'une application du périmètre du marché sans réversibilité)	-	A survenance du besoin	Bon de commande, forfait BPU pour une application
5- Développement d'évolutions d'une application existante					
5.1	Conception d'évolutions	CONCEPTION (conception d'évolutions importantes)	TF / F / M / E / TE	A survenance du besoin dans le cas d'évolutions importantes	Bon de commande forfait BPU Ou Marché subséquent avec engagement forfaitaire
5.2	Réalisation d'évolutions	REA_FRONT (Réalisation d'évolutions de type Front End)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU ou Marché subséquent avec engagement forfaitaire
		REA_BACK (Réalisation d'évolutions de type Back End)	TF / F / M / E / TE		
6- Etudes et expertises					
	Etudes et expertises	ETD_EXP (étude ou expertise fonctionnelle et/ou technique)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU Ou Marché subséquent avec engagement forfaitaire
		ETD_UX_UI (étude d'UX-UI)	TF / F / M / E / TE		
7- Migrations techniques (montées de version)					
	Migrations techniques	ETD_MIG_TECH (étude de migration technique)	Complexité de montée de version TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU ou Marché subséquent avec engagement forfaitaire
		REA_MIG_TECH (réalisation de migration technique)	Complexité de montée de version		

	Prestation	Unité d'œuvre	Complexité	Déclenchement	Modalités de commande
			TF / F / M / E / TE		
8-Conduite du changement					
	Conduite du changement	FORM_ORG (organisation et conception de formation)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU ou Marché subséquent avec engagement forfaitaire
		FORM_ANI (animation de formation)	-		
		SUPPORT_AIDE (production de supports d'aide)	TF / F / M / E / TE		
9- Tests spécifiques supplémentaires					
	Tests automatisés	TEST_AUTO (tests automatisés)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU Ou Marché subséquent avec engagement forfaitaire
	Tests de performances	TEST_PERF (tests de performances)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU Ou Marché subséquent avec engagement forfaitaire
	Tests de charge	TEST_CHARGE (tests de charge)	TF / F / M / E / TE	A survenance du besoin	Bon de commande forfait BPU Ou Marché subséquent avec engagement forfaitaire
10- Réversibilité et transfert de compétences					
	Réversibilité	SORTIE_REV (Sortie du périmètre du marché avec réversibilité d'une application)	Complexité de l'application TF / F / M / E / TE	A survenance du besoin	Bon de commande, forfait BPU par application

Articulation des principales commandes de prestations et unités d'œuvre (non exhaustif) :

Au début du marché :

- **Prestation 1 : Lancement de l'accord cadre**

Pour chaque application de la refonte

- **Prestation 2 : Développement d'une nouvelle application ou refonte d'une application existante**
  - Prestation 2.1 : Lancement d'un projet de réalisation d'une nouvelle application

2.1.1 Prestation 2.1 : Lancement d'un projet de réalisation d'une nouvelle application

2.1.1.1 Présentation de la prestation

Cette prestation consiste à lancer le projet, initialiser l'environnement de travail, définir les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire spécifiques au projet, identifier et analyser les risques et présenter la planification du projet avec a minima les tâches suivantes :

- Prendre connaissance de l'ensemble des aspects techniques, métiers et organisationnels décrits dans le cahier des charges du projet ;
- Préparer et animer la réunion de lancement projet ;
- Préparer et animer la réunion de lancement technique ;
- Mettre en place toutes les conditions techniques et humaines qui sont nécessaires à la bonne réalisation des prestations :
  - Mettre en place l'organisation de l'équipe et de l'espace de travail ;

- Rendre opérationnel l'environnement de travail et de développement (postes de travail, serveur de développement, outillage de réalisation et de suivi de projet, …).

- Ajuster les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire et adapter le plan assurance qualité (PAQ) et le plan assurance sécurité (PAS) de l'accord cadre ;
- Préciser les principes d'urbanisation et d'intégration dans le SI (règles de base, liste des flux…) ;
- Définir le nombre d'ateliers de travail à prévoir ;
- Planifier les ateliers de travail à venir et identifier les disponibilités attendues des ressources CNRS et des ressources transverses (urbanisation, sécurité, méthodologie) ;
- Identifier et analyser les risques, définir les actions de réduction des risques associées.

Le titulaire propose un planning qui intègre les différentes prestations nécessaires au projet : a minima, « Conception de la plateforme technique » et « Conception et réalisation de la version initiale de l'application », le cas échéant « Reprise de données » et « Conduite du changement ».

De plus le titulaire découpe la phase de « Conception et réalisation de la version initiale de l'application » en grandes étapes/lots afin d'engager progressivement le budget alloué au projet, lisser les développements avec des périmètres plus petits, faciliter la planification et la réalisation des tests de recette par le CNRS.

Le titulaire inclut dans sa stratégie et sa feuille de route un plan pour aboutir à une mise en service des refontes sur des versions de composants à jour (aucun composant obsolète ou en fin de support), et dont aucune dépréciation n'est prévue.

Il planifie et intègre dans la feuille de route globale de l'équipe le planning de ce nouveau développement d'application, en tenant compte des développements des autres applications déjà en cours. Les développements des applications du périmètre initial à refondre sont souhaités en parallèle par le CNRS.

2.1.1.2 Livrables

Les principaux livrables attendus sont les suivants (cf description des livrables documentaires § 5.2.1) :

- Animation et compte rendu de réunions ;
- Composition de l'équipe projet opérationnelle ;
- Plan assurance qualité mis à jour ;
- Indicateurs de suivi du projet mis en place (actions, risques, événements…);
- Plan assurance sécurité mis à jour ;
- Tableau de suivi des risques projet ;
- Planning du projet et des ateliers.

2.1.1.3 Modalités de réception par le CNRS

La réception de la prestation est prononcée après :

- la validation du plan assurance qualité et du plan assurance sécurité de l'accord cadre mis à jour;
- la validation des comptes rendus des réunions de lancement,
- La validation du planning du projet et des ateliers.

    - Prestation 2.2 : Conception de la plateforme technique d'une application
    - Prestation 2.3 : Conception et réalisation de la version initiale d'une application
  - **Prestation 3 : Reprise de données lors d'une refonte d'une application existante**
    - Prestation 3.1 : Analyse des données d'une application
    - Prestation 3.2 : Réalisation de la reprise des données d'une application

Après la mise en service d'une application :

- **MCO/MCS Prestation 4.2 : Maintien en conditions opérationnelles et de sécurité d'une application après mise en service**
  - chaque mois : Unité d'œuvre MCO_COR - Correctif, *par application*
  - chaque trimestre : Unité d'œuvre MCO_EVOLUTION – Petites évolutions : Commande d'une enveloppe de 15 j/h, *pour l'ensemble des applications mises en service*

- **Prestation 5 : Développement d'évolutions d'une application existante**

  - Prestation 5.1 : Conception d'évolutions d'une application avec Unité d'œuvre CONCEPTION – Conception d'évolutions (dans le cas où la charge de conception est importante et définit la stratégie de réalisation, sinon dans la plupart des cas, la conception sera prise en charge au titre des UO de réalisation d'évolution)
  - Prestation 5.2 : Réalisation d'évolutions avec Unité d'œuvre REA_FRONT – Réalisation d'évolutions de type Front End et Unité d'œuvre REA_BACK – Réalisation d'évolutions de type back end

- **Prestation 7 : Migrations techniques à chaque fin de support des versions logicielles du socle, comme DRUPAL, PHP, PostgreSQL, Angular…**
  - Unité d'œuvre ETD_MIG_TECH – Etude de migration technique
  - Unité d'œuvre REA_MIG_TECH – Réalisation de migration technique

En fin de marché :

- **Prestation 10 : Réversibilité et transfert de compétences**
  - Unité d'œuvre SORTIE_REV – Sortie du périmètre du marché avec réversibilité

3 ELÉMENTS COMMUNS À TOUTES LES UNITÉS D'ŒUVRE

3.1 ELÉMENTS COMMUNS À TOUTES LES UNITÉS D'ŒUVRE

Les unités d'œuvre incluent, outre les prestations relevant directement des livrables attendus, les prestations d'accompagnement de cette mission principale :

- gestion de la relation avec le donneur d'ordres, incluant la tenue de réunions de suivi régulières dans lesquelles le prestataire est représenté au minimum par son responsable de projet,
- encadrement des prestataires chargés de l'exécution de la mission principale, incluant le suivi du respect des délais et celui de la consommation effective des ressources par rapport à la prévision,
- pilotage de l'activité, planning, gestion des risques, gestion documentaire, suivi de la qualité, reporting,
- formation, expertises et assistance auprès des prestataires chargés de l'exécution de la mission principale éventuellement nécessaires, au plan technique ou méthodologique, à la bonne exécution de cette dernière.

3.2 COMPLEXITÉ D'UNE UNITÉ D'ŒUVRE

En fonction du travail demandé autour d'une unité d'œuvre, la complexité peut varier.

Dans le cadre de cet accord-cadre cinq niveaux de complexité sont définis :

- très faible,
- faible,
- moyen,
- élevé,
- très élevé.

Chaque unité d'œuvre indique objectivement la manière dont est calculé son niveau de complexité, ce niveau de complexité peut résulter :

- soit d'un calcul spécifique, explicité le cas échéant dans chacune des unités d'œuvre décrites au présent document, et fondé sur des critères objectifs,
- soit de la complexité de l'application à laquelle l'unité d'œuvre est rattachée pour sa commande.

3.2.1 Complexité basée sur un calcul spécifique

Pour chacune des unités d'œuvre, un certain nombre de critères de dimensionnement sont identifiés.

Pour la plupart des unités d'œuvre, des niveaux de complexité sont définis. Ils sont déterminés en fonction d'un ou de plusieurs critères.

Pour quelques unités d'œuvre, le nombre de critères retenus étant supérieur à trois, le niveau de complexité dépend de la somme des « dimensionnements » établis pour chacun des critères :

- certains critères sont dimensionnés par catégorie de complexité liée à la nature des objets à traiter,
- pour d'autres, le dimensionnement dépend du nombre d'éléments observés pour ce critère ; le poids total du critère est égal au produit du nombre d'éléments par le poids unitaire des éléments utilisés.

Dans le cas où le niveau complexité de l'unité est basée sur un intervalle borné (avec un max), si le nombre d'éléments définissant le niveau de complexité est supérieur à l'intervalle, alors le calcul est décomposé et peut aboutir à plusieurs unités de complexités identiques ou différentes.

La note globale de complexité est la somme des notes individuelles de chaque élément.

Le niveau de complexité de l'UO est alors calculé à partir du tableau « niveau de complexité/critères » propre à chaque UO.

3.2.2 Complexité générale d'une application

La complexité de l'unité d'œuvre correspond à la complexité de l'application, c'est-à-dire à une évaluation pondérée d'un certain nombre de critères (maturité de l'application, complétude des éléments documentaires, …).

Cette complexité est susceptible d'évoluer dans le temps : par exemple en cas d'évolution majeure de l'application, de l'évolution de la documentation ou encore de la maturité de l'application/site.

Cette note de complexité peut être réévaluée annuellement, à chaque date anniversaire de l'accord cadre.
Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Nombre de types de contenu	Simple (1–5)	Site vitrine avec 3 types (Article, Page, Événement).	1
	Moyen (6–15)	Site avec 10 types de contenu.	2
	Complexe (16-20)	Application métier avec 18 types de contenu.	3
	Très Complexe (>20)	Application métier avec 21+ types de contenu.	4
Modules custom	Aucun	-	0
	Par module simple	Modules basiques (ex: hook simple, champ calculé).	2
	Par module complexe	Modules avec logique métier (exemple workflow custom).	3
	Par module très complexe	Intégration ERP, traitement batch…	5
Workflows	Aucun	-	0
	Par workflow simple	Workflow "Brouillon → Publié".	1
	Par workflow complexe	Workflows avec transitions conditionnelles.	2
	Par workflow très complexe	Workflows multi-états avec validations dynamiques.	3
Nombre de rôles/permissions	Simple (1–3)	Rôles basiques (Admin, Éditeur, Visiteur), restrictions par champs…	1
	Moyen (4–6)		2
	Complexe (>6)		3
Intégrations externes	Aucune	-	0
	Par intégration simple	API REST basique (ex: envoi d'emails).	2
	Par intégration complexe	Intégration avec LDAP, CRM, ERP.	4
	Par intégration très complexe	Intégration avec des protocoles propriétaires (SOAP, RFC).	6
Complétude de la documentation	Complète (≈100%)	Documentation technique et fonctionnelle à jour.	-2
	Partielle (50–99%)	Documentation existante mais incomplète.	0
	Minimale (<50%)	Peu ou pas de documentation	2
Maturité de l'application	Mature (>3 ans)	Application stable, peu d'anomalies.	-1

Elément particulier	Niveau de complexité	Exemple	Note de complexité
	Stable (1–3 ans)	Application en production, quelques anomalies.	0
	Jeune (<1 an)	Application récente, risques d'anomalies	1
Nombre de templates Twig custom	Simple (1–20)	Templates pour les types de contenu	1
	Moyen (21–40)		2
	Complexe (41-60)		3
	Très complexe (>60)		4

La complexité générale de l'application se calcule en sommant les notes individuelles et se déduit avec le tableau suivant :

Niveau Complexité	Critères
Très faible	Note de complexité entre 1 et 15
Faible	Note de complexité entre 16 et 30
Moyen	Note de complexité entre 31 et 45
Elevé	Note de complexité entre 46 et 60
Très élevé	Note de complexité > 60

3.2.3 Complexité des unités d'œuvre de montée de version

On appelle « patch » :

- une nouvelle version apportant de petites corrections de bugs sans nouvelle fonctionnalité et sans impact sur l'existant, cela correspond en général au passage d'une version X.Y.Z à une version Z+1.

On appelle « version mineure » :

- une nouvelle version apportant des corrections ou de petites évolutions, avec une rétro-compatibilité par rapport à la version précédente. Cela correspond en général au passage d'une version X.Y à une version X.Y+1. Une version mineure peut introduire des dépréciations ou des changements incompatibles qui sont en général à la marge ou simples à corriger.

On appelle « version majeure » :

- une nouvelle version apportant d'importantes évolutions pouvant impacter l'existant, cela correspond en général au passage d'une version X.Y à une version X+1.

Lorsqu'il est possible de passer dans une version majeure cible sans étapes intermédiaires indispensables à la migration car indiquées comme telles par l'éditeur (autres versions majeures intermédiaires**), seule la version cible sera prise en compte dans l'évaluation de la note** (ex PG 12 vers PG 17 sans impact pour une application de complexité moyenne → on compte 1 seule fois 4 points).

La migration vers une version majeure doit se faire vers la dernière version disponible de cette version majeure supportée par le Cadre de Cohérence Technique du CNRS (CCT). Par exemple migrer de PHP 7.2 vers PHP 8 signifie migrer vers PHP 8.2.Z (version supportée par RedHat de PHP = 8.2, dernière version = 8.2.Z).

De même, la migration du système de base de données vers une version majeure comprend les fonctionnalités des versions mineures.

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Système de base de données			
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	2
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	4
	Application de complexité 'très élevée'	Version majeure	6
Migration de système de base de données	Application de complexité 'très faible' ou 'faible'	Etude	10
	Application de complexité 'moyenne' ou 'élevée'	Etude	15
	Application de complexité 'très élevée'	Etude	20
Socle applicatif : langage PHP			
Montée de version mineure	Application de complexité 'très faible' ou 'faible'	Version mineure	2
	Application de complexité 'moyenne' ou 'élevée'	Version mineure	3
	Application de complexité 'très élevée'	Version mineure	4
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	4
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	5
	Application de complexité 'très élevée'	Version majeure	6
Socle applicatif : Drupal CoRe			
Montée de version mineure	Application de complexité 'très faible' ou 'faible'	Version mineure	2
	Application de complexité 'moyenne' ou 'élevée'	Version mineure	3
	Application de complexité 'très élevée'	Version mineure	4
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	6
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	8
	Application de complexité 'très élevée'	Version majeure	10
Socle applicatif : Modules contrib Drupal			
Montée de version d'un module	Application de complexité 'très faible' ou 'faible'	Nombre de modules contrib	1
	Application de complexité 'moyenne' ou 'élevée'	Nombre de modules contrib	2
	Application de complexité 'très élevée'	Nombre de modules contrib	3
Socle applicatif : framework, Composer			
Montée de version mineure	Application de complexité 'très faible' ou 'faible'	Version mineure	2
	Application de complexité 'moyenne' ou 'élevée'	Version mineure	3
	Application de complexité 'très élevée'	Version mineure	4

Elément particulier	Niveau de complexité	Unité	Note de complexité
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	6
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	8
	Application de complexité 'très élevée'	Version majeure	10
Frontend : Angular			
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	2
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	3
	Application de complexité 'très élevée'	Version majeure	4
Serveur d'application ou système d'exploitation			
Montée de version majeure	Application de complexité 'très faible' ou 'faible'	Version majeure	4
	Application de complexité 'moyenne' ou 'élevée'	Version majeure	5
	Application de complexité 'très élevée'	Version majeure	6
Librairie tierce (par exemple JS JQuery…)			
Montée de version pour l'ensemble des librairies dont le changement de version est sans impact autre que le fichier de déclaration de dépendance	Application de complexité 'très faible' ou 'faible'	Version	1
	Application de complexité 'moyenne' ou 'élevée'	Version	2
	Application de complexité 'très élevée'	Version	3
Montée de version d'une librairie dont le changement de version entraine des impacts significatifs dans le code applicatif	Application de complexité 'très faible' ou 'faible'	Version	4
	Application de complexité 'moyenne' ou 'élevée'	Version	5
	Application de complexité 'très élevée'	Version	6
Migration de technologie (changement de librairie)	Application de complexité 'très faible' ou 'faible'	Etude	6
	Application de complexité 'moyenne' ou 'élevée'	Etude	8
	Application de complexité 'très élevée'	Etude	10
Navigateur			
Mise en place de la compatibilité avec un navigateur non supporté jusqu'à présent par l'application	Application de complexité 'très faible' ou 'faible'	Etude	4
	Application de complexité 'moyenne' ou 'élevée'	Etude	5
	Application de complexité 'très élevée'	Etude	6

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau Complexité	Critères
Très faible	Note de complexité <= 5
Faible	5 < Note de complexité <= 10
Moyen	10 < Note de complexité <= 15
Elevé	15 < Note de complexité <= 20

Niveau Complexité	Critères
Très élevé	20 < Note de complexité <= 25

3.3 COEFFICIENT DE MAJORATION OU MINORATION

Des coefficients de majoration ou minoration du coût de chaque UO peuvent être appliqués par le CNRS :

3.3.1 Coefficient de majoration

Le coefficient de majoration s'applique à toute extension par rapport au référentiel nominal d'exigences méthodologiques ou techniques défini dans le présent CCTP (ajout de tests automatisés complémentaires, ajout de règle de qualité de code, exigence de performance supérieure...).

Ce coefficient a pour maximum 1,25.

3.3.2 Coefficient de minoration

Des coefficients de minoration, variant entre 0,1 et 0,9, doivent être utilisés, dans les cas suivants :

- réalisation de tâches répétitives ou si la tâche à réaliser dérive d'une fonctionnalité déjà existante (comme par exemple une fonctionnalité mutualisée par polymorphisme, une fonctionnalité proche, …),
- certains livrables qui ne sont pas à fournir,
- une tâche normalement prévue à la charge du titulaire sera effectuée par le CNRS.

D'autres cas spécifiques sont possibles et mentionnés le cas échéant pour des UO particulières.

3.4 NIVEAUX DE SERVICES ATTENDUS POUR LA CORRECTION DES ANOMALIES APPLICATIVES

Quelle que soit la prestation (réalisation version initiale, évolution, montée de version…), les niveaux de service suivants sont attendus.

Les délais indiqués ci-dessous correspondent au temps écoulé entre la déclaration confirmée au titulaire de l'anomalie dans l'outil de gestion des anomalies et des évolutions du CNRS et la livraison effective par le titulaire de la correction dans l'environnement de livraison du CNRS.

3.4.1 Niveaux de services attendus pour la Vérification d'Aptitude (environnement de recette)

Les anomalies[2] déclarées par les équipes CNRS doivent être prises en compte, corrigées et livrées par le titulaire dans les délais suivants :

- anomalies bloquantes : sous 2 jours ouvrés maximum ;
- anomalies majeures et mineures : délai à définir conjointement avec le CNRS en fonction des contraintes opérationnelles.

3.4.2 Niveaux de services attendus pour la Vérification de Service Régulier, garantie et MCO (environnement de production)

Les anomalies déclarées par les équipes CNRS doivent être prises en compte, corrigées et livrées par le titulaire dans les délais suivants :

- anomalies bloquantes : sous 2 jours ouvrés maximum ;
- anomalies majeures : sous 5 jours ouvrés maximum ;
- anomalies mineures : délai à définir conjointement avec le CNRS en fonction des contraintes opérationnelles.

[2] La définition d'une anomalie et sa typologie sont décrites dans le glossaire au § 5.1.2

3.5 PÉNALITÉS DE RETARD

Pour la prestation de Maintien en conditions opérationnelles et de sécurité des applications après mise en service :

En cas de dépassement des délais de correction d'anomalies indiqués dans les niveaux de service attendus, les pénalités applicables sont les suivantes :

P = (V*R)/30, avec V = montant mensuel du MCO pour les composants impactés et R = nombre de jours de retard.

Pour toutes les autres prestations :

En cas de dépassement des délais de correction d'anomalies indiqués dans les niveaux de service attendus ou convenus avec le pouvoir adjudicateur, les pénalités applicables sont les suivantes :

P = (V*R)/90, avec V = montant de la commande et R = nombre de jours de retard.

3.6 NOTION DE TESTS

Les tests automatisés unitaires ou d'intégration doivent être écrits en utilisant les frameworks de tests conformes aux exigences du CCTP livret 1. Un test de performance unitaire d'un fragment applicatif ou d'un composant sera également écrit selon le même procédé.

La réalisation de tests automatisés suivant les exigences minimales du CCTP livret 1 est prévue par défaut dans l'UO de réalisation et ne fait pas l'objet de l'application d'un coefficient de majoration.

4 DESCRIPTION DES PRESTATIONS ATTENDUES

4.1 PRESTATION 1 : LANCEMENT DE L'ACCORD CADRE

La phase de lancement de l'accord cadre est ici décrite de manière théorique dans son ordonnancement, car elle est propre à la démarche du titulaire.
Il appartient donc au titulaire de proposer et mettre en œuvre un plan et une démarche permettant de sécuriser la mise en place du dispositif.

4.1.1 Présentation de la prestation

Il s'agit pour le titulaire de lancer le projet, de définir et rendre opérationnels l'ensemble des processus et pratiques liés à la gouvernance et la gestion des prestations, ainsi que l'outillage associé :

- Prendre connaissance de l'ensemble des aspects techniques, métiers et organisationnels au sein desquelles le titulaire doit assurer les prestations décrites dans le présent CCTP ;
- Préparer et animer la réunion de lancement de l'accord-cadre ;
- Préparer et animer la réunion de lancement technique ;
- Mettre en œuvre la gouvernance nécessaire au pilotage des prestations à venir dans l'exécution du marché, ainsi que les supports inhérents : supports des différents comités, indicateurs, tableaux de bord ;
- Ajuster les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire et finaliser le plan assurance qualité (PAQ) et le plan assurance sécurité (PAS) fournis dans l'offre, ainsi que les indicateurs de suivi de projet (plan de mesure) ;
- Mettre en place toutes les conditions techniques et humaines qui sont nécessaires à la bonne réalisation des prestations :
  - Mettre en place l'organisation de l'équipe et de l'espace de travail ;
  - Rendre opérationnel l'environnement de travail et de développement (postes de travail, serveur de développement, outillage de réalisation et de suivi de projet, …) ;
  - Donner les autorisations d'accès appropriées pour les outils partagés avec le CNRS ;
- Finaliser la première version du plan de réversibilité pour la future sortie du marché.

4.1.2 Livrables

Les principaux livrables attendus sont les suivants (cf description des livrables documentaires § 5.2.1) :

- Animation et compte rendu de la réunion de lancement ;
- Plan assurance qualité (PAQ) finalisé et modèles de documents opérationnels : organisation, procédures opérationnelles, modalités et aspects pratiques ;
- Plan d'assurance sécurisé (PAS) finalisé : organisation, acteurs, processus, mesures et procédures mis en œuvre par le titulaire pour assurer un niveau de sécurité en adéquation avec les exigences du contrat ;
- Plan de mesure finalisé, avec indicateurs de suivi de l'accord cadre mis en place ;
- Equipe du titulaire opérationnelle pour les activités de TMA ;
- Environnement de développement opérationnel pour les équipes du titulaire ;
- Rapport d'étonnement succinct présentant notamment les grands axes d'améliorations possibles ;
- Plan de réversibilité complet : organisation, démarche et planning projet, acteurs, éléments à transférer par le titulaire pour assurer la réversibilité des services au CNRS ou à un tiers désigné par le CNRS, dans de bonnes conditions, c'est-à-dire, sans impact majeur pour les utilisateurs des services.

4.1.3 Modalités de réception par le CNRS

La réception de la prestation est prononcée après :

- La réception et la validation de l'ensemble des livrables dont la validation de la pertinence et de l'exhaustivité du plan assurance qualité et du plan assurance sécurité projet ;
- la validation de l'ensemble des opérations de lancement.

4.2 PRESTATION 2 : DÉVELOPPEMENT D'UNE NOUVELLE APPLICATION OU REFONTE D'UNE APPLICATION EXISTANTE

Il s'agit ici de la refonte des applications identifiées dans le livret 1 et ses annexes, mais également

- un nouveau projet d'application,
- une refonte d'une application existante, depuis une même technologie ou une technologie différente, qu'elle soit ou non initialement dans le périmètre de ce marché,

4.2.1 Prestation 2.1 : Lancement d'un projet de réalisation d'une nouvelle application

4.2.1.1 Présentation de la prestation

Cette prestation consiste à lancer le projet, initialiser l'environnement de travail, définir les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire spécifiques au projet, identifier et analyser les risques et présenter la planification du projet avec a minima les tâches suivantes :

- Prendre connaissance de l'ensemble des aspects techniques, métiers et organisationnels décrits dans le cahier des charges du projet ;
- Préparer et animer la réunion de lancement projet ;
- Préparer et animer la réunion de lancement technique ;
- Mettre en place toutes les conditions techniques et humaines qui sont nécessaires à la bonne réalisation des prestations :
  - Mettre en place l'organisation de l'équipe et de l'espace de travail ;
  - Rendre opérationnel l'environnement de travail et de développement (postes de travail, serveur de développement, outillage de réalisation et de suivi de projet, …).
- Ajuster les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire et adapter le plan assurance qualité (PAQ) et le plan assurance sécurité (PAS) de l'accord cadre ;
- Préciser les principes d'urbanisation et d'intégration dans le SI (règles de base, liste des flux…) ;
- Définir le nombre d'ateliers de travail à prévoir ;
- Planifier les ateliers de travail à venir et identifier les disponibilités attendues des ressources CNRS et des ressources transverses (urbanisation, sécurité, méthodologie) ;
- Identifier et analyser les risques, définir les actions de réduction des risques associées.

Le titulaire propose un planning qui intègre les différentes prestations nécessaires au projet : a minima, « Conception de la plateforme technique » et « Conception et réalisation de la version initiale de l'application », le cas échéant « Reprise de données » et « Conduite du changement ».

De plus le titulaire découpe la phase de « Conception et réalisation de la version initiale de l'application » en grandes étapes/lots afin d'engager progressivement le budget alloué au projet, lisser les développements avec des périmètres plus petits, faciliter la planification et la réalisation des tests de recette par le CNRS.

Le titulaire inclut dans sa stratégie et sa feuille de route un plan pour aboutir à une mise en service des refontes sur des versions de composants à jour (aucun composant obsolète ou en fin de support), et dont aucune dépréciation n'est prévue.

Il planifie et intègre dans la feuille de route globale de l'équipe le planning de ce nouveau développement d'application, en tenant compte des développements des autres applications déjà en cours. Les développements des applications du périmètre initial à refondre sont souhaités en parallèle par le CNRS.

4.2.1.2 Livrables

Les principaux livrables attendus sont les suivants (cf description des livrables documentaires § 5.2.1) :

- Animation et compte rendu de réunions ;
- Composition de l'équipe projet opérationnelle ;
- Plan assurance qualité mis à jour ;
- Indicateurs de suivi du projet mis en place (actions, risques, événements…);
- Plan assurance sécurité mis à jour ;
- Tableau de suivi des risques projet ;
- Planning du projet et des ateliers.

4.2.1.3 Modalités de réception par le CNRS

La réception de la prestation est prononcée après :

- la validation du plan assurance qualité et du plan assurance sécurité de l'accord cadre mis à jour;
- la validation des comptes rendus des réunions de lancement,
- La validation du planning du projet et des ateliers.

4.2.2 Prestation 2.2 : Conception de la plateforme technique d'une application

Il s'agit ici de concevoir la plateforme technique qui sera hébergée sur un cloud privé (IaaS) fourni par le CNRS et dont la mise en œuvre sera réalisée par les équipes du CNRS.

4.2.2.1 Présentation de la prestation

Le titulaire réalise l'ensemble des opérations nécessaires à la conception de la plateforme technique sur le périmètre du présent accord-cadre.

De manière non exhaustive, il s'agit pour le titulaire de réaliser a minima les tâches suivantes, en accord avec le CNRS :

- Animer ou participer à des ateliers avec l'équipe DSI du CNRS (architecture, sécurité, exploitation etc.) ;
- Définir l'infrastructure (dimensionnement VM, RAM, CPU, stockage, etc.) pour les plateformes d'intégration, recette et production, ainsi que les configurations nécessaires au regard de l'architecture applicative et des niveaux de services attendus ;
- Définir l'architecture (Drupal, système de mise en cache, accélérateur de site, brique de recherche etc.) au regard des exigences techniques et de sécurité du CNRS ainsi des niveaux de services attendus ;
- Rédiger les procédures d'installation et de paramétrage des composants logiciels (Drupal, système de mise en cache, accélérateur de site, brique de recherche etc.) (cf. la description du contenu au § 5.2.3.3) ;
- Assurer le support aux équipes CNRS en charge de la mise en œuvre ;
- Initialiser les manuels d'exploitation (cf. la description du contenu au § 5.2.3.2) ;
- Installer et configurer l'environnement d'intégration (cf. Livret1_Annexe_RACI)

Nota : certaines activités déjà réalisées lors de la mise en œuvre de la première application ne seront plus à réaliser pour les applications suivantes. Le titulaire en tient compte pour l'estimation des charges des quatre applications dans le BPU.

4.2.2.2 Livrables

Les principaux livrables attendus sont les suivants :

- Compte rendu des ateliers animés ;
- Dossier d'architecture technique (DAT) ;
- Manuels d'installation et de paramétrage des composants logiciel ;

- Scripts d'installation éventuels ;
- Dossier d'exploitation ;
- Infrastructure et environnements opérationnels, mis en œuvre par le titulaire (développement, intégration) ou le CNRS (recette, production), cf répartition des responsabilités dans Livret1_Annexe_RACI.

4.2.2.3 Modalités de réception par le CNRS

La réception de la prestation est prononcée après :

- La réception et la validation de l'ensemble des livrables ;
- Infrastructure et environnements opérationnels.

4.2.3 Prestation 2.3 : Conception et réalisation de la version initiale d'une application

4.2.3.1 Présentation de la prestation

Cette prestation consiste à paramétrer et/ou développer la version initiale de l'application répondant aux exigences fonctionnelles et techniques décrites dans le cahier des charges. Le titulaire doit proposer au CNRS une solution s'appuyant au maximum sur du paramétrage. Le titulaire pourra proposer des solutions basées sur des modules Drupal ou du développement spécifique. La décision sera prise conjointement avec le CNRS en challengeant la confiance dans le module vs la complexité/maintenabilité du développement.

Le titulaire propose un découpage de la prestation en itérations successives avec des versions incrémentales, qui correspondent à de grandes étapes/lots du projet, comme par exemple :

- Conception (Web design, spécifications)
- MVP (Minimum Viable Product) : c'est-à-dire fonctionnalités utilisateurs minimales
- Intégration dans le SI (Authentification via Janus, Synchronisation avec les référentiels CNRS via Webservices…
- Fonctionnalités avancées (Back-office applicatif et gestion des droits, recherche, antivirus sur upload…)
- Industrialisation (automatisation, CI/CD, supervision…)

Les phases de tests et de documentation sont réalisées au fur et à mesures de ces étapes.

Le CNRS souhaite par ailleurs des livraisons régulières pour vérifier au fur et à mesure la conformité au besoin exprimé.

Cette prestation comprend les phases de conception, réalisation et livraison du paramétrage et/ou des développements de la solution cible, avec a minima les tâches suivantes :

- Suivre et mettre à jour le planning et les indicateurs du projet ;
- Animer des ateliers fonctionnels ou techniques pour concevoir la solution applicative et rédiger les comptes rendus de ces ateliers :
  - Ateliers de conception fonctionnelle, UX/UI avec la MOA, les utilisateurs métiers et la MOE ;
  - Ateliers de conception d'intégration dans le SI, avec la DSI, pour définir la solution d'échange avec le système d'intermédiation pour chacun des flux ;
- Concevoir la solution fonctionnelle :
  - Co-construire avec le CNRS la solution fonctionnelle (fonctionnalités, ergonomie, graphisme, accessibilité, …) ;
  - Définir les options permettant de garantir la modularité et l'évolutivité des fonctionnalités ;
- Concevoir la solution Drupal :
  - Définir et ajuster l'architecture applicative et technique des données et des traitements (découpage et articulation des composants logiciels...) ;
  - Définir les options permettant de garantir une modularité optimale des traitements, le niveau de paramétrage attendu, le niveau et le format des logs et les performances optimales ;

- Réaliser wireframes, maquettes ou prototypes prenant en compte les règles d'accessibilité, d'ergonomie et la charte graphique CNRS et permettant l'évaluation/validation du besoin par les utilisateurs ;
- Rédiger la documentation technico-fonctionnelle et en particulier :
  - les spécifications fonctionnelles détaillées(cf. la description du contenu au § 5.2.2.1),
  - les spécifications techniques détaillées (cf. la description du contenu au § 5.2.2.2),
  - le dossier d'architecture logicielle (cf. la description du contenu au § 5.2.3.1),
  - la conception technique détaillée de la solution (cf. la description du contenu au § 5.2.2.3) ;
- Réaliser le paramétrage et/ou les développements spécifiques constituant la solution ;
- Fournir aux équipes DSI toutes les informations nécessaires pour la mise en place de la CI/CD dans la forge de la DSI du CNRS. Cela consiste notamment à fournir (liste non exhaustive) les procédures pour lancer la compilation et les tests dans les runners Gitlab, configurer l'analyse qualité Sonar, documenter l'installation et les mises à jour afin de les automatiser avec Ansible, etc.
- Pousser régulièrement le code source dans la forge du CNRS sans attendre la fin de la réalisation ;
- Analyser et prendre en compte les résultats de l'intégration continue et de l'analyse Sonar de la forge du CNRS ;
- Rédiger le plan de test
- Réaliser les tests suivants[3] sur différentes plateformes (OS, navigateurs) depuis l'environnement de développement et l'environnement d'intégration :
  - tests unitaires automatisés et manuels,
  - tests d'intégration,
  - tests de validation,
  - tests de non régression de l'application,
  - tests de performance unitaires et/ou plus globaux,
  - tests de charge (validation du dimensionnement de la plateforme).
  - tests de sécurité (version majeure applicative).
- Vérifier par contrôle qualité interne au titulaire, le respect des normes de développement et de documentation ;
- Spécifier les procédures permettant l'installation, la mise en exploitation et l'exploitation de l'application en prenant en compte le déploiement via Ansible ;
- Rédiger les manuels d'installation et d'exploitation (cf. la description du contenu aux § 5.2.3.2 et § 5.2.3.3) ;
- Livrer la version initiale dans l'environnement de livraison du CNRS en utilisant Ansible (cf. §5.4.3 du CCTP - Livret 1) ;
- Assister l'équipe technique du CNRS pour assurer l'installation dans l'environnement de recette ;
- Prendre en compte les éventuelles anomalies[4] de recette détectée par le CNRS et procéder aux relivraisons nécessaires (phase de Vérification d'Aptitude) ;
- Assister si besoin les équipes CNRS, lors de l'installation et de la mise en exploitation dans l'environnement de production ;
- Prendre en compte les éventuelles anomalies de production détectées par le CNRS et procéder aux relivraisons nécessaires (phases de Vérification de Service Régulier et de garantie) ;

Les wireframes/maquettes et la documentation technico-fonctionnelle devront être transmises et validées par le CNRS en amont de la réalisation des développements et de façon incrémentale afin de s'assurer de la bonne compréhension du besoin par le titulaire.

4.2.3.2 Livrables

Pendant et à l'issue de la phase de conception :

- Animation et compte rendu des ateliers menés ;
- Wireframes, maquettes ou prototypes
- Spécifications fonctionnelles détaillées créé ou mis à jour (cf. la description du contenu au § 5.2.2.1) ;
- Spécifications techniques détaillées créés ou mis à jour (cf. la description du contenu au § 5.2.2.2) ;

[3] Cf. la définition des différents types de tests dans le glossaire au § 5.1.2
[4] La définition d'une anomalie est décrite dans le glossaire au § 5.1.2

- Dossier d'architecture logicielle créé (cf. la description du contenu au § 5.2.3.1),
- Conception technique détaillée de la solution créé ou mis à jour (cf. la description du contenu au § 5.2.2.3) ;

Pendant et à l'issue de la phase de réalisation :

- Spécifications fonctionnelles détaillées mis à jour (cf. la description du contenu au § 5.2.2.1), ;
- Wireframes, maquettes, prototypes,
- Spécifications techniques détaillées mises à jour (cf. la description du contenu au § 5.2.2.2) ;
- Conception technique détaillée de la solution mis à jour (cf. la description du contenu au § 5.2.2.3) ;
- Dossier d'architecture logicielle mis à jour (cf. la description du contenu au § 5.2.3.1),
- Rapport de tests unitaires (noms, date, numéro de version) ou si possible rapport d'analyse de code Sonar ;
- Manuel d'installation comprenant la configuration permettant la mise au point des playbooks par le CNRS (cf. la description du contenu au § 5.2.3.3) ;
- Manuel d'exploitation créé ou mis à jour (cf. la description du contenu au § 5.2.3.2) ;
- Code source poussé régulièrement dans la forge du CNRS sans attendre la fin de la réalisation, afin de valider la chaine d'intégration continue et de suivre l'évolution de la qualité du code ;
  - Exécutable livré selon les modalités spécifiques à la plateforme (cf. § 5.5 du CCTP – Livret 1) ;
  - Ensemble de tous les fichiers, y compris source, correspondant à la branche ou à l'étiquette livrée et permettant la génération de cet exécutable ;
- Paramétrage mis en œuvre ;
- Scripts d'installation éventuels.

Les éléments suivants sont transmis au CNRS au fur et à mesure de leur production mais ne sont pas considérés comme des livrables nécessitant une validation du CNRS :

- Plans, dossiers, jeux d'essais et résultats de tests unitaires, d'intégration, de validation, de non-régression, de performance et de sécurité initiée en fin de phase de conception puis complétés en phase de tests.

4.2.3.3 Modalités de réception par le CNRS

La recette porte, outre sur le constat de remise des livrables, sur le respect des normes techniques de programmation, de documentation et d'exploitation, ainsi que sur la conformité au dossier de spécifications détaillées.

La réception de la prestation est prononcée après la vérification que tous les éléments sont réalisés.

Lors de la recette (Vérification d'Aptitude), le CNRS peut, a minima :

- vérifier la complétude des éléments livrés ;
- vérifier le bon déroulement de l'installation de l'application livrée, sur l'environnement de recette ;
- exécuter des tests de recette sur le périmètre applicatif de la livraison (dans l'environnement de recette) ;
- vérifier la qualité du code (application des normes, détection de problèmes potentiels de qualité ou de sécurité du code par Sonar…), la complétude des tests et jeux de tests du titulaire, le passage avec succès de ces tests dans l'environnement du titulaire ;
- vérifier la « robustesse » de la solution et la montée en charge des utilisateurs ;
- vérifier l'exhaustivité et la pertinence du contenu de la documentation livrée.

Lors de la Vérification de Service Régulier (VSR), le CNRS vérifie l'utilisation de l'application dans des conditions normales en environnement de production.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[5] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

[5] La gravité/typologie d'une non-conformité/anomalie est définie dans le glossaire au § 5.1.2

4.3 PRESTATION 3 : REPRISE DE DONNÉES LORS D'UNE REFONTE D'UNE APPLICATION EXISTANTE

4.3.1 Prestation 3.1 : Analyse des données d'une application en vue d'une reprise

Présentation

Cette prestation consiste à analyser le contenu d'une application existante en vue d'alimenter une application dans le périmètre de ce marché, en réalisant une reprise des contenus.

Il s'agit pour le titulaire d'analyser les contenus existants d'une application désignée par le CNRS, de qualifier les besoins en termes de reprise, de définir la stratégie de reprise, et de proposer une catégorisation selon les types et les thématiques des contenus concernés.

Cette prestation doit permettre de dimensionner la complexité de la reprise pour évaluer ensuite la prestation de reprise.

Eléments en entrée

- Périmètre, format des données source et volume de l'application dont le contenu est à reprendre ;
- Plan de navigation/arborescence de l'application à reprendre ;
- Objectifs attendus par le CNRS (ex : fusion de différentes sections/catégories du site, optimisation du nombre de pages du site) ;
- Si existe, taxonomie de l'application existante ;

Actions à réaliser

- Analyser la demande du CNRS ;
- Animer des ateliers avec le CNRS pour affiner le besoin et rédiger les comptes rendus de ces ateliers ;
- Définir la stratégie et les modalités de reprise des données ;
- Proposer une démarche de catégorisation pour validation par le CNRS (ex : critères de catégorisation, méthodologie et/ou outil utilisé) ;
- Catégoriser les contenus à reprendre ;
- Estimer la nature et la quantité des unités d'œuvre définies à la Prestation 3.2 nécessaires à l'exécution de la reprise ;
- Taxonomie :
    - Ajuster la taxonomie cible avec le CNRS : typologie de contenu, thématiques…
    - Proposer au CNRS des règles de correspondance pour les pages qui n'ont pas de taxonomie
    - Elaborer une table de correspondance entre la taxonomie de l'application existante et la taxonomie cible
- Présenter les résultats de l'analyse au CNRS.

Livrables

- Nature et quantité des unités d'œuvre définies à la Prestation 3.2 nécessaires à l'exécution de la reprise
- Atelier de présentation des résultats de l'analyse
- Documentation présentant la stratégie et les modalités de reprise des données ;
- Documentation avec la catégorisation des types de contenus et contenus à reprendre par type de contenu et par thématique et avec les volumétries
- Table de correspondance entre les contenus de l'application existante et de l'application cible
- Table de correspondance entre la taxonomie de l'application existante et de l'application cible
- Plan de navigation/arborescence cible

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- le respect des normes de documentation,
- sur la conformité aux documents d'entrée,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée,

et – le cas échéant – aux résultats des échanges avec les représentants de la maitrise d'œuvre.

4.3.2 Prestation 3.2 : Réalisation de la reprise des données d'une application

4.3.2.1 Présentation de la prestation

Cette prestation consiste à ajouter des contenus à une application à partir de contenus d'une autre application, existante.

Cette prestation peut être déclenchée pendant la période d'exécution de l'accord-cadre, suite à la refonte d'une des applications du périmètre initial ou lors de l'ajout d'application au périmètre.

Lors de la reprise de contenus, le titulaire doit prévoir l'outillage nécessaire (interface, utilitaire de reprise de données...).

Le CNRS commandera la Prestation 3.1 : Analyse des données d'une application.

Puis en fonction des livrables de cette prestation,

- Pour la reprise des données du périmètre initial (Intranet CNRS, Portail DU, Portail RH et Bourse aux matériels), le CNRS passera des bons de commandes avec les UOs nécessaires,
- le CNRS passera un marché subséquent dans les autres cas.

Cette prestation consiste à

- Rédiger les spécifications, effectuer le paramétrage et les développements éventuels permettant la récupération et/ou l'intégration des données ; réaliser les tests nécessaires ; livrer le paramétrage et les développements éventuels au CNRS ;
- Publier dans la nouvelle application les contenus repris ;
- Appliquer la table de correspondance taxonomie issue de la prestation « Analyse des données en vue d'une reprise » entre les contenus sources et les contenus cibles

4.3.2.2 Livrables

Les principaux livrables attendus sont les suivants :

- Documentations fonctionnelles et techniques mises à jour lorsque nécessaire ;
- Sources et/ou paramétrage mis en œuvre ;
- Plan de tests et résultats des tests déroulés ;
- Table de correspondance des contenus entre l'application existante et l'application cible mise à jour
- Table de correspondance entre la taxonomie de l'application existante et de l'application cible mise à jour
- Application initialisée avec les contenus repris, indexés en se basant sur la table de correspondance taxonomie
- Rapport de migration quantifié avec taux de succès et taux de rejets.

4.3.2.3 Modalités de réception par le CNRS

La réception de la prestation est prononcée après :

- la validation de la documentation
- la validation de l'application avec son contenu.

4.3.2.4 Unités d'œuvre pour le périmètre initial

Pour la reprise des données du périmètre initial des applications Intranet du CNRS, Portail DU, Portail RH et Bourse aux matériels**, le candidat définira dans son offre les unités d'œuvre nécessaires à la reprise de données provenant de technologie SharePoint en fonction des types de contenu source et destination**.

Le format d'export des données sera défini d'un commun accord entre le titulaire et le CNRS, en tenant compte des possibilités techniques de la plateforme SharePoint (CSV, pages HTML ou TXT).

Il peut être envisagé par exemple, une UO par type de contenu comme (liste non exhaustive) :

- Pages (actualité, web, autres…),
- Fichiers / Documents hébergés (PDF, Word, Excel, Zip, images, PowerPoint, …),
- Groupes d'autorisation (administrateur, contributeur, lecteur…),
- Listes,
- Taxonomie / tags,
- Media.

Ces UO peuvent être communes à toutes les applications (Intranet du CNRS, Portail DU, Portail RH et Bourse aux matériels) ou spécifiques à chacune.

Par exemple,

- REP_MIG_PAGES : unité d'œuvre de réalisation de la migration de x données de type « page »
- REP_MIG_TAGS : unité d'œuvre de réalisation de la migration de données de type « tags »
- REP_MIG_INTRA_PDF : unité d'œuvre de réalisation de la migration de 10 données de type fichier PDF
- REP_MIG_LISTE : unité d'œuvre de réalisation de la migration de 10 données de type « liste »

Complexité

Un système de complexité pourra être associé à ces définitions d'UO en fonction par exemple :

- Du volume de données d'un type de contenu à migrer
- Du fait que le type de contenu est associé à des métadonnées (tags, proposition à la newsletter, URL…)
- De la facilité d'automatisation
- Autres (à détailler et justifier).

Par exemple,

- REP_MIG_PAGES : unité d'œuvre de réalisation de la migration de x données de type « page ».

Niveau Complexité	Critères
Faible	Pages de l'Intranet du CNRS ou du Portail DU sans métadonnées
Moyen	Pages de Portail RH ou de Bourse aux matériels
Elevé	Pages de l'Intranet du CNRS ou du Portail DU avec métadonnées

- REP_MIG_TAGS : unité d'œuvre de réalisation de la migration de données de type « tags ».

Niveau Complexité	Critères
Très faible	10 tags à migrer
Faible	50 tags à migrer
Moyen	100 tags à migrer
Elevé	500 tags à migrer
Très élevé	1000 tags à migrer

Coefficient de majoration ou minoration

Cf. §3.3 Coefficient de majoration ou minoratio.

Un système de majoration ou de minoration pourra être associé à ces définitions d'UO en fonction par exemple :

- d'un volume très important
- d'une nouvelle itération de migration de données déjà migrées précédemment (le processus a déjà été éprouvé lors de lots précédents ou d'une migration à blanc par exemple)
- autres (à définir)…

4.4 PRESTATION 4 : MAINTIEN EN CONDITIONS OPÉRATIONNELLES ET DE SÉCURITÉ DES APPLICATIONS APRÈS MISE EN SERVICE

4.4.1 Prestation 4.1 : Prise en main d'une application réalisée hors marché

4.4.1.1 Présentation de la prestation

Il s'agit pour le titulaire d'acquérir la connaissance suffisante d'une application (sous Drupal) et son environnement, pour pouvoir :

- assurer les prestations prévues à l'accord-cadre dans le respect de toutes les dispositions prévues,
- détailler les procédures de travail internes et / ou associées aux relations avec le CNRS, à partir de modalités fixées par l'accord-cadre à la notification.

Cette prestation de prise en main s'effectue en relation avec l'équipe projet DSI.

4.4.1.2 Unité d'œuvre MCO_INTRO – Prise en main d'une application

Présentation

Cette unité d'œuvre peut être utilisée en cours d'exécution de l'accord-cadre, lorsque le CNRS souhaite confier au titulaire le MCO d'une application développée par le CNRS ou par un tiers.

Le lancement d'un projet pour une nouvelle application dans le cadre du marché ne donne pas lieu à une commande de cette UO.

L'introduction en MCO d'une application développée par le titulaire ne donne pas lieu à une commande de cette UO.

Eléments en entrée
Les éléments d'entrée sont :

- Présentation orale de l'application effectuée par son responsable (éventuellement accompagné d'acteurs métiers)
- Ensemble des documents fonctionnels existants (expression des besoins, cahier des charges, documents de spécifications, manuels utilisateurs, aide en ligne…) ;
- Ensemble des documents techniques existants (modèle de données, dossier d'architecture, spécifications techniques, normes de développement (définis dans le CCT), manuels d'installation et exploitation…) ;
- Jeux de données et plans de tests disponibles ;
- Composants en langage source ;
- Accès aux outils existants : Forge, Mantis et espace collaboratif CORE.

Actions à réaliser
Il s'agit pour le titulaire d'acquérir la connaissance suffisante de l'application ainsi que de son environnement, pour pouvoir assurer les diverses unités d'œuvre relatives au « mode MCO » dans le respect de toutes les dispositions prévues.

Sont donc inclus dans ces actions, pour le titulaire les points suivants :

- Préparer et animer la réunion de lancement de la maintenance ;
- Etudier la masse documentaire afin de prendre connaissance de la mise en œuvre de l'application,
  - S'approprier les caractéristiques fonctionnelles de l'application,
  - Analyser les éléments techniques (code source, modèle de données…),
  - Analyser les aspects sécurité (niveau de respect des exigences de sécurité du CCTP) ;

- Mettre en place toutes les conditions techniques et humaines qui sont nécessaires à la bonne réalisation des prestations :
  - mettre en place l'organisation de l'équipe et de l'espace de travail,
  - rendre opérationnel l'environnement de travail et de développement (postes de travail, serveur de développement, outillage de réalisation et de suivi de projet,…),
  - Déposer sur la forge du CNRS les différents éléments (composants en langage source…) ;
- Ajuster les modalités opérationnelles de fonctionnement entre le CNRS et le titulaire ;
- Identifier et analyser les risques, définir les actions de réduction des risques associées ;
- Réaliser des correctifs et/ou évolutifs mineurs sous le contrôle du CNRS et/ou d'un tiers désigné par le CNRS ;
- Livrer pour recette par le CNRS la nouvelle version de l'application correspondant aux correctifs et/ou évolutifs réalisés sous forme d'exécutable.

Livrables
Les livrables pour cette unité d'œuvre sont :

- Compte rendu de la réunion de lancement ;
- Document identifiant les intervenants du titulaire en charge de l'application (notamment pour mise en place des droits idoines dans les outils d'accompagnement) ;
- Plan qualité, plan de mesure du marché et document cible technologique, amendés des spécificités liées à l'entrée de l'application dans le marché ;
- Equipe du titulaire opérationnelle ;
- Plateforme de développement et de test du titulaire opérationnels ;
- Déposer sur la forge du CNRS les différents éléments (composants en langage source…) ;
- Réalisation d'au minimum 3 corrections et/ou 2 petites évolutions sous contrôle du responsable de l'application ;
- Livraison pour recette par le CNRS de la nouvelle version de l'application correspondant à ces corrections et petites évolutions ;
- Rapport d'étonnement présentant :
  - les manques pouvant pénaliser le MCO/le projet : fonctionnalités attendues, documents, données de test, non-respect des normes et de bonnes pratiques de développement,
  - identification des failles de sécurité,
  - identification des défauts de performance (temps de réponses),
  - les grands axes d'améliorations possibles pour l'application/pour le projet.

Complexité
La complexité de cette unité d'œuvre est celle de l'application (cf. § 3.2 Complexité d'une unité d'œuvre).

Coefficient de majoration ou de minoration
Cf. §3.3 Coefficient de majoration ou minoratio.

Modalité de réception par le CNRS

La recette porte, outre sur le constat de remise des livrables, sur le respect des normes techniques de programmation sur les correctifs/évolutifs pris en compte, de documentation et d'exploitation, ainsi que sur la conformité au dossier de spécifications détaillées.

La réception de la prestation est prononcée après la vérification que tous les éléments sont réalisés.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[6] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.4.2 Prestation 4.2 : Maintien en conditions opérationnelles et de sécurité d'une application après mise en service

Il s'agit pour le titulaire de mettre en œuvre les activités nécessaires au maintien en conditions opérationnelles (MCO) et de sécurité (MCS) de l'application une fois sa mise en service opérationnelle réalisée.

[6] La gravité/typologie d'une non-conformité/anomalie est définie dans le glossaire au § 5.1.2

Cela concerne de la maintenance corrective et du petit évolutif.

4.4.2.1 Unité d'œuvre MCO_COR - Correctif

Présentation

Il s'agit pour le titulaire de mettre en œuvre les activités nécessaires au maintien en conditions opérationnelles (MCO) et de sécurité (MCS) de l'application une fois sa mise en service opérationnelle réalisée.

Sur une base forfaitaire mensuelle, les activités à réaliser sont :

- de prendre en compte et de corriger l'ensemble des anomalies applicatives et de cybersécurité (hors période de VSR et de garantie) signalées par le CNRS par l'intermédiaire des déclarations d'anomalies (dans l'outil CNRS Mantis),
- d'effectuer la veille cybersécurité nécessaire, de signaler les vulnérabilités, CVE et alertes, de les analyser dans un délai cohérent avec leurs niveaux de sévérité (CVSS, cf. ci-dessous) et les corriger ou à défaut, proposer une solution palliative afin d'inactiver la faille (application d'un correctif de sécurité core, module contribué, librairie), .
- d'effectuer la veille nécessaire et de mettre à jour le socle applicatif (patchs), core ou module contribué, par exemple Drupal 10.1.5 vers 10.1.6. En cas d'impact avéré, le titulaire informe le CNRS pour décision conjointe de la suite à donner.
- d'assurer un support de niveau N3 auprès de la MOE-DSI,
- d'exercer son rôle de conseil sur les risques d'obsolescences et les vulnérabilités de sécurité des différents composants de chaque application au regard des versions cibles souhaitées conformément aux exigences définies dans le CCT. Ces risques et propositions seront présentés dans un bilan semestriel (cf. §5.2.1.5 Rapport d'analyse de l'obsolescence technique et de vulnérabilités de sécurité),
- de présenter, une fois par an :
  - la complétude et la qualité des connaissances capitalisées,
  - la matrice de couverture des compétences de l'équipe,
  - le plan de réversibilité, PAS et PAQ actualisés.

Les correctifs de cybersécurité liés à des CVE ou à des bulletins d'éditeurs ;

- de CVSS supérieur ou égal à 9 (ou vulnérabilités critiques) doivent être signalées au CNRS et corrigées dans les 24 heures qui suivent leur publication. Le CNRS fermera l'accès à l'application si la correction n'est pas triviale ou en HNO. La marche à suivre sera définie en mode « gestion de crise ».
- de CVSS égal ou supérieur à 7 et strictement inférieur à 9 (ou vulnérabilités hautes) doivent être signalées au CNRS et corrigées dans les 7 jours qui suivent leur publication.
- de CVSS strictement inférieur à 7 (ou vulnérabilités moyennes ou faibles) doivent être signalées au CNRS et corrigées dans les 30 jours qui suivent leur publication.

Les réalisations de ces mises à jour de sécurité sont planifiées conjointement en COSUI (hors vulnérabilités critiques), avec des commandes d'adaptatifs si la charge le nécessite.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de l'anomalie dans l'outil de gestion des anomalies et des évolutions du CNRS. Cette fiche indique si possible un mode opératoire reproductible précis de l'anomalie ainsi que tout élément permettant d'avancer dans son diagnostic (copie d'écran, fichier de trace, …). Elle comporte également une appréciation de la gravité de l'anomalie.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Qualifier (reproduction / diagnostic / solution de contournement) les anomalies transmises par le CNRS et tracées dans l'outil de gestion des anomalies et des évolutions du CNRS ;
- Prendre en charge, réaliser, tester et livrer la correction des anomalies, des vulnérabilités, des montées de version mineures ;

- Garantir de manière tracée la qualité de la maintenance et la non régression globale du système (exécution de tests manuels ou automatiques) ;
- Mettre à jour les fiches descriptives des anomalies dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution proposée pour corriger l'anomalie (description de la modification apportée) et le statut ;
- Effectuer la mise à jour des documentations fonctionnelle et technique impactées ;
- Mettre à jour si nécessaire les manuels d'installation et d'exploitation de la version modifiée ;
- Mettre à jour (via une merge-request dans la forge du CNRS) la configuration de déploiement Ansible si elle existe, et indiquer toutes les modifications à apporter pour prendre en compte le correctif ;
- Etablir si nécessaire une note technique pour les procédures d'installation, d'intégration et d'exploitation de la version modifiée. Ceci concerne particulièrement les anomalies bloquantes et majeures pour lesquelles une livraison partielle (patch) est fortement probable ;
- Livrer le correctif (application et documentation) dans l'environnement de livraison du CNRS.

Livrables
Les livrables pour cette unité d'œuvre sont :

- Application : ensemble des composants sources et exécutables et/ou éléments de paramétrage corrigés ;
- Ensemble de la documentation impactée (technique, test, exploitation) mis à niveau (cf. § 5.2.2 Livrables documentaires relatifs à la maintenance de la solution applicative et § 5.2.3 Livrables documentaires relatifs à l'exploitation de la solution applicative) ;
- Fiches descriptives d'anomalies traitées mises à jour dans l'outil de gestion des anomalies et des évolutions du CNRS ;
- Manuel d'installation de la version corrigée ;
- Merge-request dans la forge du CNRS et explication des modifications à apporter pour prendre en compte le correctif ;
- Semestriellement : rapport d'analyse de l'obsolescence technique et de vulnérabilités de sécurité (cf description des livrables documentaires § 5.2.1)
- Annuellement : complétude qualité des connaissances capitalisées, matrice de couverture des compétences de l'équipe, plan de réversibilité actualisé, PAS et PAQ actualisés.

Complexité
La complexité de cette unité d'œuvre est celle de l'application (cf. § 3.2.2 Complexité générale d'une application).

Modalité de réception par le CNRS

L'acceptation de l'unité d'œuvre est prononcée après la vérification que tous les éléments sont réalisés. Le CNRS peut, a minima :

- vérifier le bon déroulement de l'installation des corrections livrées ;
- exécuter des tests de recette sur le périmètre applicatif de la livraison (dans l'environnement de recette) ;
- vérifier la complétude des tests et jeux de tests du titulaire, le passage avec succès de ces tests dans la forge du CNRS, et dans l'environnement du titulaire le cas échéant ;
- vérifier la qualité du code livré en s'assurant que l'analyse Sonar du nouveau code ne remonte pas de problèmes bloquants, majeurs ou de sécurité ;
- vérifier l'exhaustivité et la pertinence du contenu de la documentation livrée.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[7] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.4.2.2 Révision du forfait de maintenance corrective MCO/MCS d'une application

Le forfait de la prestation de maintenance corrective d'une application pourra être ajusté dans les cas suivants :

Après réalisation d'évolutions au titre des prestations de la 'Prestation 5.2 : Réalisation d'évolutions**' et après la période de garantie des évolutions réalisées,**

[7] La gravité/typologie d'une non-conformité/anomalie est définie dans le glossaire au § 5.1.2

Le forfait de la prestation est ajusté sur la base de la nouvelle complexité de l'application (cf. 3.2.2 Complexité générale d'une application**), afin de prendre en compte la variation de périmètre applicatif ; cette variation pouvant aussi bien être ajustée à la hausse qu'à la baisse.**

4.4.2.3 Unité d'œuvre MCO_EVOLUTION – Petites évolutions

L'unité d'oeuvre correspond à la réalisation par le titulaire d'évolutions qualifiées de « petites », c'est-à-dire les évolutions dont le périmètre s'inscrit dans le fonctionnement courant de l'application.

Une petite évolution ne peut représenter une charge supérieure à trois hommes*jours.

Sur une base forfaitaire trimestrielle, cette unité d'œuvre correspond à un volant de petites évolutions pour une **enveloppe trimestrielle de quinze hommes*jours**.

L'enveloppe de l'unité d'œuvre est décrémentée par décision du CNRS avec la charge associée à chaque demande de petit évolutif effectuée durant le trimestre, en fonction de sa complexité.

Les éléments en entrée, actions à réaliser et livrables attendus pour la réalisation de ces petites évolutions sont les mêmes que ceux décrits dans la Prestation 5 : Développement d'évolutions d'une application existante. Cependant, les livrables attendus sont limités à ceux impactés par la mise en œuvre de l'évolution.

Complexité des évolutions

La complexité de chaque demande de petites évolutions se définit selon les critères suivants :

Granularité	Complexité
Modification simple d'un élément (par exemple : mise en place d'un config ignore sur les rôles et droits).	Très faible
Modification, ajout ou paramétrage d'une fonctionnalité **simple** basée sur des éléments standards simples ou sur de l'existant simple (par exemple : installation du module « Security kit » et première configuration des Content Security Policies (CSP) ou ajout fonctionnalité de « slideshow » via librairie JavaScript).	Faible
Modification, ajout ou paramétrage d'une fonctionnalité **simple** basée sur des éléments proches d'éléments existants complexes (par exemple : compression sans perte des images sur un multisite préexistant (modules Drupal, ajustement du code, librairies supplémentaires)).	Moyen
Modification, ajout ou paramétrage d'une fonctionnalité **complexe** nécessitant de nouveaux développements simples (par exemple : mise en place du format WebP sur un multisite préexistant (modules Drupal, nombreux ajustements du code et templates)).	Elevé

4.4.2.4 Unité d'œuvre MCO_EVOLUTION_SUPPLEMENTAIRE – Petites évolutions supplémentaires

L'unité d'oeuvre correspond à la réalisation par le titulaire d'évolutions qualifiées de « petites », c'est-à-dire les évolutions dont le périmètre s'inscrit dans le fonctionnement courant des applications.

L'unité d'œuvre correspond à la prise en charge de petites évolutions pour une **enveloppe de dix hommes*jours**.

Une petite évolution ne peut représenter une charge supérieure à trois hommes*jours.

L'unité d'œuvre correspond à une enveloppe supplémentaire afin de traiter les demandes ne pouvant être traitées dans le cadre de la commande récurrente de MCO_EVOLUTION – Petites évolutions.

Contrairement à l'unité d'œuvre MCO_EVOLUTION, l'enveloppe MCO_EVOLUTION_SUPPLEMENTAIRE non épuisée en fin de trimestre ou d'année est reconduite la période suivante. En cas de crédit non épuisé en fin d'accord cadre, le montant non consommé reste acquis au titulaire.

Le nombre maximal d'UOs MCO_EVOLUTION_SUPPLEMENTAIRE pouvant être commandées est limité à 10 (dix) par an.

4.4.3 Prestation 4.3 : Sortie du périmètre du MCO d'une application

4.4.3.1 Présentation de la prestation

Il s'agit pour le CNRS de sortir une application existante du périmètre du marché sans transfert de compétence, notamment dans le cas d'une application n'ayant plus d'utilité dans le SI.

4.4.3.2 Unité d'œuvre MCO_SORTIE – Sortie d'une application du périmètre du marché sans réversibilité

Présentation
Cette unité d'œuvre permet de sortir une application existante du périmètre du marché. Dans cette UO il s'agit d'une sortie « sèche » ne bénéficiant d'aucun encadrement. Cette UO correspond notamment au cas d'une application n'ayant plus d'utilité dans le SI.
Une fois hors du périmètre, cette application ne pourra pas faire l'objet d'une UO avant d'avoir été concernée par une UO MCO_ INTRO.

Eléments en entrée

- Identification de l'application concernée

Actions à réaliser
Les actions à réaliser par le titulaire sont les suivantes :

- Transférer les informations fonctionnelles et techniques sur l'application au CNRS ;
- Décommissionner les environnements correspondant à l'application chez le titulaire ;
- Supprimer tous les documents et toutes les données liés à l'application chez le titulaire.

Livrables

- Intégralité du code de l'application non encore soumis ;
- Ensemble des éléments documentaires de l'application ;
- Un procès-verbal de sortie du périmètre de l'application.

Complexité
Cette unité d'œuvre n'accepte qu'une seule complexité car les actions à réaliser sont globalement indépendantes de l'application en elle-même. Symboliquement, cette complexité est indiquée comme étant faible.

Modalité de réception par le CNRS
La réception de la prestation est prononcée après la vérification que toutes les actions sont réalisées.

4.5 PRESTATION 5 : DÉVELOPPEMENT D'ÉVOLUTIONS D'UNE APPLICATION EXISTANTE

Cette prestation consiste à concevoir et réaliser une ou plusieurs évolutions répondant aux exigences fonctionnelles et techniques exprimées dans la demande du CNRS. Le titulaire doit proposer au CNRS une solution s'appuyant au maximum sur du paramétrage.

Les évolutions couvrent des aspects :

- fonctionnels : ajout ou modification de fonctionnalités « métier »,
- réglementaires (directives nationales, ministérielles, CNRS…),
- de conformité aux réglementations qui évoluent (RGAA, RGPD…),
- techniques : évolution d'architecture technique, nouvelles versions des logiciels et outils associés, des systèmes d'exploitation.

4.5.1 Prestation 5.1 : Conception d'évolutions d'une application

4.5.1.1 Présentation de la prestation

Cette prestation consiste à concevoir une ou plusieurs évolutions répondant aux exigences fonctionnelles et techniques exprimées dans la demande du CNRS.

Cette UO n'est déclenchée, à priori, que dans le cas d'évolutions importantes, par exemple pour un volume de modifications important ou d'évolutions complexes ou pour lesquelles différents interlocuteurs doivent être sollicités (MOA, MOE). Pour les évolutions moins importantes, seule les UO de réalisation d'évolutions seront commandées.

La décision de commander cette unité d'œuvre relève du seul choix du CNRS.

4.5.1.2 Unité d'œuvre CONCEPTION – Conception d'évolutions

Présentation

L'objet de cette unité d'œuvre est de mettre en œuvre des ateliers de conception et de réaliser des maquettes (si demandées) en vue de produire les spécifications fonctionnelles détaillées et spécifications techniques détaillées d'une évolution telle qu'exprimée par le CNRS (cf. la description du contenu des documents au § 5.2).

L'affinage des règles de gestion et de l'expression du besoin se fait dans le cadre d'un *dialogue utilisateur* :

- avec la MOA, les utilisateurs métiers et la MOE pour les aspects fonctionnels,
- avec la MOE pour l'intégration dans le SI,
- avec la MOE pour l'architecture et la sécurité.

Le titulaire assurera son devoir de conseil en étant force de propositions et garantira dans ses propositions une parfaite homogénéité de l'application ainsi que la mise en pratique des exigences ergonomiques définies.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin de spécifications fonctionnelles générales de la demande ;
- Règles d'ergonomie et charte graphique CNRS ;
- Spécifications techniques détaillées, si le document existe ;
- Description des contraintes particulières d'exploitation éventuelles.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Animer, si nécessaire, des ateliers fonctionnels ou techniques pour affiner la solution et rédiger les comptes rendus de ces ateliers :
  - Ateliers de conception fonctionnelle, UX/UI avec la MOA, les utilisateurs métiers et la MOE ;
  - Ateliers de conception d'intégration dans le SI, avec la DSI, pour définir la solution d'échange avec le système d'intermédiation pour chacun des flux ;
  - Si besoin, ateliers architecture et sécurité ;
- Concevoir la solution fonctionnelle :
  - Co-construire avec le CNRS la solution fonctionnelle (fonctionnalités, ergonomie, graphisme, accessibilité…)
  - Définir les options permettant de garantir la modularité et l'évolutivité des fonctionnalités ;
- Concevoir la solution Drupal :
  - Définir et ajuster l'architecture applicative et technique des données et des traitements (découpage et articulation des composants logiciels...) ;

    - Définir les options permettant de garantir une modularité optimale des traitements, le niveau de paramétrage attendu, le niveau et le format des logs et les performances optimales ;
- Si besoin, réévaluer le dimensionnement (puissance et stockage) des plateformes de recette et de production ;
- Réaliser wireframes, maquettes ou prototypes prenant en compte les règles d'accessibilité, d'ergonomie et la charte graphique CNRS et permettant l'évaluation/validation du besoin par les utilisateurs ;
- Rédiger ou mettre à jour la documentation technico-fonctionnelle et en particulier :
    - les spécifications fonctionnelles détaillées(cf. la description du contenu au § 5.2.2.1),
    - les spécifications techniques détaillées (cf. la description du contenu au § 5.2.2.2),
    - le dossier d'architecture logicielle (cf. la description du contenu au § 5.2.3.1),
    - la conception technique détaillée de la solution (cf. la description du contenu au § 5.2.2.3) ;
- Vérifier par contrôle qualité interne au titulaire le respect des normes de documentation.

Livrables
Les livrables pour cette unité d'œuvre sont :

- Animation et compte rendu des ateliers menés ;
- Wireframe, maquettes ou prototypes le cas échéant ;
- Spécifications fonctionnelles détaillées crées ou mises à jour (cf. la description du contenu au § 5.2.2.1),
- Spécifications techniques détaillées crées ou mises à jour (cf. la description du contenu au § 5.2.2.2),
- Dossier d'architecture logicielle créé ou mis à jour (cf. la description du contenu au § 5.2.3.1),
- Conception technique détaillée de la solution créé ou mis à jour (cf. la description du contenu au § 5.2.2.3) ;
- Fiches descriptives de la demande traitées mises à jour dans l'outil de gestion des anomalies et des évolutions du CNRS.
- Proposition de planning et de charge associée pour mettre en œuvre l'objet de la demande, exprimée en unités d'œuvre.

Les éléments suivants sont transmis au CNRS au fur et à mesure de leur production mais ne sont pas considérés comme des livrables nécessitant une validation du CNRS :

- Plans, dossiers, jeux d'essais et résultats de tests unitaires, d'intégration, de validation, de non-régression, de performance et de sécurité initiée en fin de phase de conception puis complétés en phase de tests.

Complexité

Le calcul de complexité ne doit prendre en compte – en tant que nombre d'informations mises en jeu – que les éléments faisant l'objet de l'évolution. Par exemple, en l'absence de demande d'une maquette IHM, l'item est valorisé à 0.

Une information est définie au sens métier du terme (du niveau expression de besoin métier).

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Ateliers technico-fonctionnel (comprend la préparation en amont, la tenue de l'atelier et la rédaction du CR)	Simple : atelier inférieur ou égal à 2h (réponse points ouverts, clarification, validation de maquettes…)	Atelier	2
	Normal : atelier dont la durée est > 2h et <= 4h inclus (réponse points ouverts, clarification, validation de maquettes…)	Atelier	4
Cas d'utilisation	Très simple : nombre de cas d'utilisation mis en jeu inférieur ou égal à 5	Cas	2
	Simple : nombre de cas d'utilisation mis en jeu entre 6 et 10 inclus	Cas	5
	Normal : nombre de cas d'utilisation mis en jeu entre 11 et 20 inclus	Cas	8

Elément particulier	Niveau de complexité	Unité	Note de complexité
	Complexe : nombre de cas d'utilisation mis en jeu entre 21 et 30 inclus	Cas	10
Réalisation d'une maquette **Maquette simple** : nombre d'informations mises en jeu <10 **Maquette normale :** 10<=nombre d'informations mises en jeu <25 **Maquette Complexe :** nombre d'informations mises en jeu >=25	Simple : Création maquette simple ou réutilisation d'une maquette normale (exemple, maquette normale pour la création, réutilisation pour la modification ou la consultation)	Cas	2
	Normal : Création d'une maquette normale ou réutilisation d'une maquette complexe (exemple, maquette complexe pour la création, réutilisation pour la modification ou la consultation)	Cas	4
	Complexe : Création d'une maquette complexe	Cas	6
Modification d'une maquette	Simple : - Modification d'une maquette existante dont le nombre d'informations modifiés <=6 éléments nouveaux - Ou modification d'une maquette avec seulement des modifications de libellés quel que soit le nombre d'éléments	Cas	1
	Normal : modification d'une maquette existante dont le nombre d'informations mises en jeu > 6 éléments nouveaux	Cas	2

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau Complexité	Critères
Très faible	Note de complexité <= 10
Faible	10 < Note de complexité <= 15
Moyen	15 < Note de complexité <= 20
Elevé	20 < Note de complexité <= 30
Très élevé	30 < Note de complexité <= 40

Coefficient de majoration ou de minoration
Cf. §3.3 Coefficient de majoration ou minoratio.

Modalité de réception par le CNRS

La recette porte, outre sur le constat de remise des livrables, sur le respect des normes de documentation.

La réception de la prestation est prononcée après la vérification que tous les éléments sont réalisés.

Lors de la recette (Vérification d'Aptitude), le CNRS peut, a minima :

- vérifier la complétude des éléments livrés (dont les maquettes si demandé) ;
- vérifier le bon déroulement des ateliers (si demandé) ;
- vérifier l'exhaustivité et la pertinence du contenu de la documentation livrée.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[8] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.5.2 Prestation 5.2 : Réalisation d'évolutions

4.5.2.1 Présentation de la prestation

Cette prestation consiste à concevoir, réaliser et tester une évolution répondant aux exigences fonctionnelles et techniques exprimées dans la demande du CNRS.

[8] La typologie des anomalies est décrite dans le glossaire au § 5.1.2

Au cours de la prestation, et de façon incrémentale, le titulaire présente ses réalisations au CNRS pour validation afin de s'assurer de l'avancement de la prestation par le titulaire.

Le titulaire, outre les livrables liés à la réalisation, présente son plan de tests et le résultat de l'exécution du plan de test sur les fonctionnalités impactées par la demande.

Le titulaire doit proposer au CNRS une solution s'appuyant au maximum sur du paramétrage.

Le titulaire répond à la demande en mobilisant les ressources et profils nécessaires à la réalisation de la prestation.

4.5.2.2 Eléments communs aux unités d'œuvre de réalisation

Présentation
La réalisation d'évolutions s'entend au sens fonctionnel (ajout ou modification d'une fonctionnalité) ou technique (changement de présentation ou d'ergonomie de l'application).
La réalisation des changements de version majeure ou mineure des logiciels sont compris dans les UO de migrations techniques.

Le titulaire assurera son devoir de conseil en étant force de propositions et garantira dans ses propositions une parfaite homogénéité de l'application ainsi que la mise en pratique des exigences ergonomiques définies.

Eléments en entrée
Les éléments d'entrée sont :

- Fiche descriptive de l'évolution dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin de spécifications fonctionnelles générales (éventuellement complétées de documents de maquettage : schéma, copie d'écran annotée)
- Spécifications fonctionnelles et techniques détaillées, si elles existent ;
- Règles d'ergonomie et charte graphique CNRS ;
- Normes et standard de programmation et d'exploitation (définis dans le CCT) ;
- Description des contraintes particulières d'exploitation éventuelles ;
- Données de tests si besoin.

Actions à réaliser
Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande
- Identifier les points bloquants
- Concevoir la solution à mettre en place
- Spécifier la solution
- Si nécessaire, animer des ateliers fonctionnels ou techniques pour valider la solution et rédiger les comptes rendus de ces ateliers ;
- Si nécessaire, réaliser wireframes, maquettes ou prototypes prenant en compte les règles d'accessibilité, d'ergonomie et la charte graphique CNRS et permettant l'évaluation/validation du besoin par les utilisateurs ;
- Mettre à jour la documentation technico-fonctionnelle et en particulier :
  - les spécifications fonctionnelles détaillées(cf. la description du contenu au § 5.2.2.1),
  - les spécifications techniques détaillées (cf. la description du contenu au § 5.2.2.2),
  - le dossier d'architecture logicielle (cf. la description du contenu au § 5.2.3.1),
  - la conception technique détaillée de la solution (cf. la description du contenu au § 5.2.2.3) ;
- Mettre à niveau l'environnement de développement afin d'être en conformité avec l'architecture cible définie
- Réaliser le paramétrage et/ou les développements spécifiques constituant l'évolution ;
- Pousser régulièrement le code source dans la forge du CNRS sans attendre la fin de la réalisation ;
- Analyser et prendre en compte les résultats de l'intégration continue et de l'analyse Sonar de la forge du CNRS ;
- Rédiger le plan de test

- Réaliser les tests suivants[9] sur différentes plateformes (OS, navigateurs) depuis l'environnement de développement et l'environnement d'intégration :
  - tests unitaires automatisés et manuels,
  - tests d'intégration,
  - tests de validation,
  - tests de non régression de l'application,
  - tests de performance unitaires et/ou plus globaux en fonction des travaux réalisés,
  - tests de sécurité (pour les versions majeures) ;
- Vérifier par contrôle qualité interne au titulaire, le respect des normes de développement et de documentation ;
- Spécifier les procédures permettant l'installation, la mise en exploitation et l'exploitation de l'application en prenant en compte le déploiement via Ansible ;
- Mettre à jour les manuels d'installation et d'exploitation (cf. la description du contenu aux § 5.2.3.2 et § 5.2.3.3) ;
- Livrer la version réalisée dans l'environnement de livraison du CNRS, en utilisant Ansible ;
- Assister l'équipe technique du CNRS pour assurer l'installation dans l'environnement de recette ;
- Prendre en compte les éventuelles anomalies[10] de recette détectée par le CNRS et procéder aux relivraisons nécessaires (phase de Vérification d'Aptitude) ;
- Assister si besoin les équipes CNRS, lors de l'installation et de la mise en exploitation dans l'environnement de production ;
- Prendre en compte les éventuelles anomalies de production détectées par le CNRS et procéder aux relivraisons nécessaires (phases de Vérification de Service Régulier et de garantie) ;
- Mettre à jour la fiche descriptive du développement réalisé dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution mise en œuvre et le statut ;

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Fiche descriptive du développement réalisé mise à jour dans l'outil de gestion des anomalies et des évolutions du CNRS ;
- Spécifications fonctionnelles détaillées mises à jour (cf. la description du contenu au § 5.2.2.1) ;
- Wireframes, maquettes, prototypes
- Spécifications techniques détaillées mises à jour (cf. la description du contenu au § 5.2.2.2) ;
- Conception technique détaillée de la solution créé ou mis à jour (cf. la description du contenu au § 5.2.2.3) ;
- Dossier d'architecture logicielle mis à jour (cf. la description du contenu au § 5.2.3.1) ;
- Rapport de tests unitaires (noms, date, numéro de version) ;
- Manuel d'installation comprenant la configuration permettant la mise au point des playbooks par le CNRS (cf. la description du contenu au § 5.2.3.3) ;
- Manuel d'exploitation mis à jour (cf. la description du contenu au § 5.2.3.2) ;
- Code source poussé régulièrement dans la forge du CNRS sans attendre la fin de la réalisation, afin de valider la chaine d'intégration continue et de suivre l'évolution de la qualité du code ;
  - Exécutable livré selon les modalités spécifiques à la plateforme (cf. § 5.5 du CCTP – Livret 1) ;
  - Ensemble de tous les fichiers, y compris source, correspondant à la branche ou à l'étiquette livrée et permettant la génération de cet exécutable ;
- Paramétrage mis en œuvre ;
- Scripts d'installation éventuels.

Les éléments suivants sont transmis au CNRS au fur et à mesure de leur production mais ne sont pas considérés comme des livrables nécessitant une validation du CNRS :

[9] Cf. la définition des différents types de tests dans le glossaire au § 5.1.2
[10] La définition d'une anomalie est décrite dans le glossaire au § 5.1.2

- Plans, dossiers, jeux d'essais et résultats de tests unitaires, d'intégration, de validation, de non-régression et de performance, et de sécurité initiés en fin de phase de conception puis complétés en phase de tests.

Modalité de réception par le CNRS

La recette porte, outre sur le constat de remise des livrables, sur le respect des normes techniques de programmation, de documentation et d'exploitation, ainsi que sur la conformité au dossier de spécifications détaillées.

La réception de la prestation est prononcée après la vérification que tous les éléments sont réalisés.

Lors de la recette (Vérification d'Aptitude), le CNRS peut, a minima :

- vérifier la complétude des éléments livrés ;
- vérifier le bon déroulement de l'installation de l'application livrée, sur l'environnement de recette ;
- exécuter des tests de recette sur le périmètre applicatif de la livraison (dans l'environnement de recette) ;
- vérifier la qualité du code (application des normes, détection de problèmes potentiels de qualité ou de sécurité du code par Sonar…), la complétude des tests et jeux de tests du titulaire, le passage avec succès de ces tests dans l'environnement du titulaire ;
- vérifier la « robustesse » de la solution et la montée en charge des utilisateurs ;
- vérifier l'exhaustivité et la pertinence du contenu de la documentation livrée.

Lors de la Vérification de Service Régulier (VSR), le CNRS vérifie l'utilisation de l'application dans des conditions normales en environnement de production.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[11] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.5.2.3 Unité d'œuvre REA_FRONT – Réalisation d'évolutions de type Front End

Présentation

Cette unité d'œuvre consiste à réaliser une évolution ou une adaptation de l'application front-end portant par exemple sur les thèmes Drupal, templates Twig, CSS, JavaScript (côté client), ergonomie, accessibilité (RGAA) et blocs/regions. Toute logique métier, configuration Drupal, ou intégration back-end relève de REA_BACK.

Eléments en entrée
Les éléments d'entrée sont :

- Les éléments décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation - Eléments en entrée

Actions à réaliser
Les actions à réaliser par le titulaire sont les suivantes :

- Les éléments décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation – Actions à réaliser;

Livrables

Les livrables pour cette unité d'œuvre sont :

- Les livrables décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation – Livrables ;

Complexité
Le calcul de complexité ne doit prendre en compte – en tant que nombre d'informations mises en jeu – que les éléments faisant l'objet de l'évolution. Une information est définie au sens métier du terme (du niveau expression de besoin métier). Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Gabarits (Templates Twig)			

[11] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Gabarit simple	Personnalisation minime d'un template existant (par exemple : ajout d'une classe CSS, modification de balises HTML).	Modifier node.html.twig pour ajouter une classe en fonction du type de contenu.	1
Gabarit moyen	Création ou modification d'un template pour un type de contenu/entité (par exemple : node--mon-type.html.twig).	Créer un template personnalisé pour afficher un type de contenu avec un layout spécifique.	2
Gabarit complexe	Personnalisation avancée (préprocesseurs, inclusion de composants, logique conditionnelle complexe).	Développer un template avec des conditions basées sur des champs, des boucles, et des appels à des services.	3
Thèmes Front-Office			
Thème simple	Modification mineure d'un thème existant (CSS, quelques templates, ou paramètres de thème).	Modifier les couleurs et polices d'un thème existant via son fichier .theme.	2
Thème moyen	Création d'un sous-thème ou adaptation d'un thème existant avec plusieurs templates et assets CSS/JS.	Créer un sous-thème de Olivero avec des templates personnalisés et un CSS spécifique.	4
Thème complexe	Création d'un thème custom complet (tous les templates, préprocesseurs SASS/LESS, structure responsive).	Développer un thème sur mesure avec des templates pour chaque type de contenu, un design responsive, et des animations.	6
Adaptation Back-Office			
CSS simple	Modifications mineures (couleurs, polices, marges, padding).	Ajouter une classe pour styliser un bouton ou un bloc.	1
CSS moyen	Refonte partielle du design (plusieurs pages, responsive basique, utilisation de variables CSS).	Restyler une page complète avec un design responsive pour mobile/tablette.	2
CSS complexe	Refonte complète du design (toutes les pages, animations CSS, framework custom, ou utilisation de SASS/LESS).	Créer un design custom avec des animations, des transitions, et un système de grille avancé.	4
JavaScript (côté client)			
JS simple	Scripts simples (validation de formulaire client, animations basiques, gestion d'événements).	Ajouter une validation côté client sur un champ de formulaire.	1
JS moyen	Interactions dynamiques (AJAX, manipulation du DOM, libs légères comme jQuery UI).	Développer un filtre dynamique sur une liste (ex: filtre par catégorie sans rechargement).	3
JS complexe	Intégration de composants React/Vue dans Drupal (par exemple : affichage dynamique de données via une API REST)	Intégrer une librairie de graphiques (ex: Chart.js) pour afficher des données dynamiques.	5
Ergonomie/Accessibilité (RGAA)			
Ergonomie simple	Corrections mineures (amélioration de l'UX basique, corrections RGAA niveau A).	Ajouter des labels manquant pour les champs de formulaire (RGAA 1.1).	1
Ergonomie moyenne	Audit RGAA partiel ou refonte de l'ergonomie d'une page (niveau AA).	Corriger les contrastes de couleurs et la navigation au clavier pour une page.	2

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Ergonomie complexe	Audit RGAA complet (niveau AA ou AAA) avec corrections ou refonte complète de l'ergonomie de l'application.	Auditer et corriger l'ensemble du site pour une conformité RGAA niveau AA.	4
Blocs et régions			
Bloc simple	Ajout ou configuration d'un bloc simple (par exemple bloc de menu, bloc HTML statique).	Ajouter un bloc HTML dans le footer.	1
Bloc moyen	Développement d'un bloc custom simple (par exemple : affichage dynamique de données).	Créer un bloc affichant les 5 derniers nœuds d'un type de contenu.	2
Bloc complexe	Développement d'un bloc custom complexe (par exemple : logique métier, interactions utilisateur).	Développer un bloc avec un formulaire de recherche avancée et affichage dynamique des résultats.	3

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau Complexité	Critères
Très faible	Note de complexité <= 5
Faible	6 <= Note de complexité <= 10
Moyen	11 <= Note de complexité <= 15
Elevé	16 <= Note de complexité <= 20
Très élevé	21 <= Note de complexité <= 25

Si la note de complexité dépasse 25 une unité d'œuvre supplémentaire sera commandée.

Coefficient de majoration ou de minoration
Cf. §3.3 Coefficient de majoration ou minoratio.

4.5.2.4 Unité d'œuvre REA_BACK – Réalisation d'évolutions de type Back End

Présentation

Cette unité d'œuvre consiste à réaliser une évolution ou une adaptation back-end portant par exemple sur les types de contenu, entités, champs, vues (Views), modules (contrib/custom), workflows, permissions, API REST/GraphQL, intégrations de services externes.

Toute mise en forme, design, ou interaction front-end relève de REA_FRONT.

Eléments en entrée

Les éléments d'entrée sont :

- Les éléments décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation - Eléments en entrée

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Les éléments décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation – Actions à réaliser;

Livrables

Les livrables pour cette unité d'œuvre sont :

- Les livrables décrits au paragraphe : 4.5.2.2 Eléments communs aux unités d'œuvre de réalisation – Livrables ;

Complexité

Le calcul de complexité ne doit prendre en compte – en tant que nombre d'informations mises en jeu – que les éléments faisant l'objet du besoin. Une information est définie au sens métier du terme (du niveau expression de besoin métier).

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Création/Modification de type de contenu			
Type de contenu Simple	Ajout, suppression ou modification d'un type de contenu (par exemple : Article, Événement). Inclut la définition des champs, des formulaires d'édition, et des paramètres de publication.	Ajout d'un champ « Date » à un type existant	1
Type de contenu moyen	Création d'un type de contenu avec 5 à 10 champs (dont au moins 1 référence à une taxonomie ou une entité).	Créer un type "Offre d'emploi" avec champs : titre, description, date limite, catégorie (taxonomie).	2
Type de contenu complexe	Création d'un type de contenu avec plus de 10 champs ou avec des validations conditionnelles complexes (par exemple : champ obligatoire si un autre champ a une valeur spécifique).	Créer un type "Projet" avec 12 champs, dont 3 références à d'autres entités et 2 validations croisées.	3
Entité personnalisée			
Entité personnalisée	Développement d'une entité custom (via hook_entity_info() ou Drupal::entityTypeManager()). Inclut la définition des champs, des formulaires, et des méthodes de stockage.	Créer une entité "Laboratoire" avec des champs spécifiques et une table dédiée en base.	3
Champs et référentiels			
Champs et référentiels simples	Ajout ou modification de champs simples (texte, date, booléen) sans logique métier associée.	Ajouter un champ "Couleur" (liste de valeurs) à un type de contenu.	1

Elément particulier	Niveau de complexité	Exemple	Note de complexité
Champs et référentiels moyens	Ajout ou modification de champs avec dépendances (par exemple : référence à une taxonomie, une entité, ou un utilisateur) ou validations basiques (par exemple : date > aujourd'hui).	Ajouter un champ "Responsable" (référence à un utilisateur) avec validation.	2
Champs et référentiels complexes	Ajout ou modification de champs avec logique métier complexe (par exemple : calcul dynamique, validation conditionnelle multi-champs, ou champ calculé).	Créer un champ "Budget restant" calculé à partir d'un champ "Budget total" et d'un champ "Dépenses".	3
Vues (Views)			
Vue simple	Configuration d'une vue simple (1 table, 1-2 filtres basiques, tri standard).	Créer une vue listant tous les nœuds d'un type de contenu.	1
Vue moyenne	Configuration d'une vue moyenne (2-3 tables, filtres avancés, relations simples, ou agrégations basiques).	Créer une vue listant les nœuds avec des champs de référence, filtrés par taxonomie.	2
Vue complexe	Configuration d'une vue complexe (relations multiples, filtres dynamiques, agrégations, ou calculs personnalisés).	Créer une vue affichant le nombre de nœuds par catégorie, avec un filtre par date et un tri personnalisé.	3
Modules (Contrib)			
Module simple	Installation et configuration basique d'un module contrib (par exemple : Webform, Pathauto).	Installer et configurer Pathauto pour des URLs automatiques.	1
Module moyen	Configuration avancée d'un module contrib (par exemple : Views avec relations, Rules avec actions conditionnelles).	Configurer une vue avec des relations vers d'autres entités et des filtres dynamiques.	2
Modules (Custom)			
Module Moyen	Développement d'un module custom simple (par exemple hook basique, formulaire simple).	Créer un module pour ajouter un champ calculé à une entité.	2
Module complexe	Développement d'un module custom avec logique métier (par exemple : gestion de workflows, intégration API basique).	Développer un module pour gérer des inscriptions à des événements avec validation.	3
Module très complexe	Développement d'un module custom complexe (par exemple intégration avec une API externe, gestion de données massives, ou logique métier avancée).	Créer un module pour synchroniser des données avec un CRM externe via API REST avec gestion des erreurs et reprise sur échec.	4
Rôles et permissions			
Rôles et permissions simples	Configuration basique des rôles/permissions (par exemple : donner l'accès à un type de contenu).	Créer un rôle "Éditeur" avec permission de modifier un type de contenu.	1
Rôles et permissions avancés	Configuration avancée des rôles/permissions (ex: restrictions par champ, permissions granulaire).	Restreindre l'accès à certains champs d'un type de contenu pour un rôle spécifique.	2
Workflow			
Workflow simple	Configuration d'un workflow simple (2-3 états, transitions basiques).	Créer un workflow "Brouillon → Publié" pour un type de contenu.	2
Workflow complexe	Configuration d'un workflow complexe (4+ états, transitions conditionnelles, ou intégration avec d'autres modules).	Développer un workflow avec 5 états et des validations dynamiques (par exemple : date de fin > aujourd'hui).	3
API REST/GraphQL			

Elément particulier	Niveau de complexité	Exemple	Note de complexité
API REST moyenne	Exposition basique de données via API REST (par exemple : endpoint pour lister des nœuds).	Créer un endpoint REST pour exposer un type de contenu.	2
API REST complexe	Exposition avancée de données via API REST/GraphQL (par exemple : filtres dynamiques, authentification, ou mutation custom).	Développer une API GraphQL avec des requêtes personnalisées et une authentification JWT.	3
Intégration de services externes			
Intégration moyenne	Intégration basique avec un service externe (par exemple : envoi d'emails via SMTP, connexion LDAP simple).	Configurer l'envoi d'emails via Symfony Mailer.	2
Intégration complexe	Intégration avancée avec un service externe (par exemple : synchronisation bidirectionnelle, gestion d'erreurs, ou API complexe).	Synchroniser des utilisateurs avec un annuaire LDAP et gérer les erreurs.	3
Intégration très complexe	Intégration avec un ERP via un protocole propriétaire (par exemple SAP via RFC/BAP ou Oracle EBS).	Intégration d'un SOAP complexe et mapping de données avancé et gestion des erreurs.	4

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau Complexité	Critères
Très faible	Note de complexité <= 5
Faible	6 <= Note de complexité <= 10
Moyen	11 <= Note de complexité <= 15
Elevé	16 <= Note de complexité <= 20
Très élevé	21 <= Note de complexité <= 25

Si la note de complexité dépasse 25 une unité d'œuvre supplémentaire sera commandée.

Coefficient de majoration ou de minoration
Cf. §3.3 Coefficient de majoration ou minoratio.

4.6 PRESTATION 6 : ÉTUDES ET EXPERTISES

4.6.1 Présentation de la prestation

La réalisation d'études ou d'expertises n'est confiée au titulaire que lorsque ponctuellement le CNRS en exprime le besoin.

Suite à une expression de besoins du CNRS, le titulaire doit intervenir en étroite collaboration avec les équipes CNRS.

Il s'agit pour le titulaire, de répondre aux demandes d'étude fonctionnelle et/ou technique émises par le CNRS.

4.6.2 Unité d'œuvre ETD_EXP – Etude ou expertise fonctionnelle et/ou technique

Présentation
Cette unité d'œuvre permet de demander une étude ou une expertise fonctionnelle et/ou technique.

Ces études peuvent être, par exemple (liste non exhaustive) :

- des études d'impacts (*par exemple, l'impact de la modification demandée*) ;
- des études d'opportunité (*par exemple, avantages et inconvénients d'installer une nouvelle fonctionnalité d'un des composants utilisés*) ;

- des études de faisabilité (*par exemple, nouvelles fonctionnalités lors de l'ajout d'un nouveau module*) ;
- des expertises sur des éléments de sécurité (Tests d'intrusion, Audits de sécurité, Qualité de code sonar, …).

Eléments en entrée
Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins.

Actions à réaliser
Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande du CNRS ;
- Animer des ateliers fonctionnels ou techniques pour affiner le besoin et rédiger les comptes rendus de ces ateliers ;
- Réaliser l'étude et rédiger un rapport faisant clairement apparaître les préconisations du titulaire (étude d'impact sur la solution existante et sur les SI du CNRS, différents scénarii possibles, coûts associés, conclusions, conseils...) et les éléments y ayant conduit ;
- Présenter les résultats de l'étude au CNRS pouvant contenir les points suivants en fonction de la nature de l'étude (non exhaustif) :
  - l'identification des entités Drupal à créer/modifier (par exemple : Content Types, Taxonomy, Custom Entities),
  - la description des relations entre entités (par exemple : Entity Reference, Relationships dans Views),
  - l'identification des écrans à modifier ou la description des pages/formulaires à créer (par exemple : Views, Blocks, Webform, Custom Forms avec Form API),
  - l'identification et la compatibilité des thèmes, des librairies frontend (jQuery, Bootstrap, etc.),
  - l'identification des hooks ou events à utiliser (par exemple : hook_node_presave, Event Subscribers),
  - la description des services ou plugins à développer (par exemple : Custom Modules, Services Symfony),
  - l'identification des traitements batch à implémenter (par exemple : Batch API pour les imports massifs, Queue API pour les tâches asynchrones),
  - la description des règles à appliquer en cas de modification du modèle des données pour assurer la migration des données,
  - l'identification des traitements à modifier ou la description des traitements à créer,
  - l'identification des reports ou exports à créer (par exemple : Views avec export CSV/PDF, Print module pour des PDF).,
  - l'identification des contraintes métiers ou organisationnelles ;
  - …
- Proposer, que ce soit ou non sur la demande du CNRS, des exemples de développement sur lesquels le CNRS pourra s'appuyer pour les reproduire, ou pour effectuer une qualification des solutions proposées, même si le but de l'étude/de l'expertise n'est pas de faire réaliser par le titulaire une maintenance ou une évolution de l'application en fonction des préconisations énoncées ;
- Proposer un séquencement des actions à réaliser ;
- Evaluer un premier chiffrage de la charge de travail de réalisation, exprimée en unités d'œuvre.
- Le titulaire assurera son devoir de conseil en étant force de propositions et garantira dans ses propositions une parfaite homogénéité de l'application ainsi que la mise en pratique des exigences ergonomiques définies

Livrables
Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Rapport d'étude ou d'expertise regroupant les conclusions sur l'ensemble des éléments sur lesquels a porté l'étude et contenant a minima :
  - des préconisations des solutions et améliorations, tant au niveau de la définition des objets dans l'application, que de leur description, de leur utilisation, que des écrans, des éditions réalisés dans l'application ;
  - des propositions, si nécessaire, d'exemples de développement ayant été éprouvés, dans la mesure du possible, sur une plate-forme de développement ou d'intégration.

- Présentation du rapport d'étude ou d'expertise ;
- Proposition de planning et de charge associée pour mettre en œuvre l'objet de la demande, exprimée en unités d'œuvre.

Complexité

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Dans le cas où la note globale de complexité d'une application est supérieure à la valeur maximum du tableau « niveau de complexité/critères » (§ 3.2.2 Complexité générale d'une application), la complexité de l'application utilisée est la complexité la plus haute.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Note de complexité
Ateliers La durée d'un atelier varie entre 2h et 4h. Un atelier peut être scindé sur 1 ou 2 jours	1 ou 2 ateliers pour recueillir l'ensemble des éléments.	1
	3 à 5 ateliers sont nécessaires afin de préciser un besoin fonctionnel ou une contrainte technique moyennement complexe.	2
	6 ateliers ou plus sont nécessaires afin de préciser un besoin fonctionnel complexe ou une contrainte technique importante.	3
Analyse de données	Application de complexité très faible	1
	Application de complexité faible	2
	Application de complexité moyenne	3
	Application de complexité élevée	4
	Application de complexité très élevée	5
	Projet en cours de développement	3

Niveau Complexité	Critères
Très faible	Note de complexité = 1
Faible	Note de complexité = 2
Moyen	2 < Note de complexité <= 4
Elevé	4 < Note de complexité <= 7
Très élevé	7 < Note de complexité <= 10

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- le respect des normes de documentation,
- sur la conformité aux documents d'entrée,
- sur 'exhaustivité et la pertinence du contenu de la documentation livrée.

et – le cas échéant – aux résultats des échanges avec les représentants utilisateurs ou des maîtrises d'ouvrage ou d'œuvre.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[12] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP

4.6.3 Unité d'œuvre ETD_UX_UI – Etude d'UX-UI

Présentation

Il s'agit pour le titulaire de réaliser une étude UX/UI en adoptant une démarche centrée sur l'utilisateur sur une application dans l'objectif d'en améliorer son utilisabilité ou pour un projet en phase de conception.

Le titulaire décrit dans son offre son approche UX/UI et la démarche qui sera mis en œuvre.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée d'un document d'expression de besoins.
- Charte graphique du CNRS

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Animer les ateliers et rédiger les comptes rendus de ces ateliers :
    - Ateliers d'analyses des besoins et des attentes des utilisateurs finaux et des exigences et des attentes de la maitrise d'ouvrage (MOA) et de la maîtrise d'œuvre (MOE)
    - Ateliers d'analyse de l'existant et du parcours utilisateur ;
    - Ateliers de stratégie UX
    - Ateliers de restitution
- Analyser les besoins et des attentes des utilisateurs finaux et des exigences et des attentes de la maitrise d'ouvrage (MOA) et de la maîtrise d'œuvre (MOE)
- Analyser le contexte d'utilisation de la solution et la qualité de l'expérience proposée par la solution ;
- Faire des propositions de choix à opérer en matière de design et d'expérience utilisateur (stratégie UX) en prenant en compte les règles d'accessibilité (cf. CCT).
- Rédiger un rapport de diagnostic de la situation présentant notamment
    - les points faibles de la solution, en faisant ressortir ce qui est à conserver et ce qui est à optimiser,
    - les axes de la stratégie digitale et les choix à opérer en matière de design d'expérience utilisateur (cartographie de l'expérience utilisateurs, stratégie UX) ;
- Proposer et concevoir en collaboration avec le CNRS un nouveau parcours utilisateur :
    - les wireframes,
    - le design visuel,
    - les maquettes ou/et les prototypes interactifs, simulation de l'expérience utilisateur sur la solution ;
- Rédiger les éléments permettant l'évaluation/validation de la réponse au besoin par les utilisateurs :
    - le plan de test utilisateur (types et scénarios des tests, critères de succès, …) ;
    - le rapport de test utilisateur (résumé des observations, retours et recommandations basées sur les tests utilisateur) ;
- Vérifier par contrôle qualité interne au titulaire, le respect des normes de documentation ;

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Rapport de recherche utilisateur incluant le rapport de diagnostic (résultats d'enquêtes, interviews, tests utilisateur, etc.), les personas UX, la cartographie d'expérience utilisateur, la stratégie UX ;
- Plan de test utilisateur (types et scénarios des tests, critères de succès, …) ;
- Rapport de test utilisateur (résumé des observations, retours et recommandations basées sur les tests utilisateur) ;
- Fichiers sources des wireframes, maquettes ou prototypes ;

- Déclinaison de la charte graphique CNRS.

Complexité

Le calcul de complexité ne doit prendre en compte – en tant que nombre d'informations mises en jeu – que les éléments faisant l'objet de la demande.

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Ateliers (comprend : la préparation en amont, la tenue de l'atelier et la rédaction du CR)	Simple : atelier inférieur ou égal à 2h (réponse points ouverts, clarification, validation de maquettes, restitution…)	Atelier	2
	Normal : atelier dont la durée est > 2h et <= 4h inclus (analyse de l'existant, présentation de maquettes, …)	Atelier	4
Nombre d'écrans à étudier	Simple : nombre d'écrans inférieur ou égal à 5	Ecran	2
	Normal : nombre d'écrans entre 6 et 10 inclus	Ecran	6
	Complexe : nombre d'écrans entre 11 et 20 inclus	Ecran	12
Réalisation de maquettes Maquette simple : nombre d'informations mises en jeu <10 **Maquette normale :** 10<=nombre d'informations mises en jeu <25 Maquette Complexe : nombre d'informations mises en jeu >=25	Simple : Création maquette simple ou réutilisation d'une maquette normale (exemple, maquette normale pour la création, réutilisation pour la modification ou la consultation)	Maquette	2
	Normal : Création d'une maquette normale ou réutilisation d'une maquette complexe (exemple, maquette complexe pour la création, réutilisation pour la modification ou la consultation)	Maquette	4
	Complexe : Création d'une maquette complexe	Maquette	6

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau Complexité	Critères
Très faible	Note de complexité <= 15
Faible	15 < Note de complexité < 20
Moyen	20 < Note de complexité <= 25
Elevé	25 < Note de complexité <= 30
Très élevé	30 < Note de complexité <= 35

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- le respect des normes de documentation,
- sur la conformité aux documents d'entrée,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée.

et – le cas échéant – aux résultats des échanges avec les représentants de la maitrise d'œuvre.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[13] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.7 PRESTATION 7 : MIGRATIONS TECHNIQUES (MONTÉES DE VERSION)

4.7.1 Présentation de la prestation

Cette prestation consiste à réaliser ou mettre à jour une ou plusieurs briques techniques.

Ces briques peuvent être :

- la mise à jour de composants techniques (montée de version majeure ou mineure), cf **Erreur ! Source du renvoi introuvable. Erreur ! Source du renvoi introuvable.**
- la migration de composants techniques (changement de composant…)..

4.7.2 Unité d'œuvre ETD_MIG_TECH – Etude de migration technique

Présentation

Il s'agit pour le titulaire de rédiger une étude détaillée des actions à entreprendre lors de la migration/montée de version majeure ou mineure d'un composant technique. L'UO porte uniquement sur l'impact de la migration sur l'application et non sur la mise en œuvre proprement dite de la migration.

Cette migration/montée de version peut concerner :

- le système d'exploitation
- le système de base de données
- le CMS (DRUPAL, ..) et ses modules
- le langage de l'application (PHP, …)
- le cadriciel utilisé (Symfony, Angular, …)
- la mise en place de la compatibilité avec un navigateur non supporté jusqu'à présent par l'application
- une librairie tierce utilisée par l'application
- un changement de technologie des flux
- ...

Les montées de versions mineures des composants applicatifs doivent être analysées maximum 3 mois après leur publication, en accord avec le CNRS.

Pour les versions majeures, la date de démarrage de l'étude sera décidée conjointement avec le CNRS.

Les réalisations sont ensuite planifiées conjointement en COSUI, avec des commandes d'adaptatifs si la charge le nécessite

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande du CNRS ;
- Animer les ateliers éventuels et rédiger les comptes rendus de ces ateliers ;
- Réaliser l'étude d'impact et rédiger un rapport faisant indiquant les actions à effectuer, les préconisation du titulaire, les points de vigilance à vérifier à l'issue de la migration et les éléments ayant conduits au résultat ;

[13] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

- Présenter les résultats de l'étude au CNRS, pouvant contenir les points suivants en fonction de la nature de l'étude :
  - l'identification des entités Drupal à créer/modifier (par exemple : Content Types, Taxonomy, Custom Entities),
  - la description des relations entre entités (par exemple : Entity Reference, Relationships dans Views),
  - l'identification des écrans à modifier ou la description des pages/formulaires à créer (par exemple : Views, Blocks, Webform, Custom Forms avec Form API),,
  - l'identification et la compatibilité des thèmes, des librairies frontend (jQuery, Bootstrap, etc.),
  - l'identification des services ou plugins à développer ou à adapter (par exemple : Custom Modules, Services Symfony) en identifiant la complexité,
  - l'identification des hooks ou events à utiliser (par exemple : hook_node_presave(), Event Subscribers),
  - l'identification des traitements batch à implémenter (par exemple : Batch API pour les imports massifs, Queue API pour les tâches asynchrones),
  - la description des règles à appliquer en cas de modification du modèle des données pour assurer la migration des données,
  - l'identification des traitements à modifier ou la description des traitements à créer,
  - l'identification des reports ou exports à créer (par exemple : Views avec export CSV/PDF, Print module pour des PDF),
  - l'identification des autres composants directement impactés par cette migration technique,
  - l'identification des modifications à apporter sur l'intégration continue,
  - l'identification des documentations à modifier.

Le titulaire assurera son devoir de conseil en étant force de propositions et garantira dans ses propositions une parfaite homogénéité de l'application.

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Rapport d'étude ou d'expertise regroupant les conclusions sur l'ensemble des éléments sur lesquels a porté l'étude; indiquant clairement les actions à effectuer, les éléments d'architecture à modifier mais aussi les points de vigilance à vérifier à l'issue de la migration et les éléments ayant conduits au résultat ;
- Fiche de traçage des actions à effectuer lors de la migration ;
- Présentation du rapport d'étude ou d'expertise ;
- Proposition de planning et de charge associée pour mettre en œuvre l'objet de la demande, exprimée en unités d'œuvre.

Complexité

Cf § 3.2.3 Complexité des unités d'œuvre de montée de version

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- le respect des normes de documentation,
- sur la conformité aux documents d'entrée,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée.

et – le cas échéant – aux résultats des échanges avec les représentants de la maitrise d'œuvre.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[14] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

[14] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

4.7.3 Unité d'œuvre REA_MIG_TECH – Réalisation de migration technique

Présentation

Cette unité d'œuvre a pour objectif la migration ou la montée de version majeure ou mineure, d'outils ou d'éléments technologiques.

Cette unité d'œuvre couvre également la mise en place de la compatibilité avec un navigateur non supporté jusqu'à présent par l'application.

Eléments en entrée

Les éléments d'entrée peuvent être :

- Etude d'impact de la migration sur l'application ;
- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins.
- Fiche de traçage des actions à effectuer lors de la migration ;
- Spécifications fonctionnelles et techniques détaillées ;
- Dossier d'architecture ;
- Autres éléments techniques nécessaires à la migration.

Actions à réaliser

Dans tous les cas, les actions à réaliser par le titulaire sont les suivantes :

- Animer, si nécessaire, des ateliers fonctionnels ou techniques pour réaliser la solution et rédiger les comptes rendus de ces ateliers ;
- Mettre en œuvre la solution technique et fonctionnelle:
  - Ajuster l'architecture applicative et technique des données (modèle physique des données) et des traitements (découpage et articulation des composants logiciels...) ;
  - Ajuster le code si nécessaire ;
  - Ajuster les options permettant de garantir une modularité optimale des traitements, le niveau de paramétrage attendu, le niveau et le format des logs et les performances optimales ;
  - Si besoin, définir le dimensionnement (puissance et stockage) des plateformes de recette et de production ;
  - Réaliser les traitements correspondant à la demande ;
- Mettre à jour la documentation technico-fonctionnelle et en particulier :
  - les spécifications fonctionnelles détaillées(cf. la description du contenu au § 5.2.2.1) si nécessaire,
  - les spécifications techniques détaillées (cf. la description du contenu au § 5.2.2.2),
  - le dossier d'architecture logicielle (cf. la description du contenu au § 5.2.3.1),
- Mettre à jour la fiche descriptive du développement réalisé dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution mise en œuvre et le statut ;
- Mettre à niveau l'environnement de développement afin d'être en conformité avec l'architecture cible définie
- Exécuter les migrations dans l'environnement d'intégration lorsqu'il existe avant livraison au CNRS ;
- Réaliser les tests suivants[15] sur différentes plateformes (OS, navigateurs) depuis l'environnement de développement et l'environnement d'intégration lorsqu'il existe :
  - tests unitaires automatisés,
  - tests unitaires manuels,
  - tests d'intégration,
  - tests de validation,
  - tests de non régression de l'application,
  - tests de performance unitaires et/ou plus globaux en fonction des travaux réalisés,
  - tests de sécurité (pour les versions majeures) ;

Le plan de test n'est pas fourni par le CNRS mais des données de test peuvent être fournies par le CNRS.

[15] Cf. la définition des différents types de tests dans le glossaire au § 5.1.2

- Spécifier les procédures permettant l'installation, la mise en exploitation et l'exploitation de l'application,
  - Créer si besoin un script d'installation (script automatisant l'initialisation/mise à jour de la base de données, etc.) ;
  - Rédiger ou mettre à jour les manuels d'installation et d'exploitation ; Pour les applications déployées par Ansible, mettre à jour via une merge-request dans la forge du CNRS la configuration de déploiement Ansible, et indiquer toutes les modifications à apporter pour prendre en compte la migration;
- Assister techniquement si besoin les équipes CNRS pour la mise en œuvre de la plateforme de recette dans les conditions d'architecture spécifiées (conformément au cadre de cohérence technique tel que décrit dans le paragraphe 4 du CCTP Livret 1 et dans le CCT_versions_cibles du CNRS) ;
- Livrer la version réalisée dans l'environnement de livraison du CNRS avec Ansible le cas échéant ;
- Vérifier la qualité du code livré en s'assurant que l'analyse Sonar du nouveau code ne remonte pas de problèmes bloquants, majeurs ou de sécurité ;
- Assister l'équipe technique du CNRS pour assurer l'installation dans l'environnement de recette ;
- Prendre en compte les éventuelles anomalies[16] de recette détectée par le CNRS et procéder aux relivraisons nécessaires (phase de Vérification d'Aptitude) ;
- Assister si besoin les équipes CNRS, lors de l'installation et de la mise en exploitation dans l'environnement de production ;
- Prendre en compte les éventuelles anomalies de production détectées par le CNRS et procéder aux relivraisons nécessaires (phases de Vérification de Service Régulier et de garantie) ;
- Vérifier par contrôle qualité interne au titulaire, le respect des normes de développement (définis dans le CCT) et de documentation ;

Le titulaire doit réaliser les actions indiquées dans la fiche de traçage des actions. Elles dépendent du type de migration :

Dans le cas d'une migration concernant le SGBD :

- Créer la nouvelle base de données ;
- Faire évoluer les scripts et éléments en langage SGBD ;
- Adapter le code applicatif si nécessaire
- Adapter le paramétrage ;
- Vérifier la migration.

Dans le cas d'une migration concernant le système d'exploitation :

- Faire évoluer les scripts systèmes (shell Script)
- Adapter le paramétrage
- Adapter le code applicatif si nécessaire
- Vérifier la migration

Dans le cas d'une migration concernant le CMS

- Faire évoluer les composants applicatifs impactés ;
- Adapter le code applicatif si nécessaire
- Adapter le paramétrage ;
- Vérifier la migration.

Dans le cas d'une migration concernant le socle applicatif, une librairie tierce ou de technologie de flux

- Faire évoluer les composants applicatifs impactés ;
- Adapter le paramétrage ;
- Vérifier la migration.

[16] La définition d'une anomalie est décrite dans le glossaire au § 5.1.2.

Dans le cas d'une mise en place de la compatibilité avec un navigateur non supporté :

- Faire évoluer les composants applicatifs impactés ;
- Vérifier la compatibilité.

Livrables

Dans tous les cas le titulaire remet les livrables suivants :

- Animation et comptes rendus des ateliers menés ;
- Fiche descriptive du développement réalisé mise à jour dans l'outil de gestion des anomalies et des évolutions du CNRS ;
- Spécifications fonctionnelles détaillées mises à jour (cf. la description du contenu au § 5.2.2.1) si nécessaire ;
- Spécifications techniques détaillées mises à jour (cf. la description du contenu au § 5.2.2.2) ;
- Dossier d'architecture logicielle mis à jour (cf. la description du contenu au § 5.2.3.1) ;
- Selon les modalités spécifiques à la plateforme :
  - Ensemble de tous les fichiers, y compris les composants en langage source pour tous les éléments modifiés (y compris les scripts Shell, procédures SGBD, fichiers de paramétrages…)., correspondant à la branche ou à l'étiquette livrée et permettant la génération de l'exécutable, livré dans la forge du CNRS ;
  - Ensemble des composants en forme exécutable livrés dans l'environnement de livraison du CNRS, y compris les fichiers de paramétrage, les ressources statiques ;
  - Scripts d'installation éventuels ;
- Documents de paramétrage créé ou mis à jour *(facultatif)* ;
- Rapport de tests unitaires (noms, date, numéro de version) ;
- Plan de tests et résultats des tests déroulés ;
- Dossier de validation avec fiche de traçage des actions remplie ;
- Manuel d'exploitation mis à jour (cf. la description du contenu au § 5.2.3.2) ;
- Manuel d'installation, comprenant la mise à niveau si nécessaire de la configuration des playbooks par le CNRS (cf. la description du contenu au § 5.2.3.3) ;

Le cas échéant :

- Nouvelle base de données.

Complexité

Cf § 3.2.3 Complexité des unités d'œuvre de montée de version

Modalité de réception par le CNRS

La recette porte, outre sur le constat de remise des livrables, sur le respect des normes techniques de programmation, de documentation et d'exploitation, ainsi que sur la conformité de la documentation technico-fonctionnelle.

La réception de la prestation est prononcée après la vérification que tous les éléments sont réalisés.

Lors de la recette (Vérification d'Aptitude), le CNRS peut, a minima :

- vérifier la complétude des éléments livrés ;
- vérifier le bon déroulement de l'installation de l'application livrée ;
- exécuter des tests de recette sur le périmètre applicatif de la livraison (dans l'environnement de recette) ;
- vérifier la qualité du code (application des normes…), la complétude des tests et jeux de tests du titulaire, le passage avec succès de ces tests dans l'environnement du titulaire ;
- vérifier la « robustesse » de la solution et la montée en charge des utilisateurs ;
- vérifier l'exhaustivité et la pertinence du contenu de la documentation livrée.

Lors de la Vérification de Service Régulier (VSR), le CNRS vérifie l'utilisation de l'application dans des conditions normales en environnement de production.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[17] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.8 PRESTATION 8 : CONDUITE DU CHANGEMENT

4.8.1 Présentation de la prestation

Le changement de plateforme technologique pour les 4 applications du périmètre initial nécessite un accompagnement structuré pour garantir une adoption efficace par les contributeurs. Cet accompagnement repose sur deux dimensions principales : la formation et la réalisation de support d'aide.

Ces activités de conduite du changement ne sont confiées au titulaire que lorsque ponctuellement le CNRS en exprime le besoin.

4.8.2 Unité d'œuvre FORM_ORG – Organisation et conception de formation

Présentation

Il s'agit d'organiser et de concevoir des formations permettant aux contributeurs de maîtriser l'utilisation des applications, les interfaces et les fonctionnalités spécifiques développées.

L'objet de ces formations est d'acquérir des connaissances théoriques et pratiques. La formation doit prendre en compte les spécificités mises en œuvre dans les applications pour les besoins du CNRS.

L'enseignement devant largement faire appel à des travaux pratiques, il convient de bien définir les moyens pédagogiques mis à la disposition des stagiaires.

Pour accompagner la formation des utilisateurs, un ensemble de supports pédagogiques est requis pour permettre aux différents profils d'utilisateurs de maîtriser les interfaces front et back notamment sur les spécificités fonctionnelles développées pour répondre aux besoins du CNRS. La documentation proposée (en français) pourrait être constituée par exemple d'un livret destiné aux formateurs, et de supports de présentation pour les participants. Ces documents doivent être régulièrement mis à jour afin de rester alignés avec les évolutions des applications.

Les formations doivent être conçues sous forme de modules distincts, permettant d'isoler les séquences théoriques des exercices pratiques. Lors de l'animation des formations, les séquences pourront être enregistrées afin de servir de référence pour les formations ultérieures ou pour de l'auto-formation des utilisateurs. L'enregistrement doit être structuré de manière à pouvoir être consulté de façon segmentée, selon les besoins des utilisateurs.

La formation étant amenée à être poursuivie en interne CNRS, un transfert de compétences progressif doit être organisé afin que les équipes CNRS puissent assurer la montée en compétence des nouveaux contributeurs de manière autonome.

Qualification des intervenants

Les personnels du titulaire intervenant sur les actions de formation sont qualifiés pour ces activités : par formation et/ou expérience.

Evaluation des stagiaires

Les modalités d'évaluation, de reconnaissance et/ou de validation des acquis des stagiaires sont formalisées en amont de la formation et diffusées aux stagiaires. Le titulaire s'engage à être à l'écoute des stagiaires et à recueillir leurs attentes et réclamations ainsi qu'à en tenir compte dans la mesure de leur recevabilité. Il doit prévoir les modalités de réajustement en cours de formation.

Un taux de satisfaction minimal des stagiaires à atteindre est défini en début de prestation d'un commun accord entre le titulaire le CNRS.

Initialisation de la base de formation

Il s'agit d'initialiser la base de formation avec les données et jeux de tests pertinents nécessaires aux exercices pratiques.

Cette base de formation doit être mise à niveau avec le paramétrage et les programmes de la version la plus récente pour permettre le bon déroulement des sessions de formation.

Eléments en entrée

- Document d'expression des besoins précisant le périmètre de la formation et la population ciblée ;

[17] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

- Fonctionnalités des applications.
- Documentation technique et fonctionnelle.

Actions à réaliser

- Réaliser le plan de formation détaillé incluant le nombre de modules, le contenu adapté aux profils utilisateurs, les moyens pédagogiques et le calendrier des sessions ;
- Concevoir et réaliser les supports de formation (livret formateur, supports de présentation, cahier d'exercices) ;
- Initialiser la base de formation : effectuer le paramétrage et compléter les données (données et jeux de tests pour les exercices, rôles, etc...).
- Définir les modalités d'évaluation et d'ajustement des formations

Livrables

- Plan de formation (nombre de modules de formation, contenu en fonction des personnels cibles, moyens nécessaires, planning des sessions de formation, scénario de formation, …) ;
- Supports de formation mis à jour avec la version courante de l'application (livret formateur, supports de présentation, cahier d'exercices avec jeux d'essai associés, modèles des fiches de présence et d'évaluation) ;
- Base de formation initiée et mise à jour avec la version courante de l'application.

Complexité

Basée sur le nombre de jours de la formation à préparer :

Niveau Complexité	Critères
Très faible	< 1 jour de formation
Faible	1 jour de formation
Moyen	2 jours de formation
Elevé	3 jours de formation
Très élevé	> 3 jours de formation

4.8.3 Unité d'œuvre FORM_ANI – Animation de formation

Présentation

Il s'agit d'animer ou de co-animer avec le CNRS des sessions de formation auprès de formateurs ou auprès des contributeurs et de faire un bilan des formations réalisées.

Les sessions de formation pourront être enregistrées afin de servir de référence pour les formations ultérieures ou de l'auto-formation des utilisateurs.

Éléments en entrée

- Plan de formation avec le planning des formations et le lieu du déroulement ;
- Supports de formation (livret formateur, supports de présentation, cahier d'exercices avec jeux d'essai associés, modèles des fiches de présence et d'évaluation) ; les supports sont fournis sous forme électronique ;
- Base de formation initiée ;

Actions à réaliser

- Animer ou co-animer avec le CNRS les sessions de formation ;
- Enregistrer les sessions pour permettre des consultations ultérieures ;
- Assurer un suivi des stagiaires en recueillant leurs évaluations et en réalisant une synthèse des retours.

Livrables

- Support de formation remis à chaque stagiaire sous format électronique ;

- Formations dispensées ;
- Fiches d'évaluations remplies ;
- Synthèse des évaluations et bilan des formations ;
- Taux de satisfaction minimal des stagiaires atteint.

Complexité

La complexité de chaque demande d'animation de formation est basée sur le nombre de jours de la formation à animer et les modalités opérationnelles :

- Formation en présentiel à Paris ou Région Parisienne
- Formation en présentiel en Province (dans les délégations régionales du CNRS)
- Formation en distanciel.

L'unité d'œuvre correspond à un coût journalier.

4.8.4 Unité d'œuvre SUPPORT_AIDE – Production de supports d'aide

Présentation

Il s'agit de produire des supports d'aide à l'utilisation des applications par les contributeurs pour une prise en main efficace et une utilisation fluide des fonctionnalités mises en place.

Ces supports doivent être régulièrement actualisés pour refléter les évolutions fonctionnelles des applications. Ils doivent être conçus de manière claire et pédagogique, en privilégiant une approche didactique adaptée à différents niveaux d'expertise. Les supports d'aide pourront être organisés par profil d'utilisateur.

Ces supports peuvent prendre la forme de guides détaillés, de notices explicatives ou de tutoriels sous format texte et vidéo. Ils doivent couvrir l'ensemble des actions récurrentes effectuées, en mettant l'accent sur les bonnes pratiques et les erreurs à éviter.

Éléments en entrée

- Document d'expression des besoins précisant la volumétrie et le type de supports attendus ;
- Documentation fonctionnelle des applications

Actions à réaliser

- Réaliser tout type de support (notice, guide, manuel, tutoriel…) à destination des utilisateurs.
- Mettre à jour les supports en fonction des évolutions des applications

Livrables

- Supports produits.

Complexité

Basée sur le type et le nombre de supports à produire :

Niveau Complexité	Critères
Très faible	1 document ou tutoriel
Faible	Entre 2 et 5 documents ou tutoriels
Moyen	Entre 6 et 10 documents ou tutoriels
Elevé	Entre 11 et 15 documents ou tutoriels
Très élevé	Entre 16 et 20 documents ou tutoriels

.

4.9 PRESTATION 9 : TESTS SPÉCIFIQUES SUPPLÉMENTAIRES

4.9.1 Présentation de la prestation

Les prestations de tests spécifiques supplémentaires ne sont confiées au titulaire que lorsque ponctuellement le CNRS en exprime le besoin sur une partie de l'application, ou sur l'ensemble de l'application. Le titulaire doit intervenir en étroite collaboration avec les équipes CNRS

Ces prestations ne viennent pas en remplacement des tests prévus dans les prestations de développement ou de réalisation d'évolutions.

Le titulaire ne bénéficie pas d'une exclusivité pour l'exécution de cette prestation

Le titulaire décrit dans son offre la démarche et l'outillage mis en œuvre pour chacun des tests.

4.9.2 Unité d'œuvre TEST_AUTO - Tests automatisés

Présentation

Cette unité d'œuvre a pour objectif la mise en œuvre et l'exécution de tests automatisés (sur le front et/ou sur le back), en particulier pour une application existante n'ayant pas suffisamment de tests automatisés lors de son intégration dans le marché.
L'ajout de nouveaux tests et leur première exécution est déclenché uniquement si des risques sont identifiés à la demande du CNRS. Les exécutions suivantes sont obligatoires dans le cadre des tests « usuels » prévus dans les UO de réalisation y compris si le code concerné n'est pas modifié dans la version livrée.

L'unité d'œuvre comprend une partie « étude » qui ne fera pas l'objet d'une commande d'unités d'œuvre d'études complémentaires.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins précisant a minima :
  - soit la liste des fonctionnalités à couvrir par les tests automatisés,
  - soit la liste des cas de tests à mettre en œuvre ;
- Spécifications techniques détaillées, si le document existe ;
- Normes et standard de programmation et d'exploitation ;
- Description des contraintes particulières d'exploitation éventuelles ;
- Données de tests si besoin.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande du CNRS ;
- Animer des ateliers pour affiner le besoin et rédiger les comptes rendus de ces ateliers ;
- Définir la stratégie de test
  - Rédiger ou mettre à jour le plan de tests automatisés liés aux scénarios spécifiés,
  - Identifier la liste des composants devant faire l'objet de tests et les cas de tests à automatiser sur ces composants ;
- Mettre en place l'outillage de tests si l'application n'en est pas déjà pourvue ;
- Créer ou mettre à jour les scripts liés aux cas de tests à automatiser ;
- Intégrer les tests automatisés dans le développement de l'application ;
- Exécuter le plan de test ;

- Adapter si nécessaire le code applicatif en cas de détection d'anomalie remontée par la mise en place des tests ;
- Adapter si nécessaire des jeux de données existants au périmètre à valider ;
- Rédiger un rapport de résultat des tests ;
- Mettre à jour la fiche descriptive du développement réalisé dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution mise en œuvre et le statut ;
- Présenter le rapport de résultat des tests.

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Stratégie de test ;
- Outillage de tests : les scénarios développés doivent être mis à disposition CNRS, selon les modalités spécifiques à la plateforme :
  - Ensemble de tous les fichiers, y compris les composants en langage source, correspondant à la branche ou à l'étiquette livrée et permettant l'exécution des tests automatisés et la génération de l'exécutable, livrés dans la forge du CNRS ;
  - Ensemble des composants en forme exécutable livrés dans l'environnement de livraison du CNRS, y compris les fichiers de paramétrage, les ressources statiques ;
  - Scripts d'installation éventuels ;
- Manuel d'exploitation mis à jour (cf. la description du contenu au § 5.2.3.2) si nécessaire ;
- Manuel d'installation, comprenant la mise à niveau si nécessaire de la configuration Ansible et la description des modifications à apporter sous forme de merge-request dans la forge du CNRS (cf. la description du contenu au § 5.2.3.3) ;
- Rapport de résultat des tests.

Complexité

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Ecriture d'un test automatisé	Normal : écriture de test pour scénario de consultation par exemple se connecter, recherche/consultation …	Scenario de test	2
	Complexe : écriture de test pour un scenario de saisie dans un formulaire …	Scenario de test	4
Mise à jour d'un test automatisé	Simple : mise à jour d'un test existant pour un scenario de test de type normal	Scenario de test	1
	Normal : mise à jour d'un test existant pour un scenario de test de type complexe	Scenario de test	2
Mise en place initiale de l'outillage [18]	Installation de l'outillage	L'installation	2
Exécution et analyse		Exécution	2

Le niveau de complexité de l'UO résulte de la somme des notes de complexité des critères ci-dessus :

Niveau de Complexité	Critères
Très faible	Note de complexité <= 4
Faible	4 < Note de complexité <= 6
Moyenne	6 < Note de complexité <= 8
Elevée	8< Note de complexité <=12

[18] L'installation de l'outil de tests ne sera valorisée que la première fois.

Niveau de Complexité	Critères
Très élevée	12 < Note de complexité <=18

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

De plus, lors de l'écriture et de l'exécution de scénario de tests sur des fonctionnalités similaires ou proches des coefficients de réutilisation variant entre 0,5 et 0,9 doivent être utilisés.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- sur la conformité aux documents d'entrée,
- sur la conformité des résultats obtenus à ceux attendus,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[19] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.9.3 Unité d'œuvre TEST_PERF - Tests de performances

Présentation

Il s'agit pour le titulaire de mettre en œuvre et d'exécuter des tests (automatisés ou non) pour évaluer la performance de l'application dans le cadre de scenarios de test donnés, d'analyser le résultat et de proposer des axes d'amélioration.

Les niveaux de performances attendus de l'application et les modalités de mesure des temps de réponse, sont décrits au §4.2.6.3 du CCTP livret 1.

Des tests sont déclenchés uniquement si des risques sont identifiés, à la demande du CNRS.

L'unité d'œuvre comprend une partie « étude » qui ne fera pas l'objet d'une commande d'unités d'œuvre d'études complémentaires.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins précisant a minima :
  - soit la liste des fonctionnalités à couvrir par les tests automatisés,
  - soit la liste des cas de tests à mettre en œuvre ;
- Spécifications techniques détaillées, si le document existe ;
- Normes et standard de programmation et d'exploitation ;
- Description des contraintes particulières d'exploitation éventuelles ;
- Données de tests si besoin.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande du CNRS ;
- Animer des ateliers pour affiner le besoin et rédiger les comptes rendus de ces ateliers ;
- Créer et actualiser la feuille de route avec les acteurs concernés (y compris l'exploitant) pour le planning et l'organisation de tests,
- En tant que de besoin, créer ou mettre à jour des scripts liés aux scenarios spécifiés,
- Adapter si nécessaire des jeux de données existants au périmètre à valider,

[19] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

- Mettre en œuvre l'outillage initial de tests s'il n'est pas déjà en place ;
- Ensemble de tous les fichiers, y compris les composants en langage source, correspondant au code écrit pour les tests de performances, livrés dans la forge du CNRS ;
- Installer et exécuter les scénarios dans l'environnement de développement du titulaire et collecter des performances,
- Installer et exécuter les scénarios dans l'environnement d'intégration, s'il existe, et collecter des performances,
- Assister le CNRS pour installer et exécuter les scénarios dans l'environnement de recette spécifié par le CNRS et collecter des performances,
- Analyser les résultats,
- Rédiger un rapport de tests de performances :
  - mettant particulièrement en avant les éléments chiffrés concrets qui permettent de valider les temps de réponses exigés par le CNRS ;
  - mettant aussi en évidence les composantes techniques de l'application nécessitant le plus de ressources techniques et induisant des temps de réponses significatifs ;
  - présentant des propositions d'améliorations si nécessaire ;[20]
- Mettre à jour la fiche descriptive du développement réalisé dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution mise en œuvre et le statut ;
- Présenter le rapport de tests de performance.

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Feuille de route ;
- Scripts de charge liés aux scenarios spécifiés et leur documentation ;
- Modules logiciels nécessaires à l'exécution des tests (bouchons…) ;
- Rapports de test suite à (aux) exécution(s).

Complexité

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Ecriture du script	Normal : écriture de test pour scénario de consultation par exemple se connecter, recherche/consultation …	Scenario de test	2
	Complexe : écriture de test pour un scenario de saisie dans un formulaire …	Scenario de test	4
Mise à jour d'un test automatisé	Simple : mise à jour d'un test existant pour un scenario de test de type normal	Scenario de test	1
	Normal : mise à jour d'un test existant pour un scenario de test de type complexe	Scenario de test	2
Mise en place initiale de l'outillage [21]	Installation de l'outillage	L'installation	2
Exécution et analyse		Exécution	2

Niveau de Complexité	Critères
Très faible	Note de complexité <= 4
Faible	4 < Note de complexité <=6

[20] Dans le cas où les résultats des tests aboutiraient à des propositions d'amélioration relevant de corrections de code non conforme aux best-practices et dans le cas où ce code relève de la responsabilité du titulaire, ces corrections seront pris en charge par le titulaire dans le cadre du MCO.

[21] L'installation de l'outil de tests ne sera valorisée que la première fois.

Niveau de Complexité	Critères
Moyenne	6 < Note de complexité <=8
Elevée	8 < Note de complexité <=12
Très élevée	12 < Note de complexité <=18

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

De plus, lorsqu'une exécution supplémentaire est à effectuer des coefficients de réutilisation variant entre 0,5 et 0,9 doivent être utilisés.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- sur le respect des normes techniques de programmation, de documentation et d'exploitation,
- sur la conformité aux documents d'entrée,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[22] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.9.4 Unité d'œuvre TEST_CHARGE - Tests de charge

Présentation

Il s'agit pour le titulaire de mettre en œuvre et d'exécuter des tests automatisés pour évaluer la résistance à la charge de l'application dans le cadre de scénarios de tests donnés puis d'analyser le résultat et de proposer des axes d'amélioration.

Ces tests seront réalisés dans l'environnement de développement du titulaire. Toutefois ils pourront être réalisés dans un autre environnement d'un commun accord entre le CNRS et le titulaire.

Tests déclenchés uniquement si des risques sont identifiés à la demande du CNRS.

L'unité d'œuvre comprend une partie « étude » qui ne fera pas l'objet d'une commande d'unités d'œuvre d'études complémentaires.

Eléments en entrée

Les éléments d'entrée sont :

- Fiche descriptive de la demande dans l'outil de gestion des anomalies et des évolutions du CNRS (Mantis) complétée si besoin d'un document d'expression de besoins précisant :
  - les scénarios à exécuter,
  - la charge à simuler (par exemple exécuter pour 100 utilisateurs simultanés, puis 1000, …),
- Spécifications techniques détaillées, si le document existe ;
- Normes et standard de programmation et d'exploitation ;
- Description des contraintes particulières d'exploitation éventuelles ;
- Données de tests si besoin.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Analyser la demande du CNRS ;
- Animer des ateliers pour affiner le besoin et rédiger les comptes rendus de ces ateliers ;
- Créer et actualiser la feuille de route avec les acteurs concernés pour le planning et l'organisation de tests ;
- Créer ou mettre à jour des scripts de charges liés aux scenarios spécifiés ;

[22] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

- Installer et exécuter les scénarios dans l'environnement de développement du titulaire et collecter des performances ;
- Rédiger un rapport de tests de charge :
  - mettant particulièrement en avant les éléments chiffrés concrets qui permettent de valider les temps de réponses exigés par le CNRS en condition de forte charge applicative ;
  - mettant aussi en évidence les composantes techniques de l'application nécessitant le plus de ressources techniques et induisant des temps de réponses significatifs ;
  - présentant des propositions d'améliorations si nécessaire ;
- Analyser les résultats,
- Mettre à jour la fiche descriptive du développement réalisé dans l'outil de gestion des anomalies et des évolutions du CNRS : indiquer, en particulier, la solution mise en œuvre et le statut ;
- Présenter le rapport de tests de charge.

Livrables

Les livrables pour cette unité d'œuvre sont :

- Animation et comptes rendus des ateliers menés ;
- Feuille de route ;
- Scripts de charge liés aux scenarios spécifiés et leur documentation ;
- Modules logiciels nécessaires à l'exécution des tests (bouchons…) ;
- Rapports de test suite à (aux) exécution(s).

Complexité

Chaque élément possède une note. La note globale est la somme des notes individuelles de chaque élément.

Voici la liste des notes individuelles de chaque élément possible :

Elément particulier	Niveau de complexité	Unité	Note de complexité
Ecriture du script	Normal : écriture de test pour scénario de consultation par exemple se connecter, recherche/consultation …	Scenario de test	2
	Complexe : écriture de test pour un scenario de saisie dans un formulaire …	Scenario de test	4
Mise à jour d'un script	Simple : mise à jour d'un test existant pour un scenario de test de type normal	Scenario de test	1
	Normal : mise à jour d'un test existant pour un scenario de test de type complexe	Scenario de test	2
Mise en place initiale de l'outillage [23]	Installation de l'outillage	L'installation	2
Exécution et analyse	De 1 à 5 tirs (exécution d'un ou plusieurs scripts sur N utilisateurs simulés)	Nombre de tirs	2
	De 6 à 10 tirs (exécution d'un ou plusieurs scripts sur N utilisateurs simulés)	Nombre de tirs	4
	>10 tirs (exécution d'un ou plusieurs scripts sur N utilisateurs simulés)	Nombre de tirs	6

Niveau de Complexité	Critères
Très faible	Note de complexité <= 4
Faible	4 < Note de complexité <= 6
Moyenne	6 < Note de complexité <= 8
Elevée	8 < Note de complexité <=12
Très élevée	12 < Note de complexité <=18

[23] L'installation de l'outil de tests ne sera valorisée que la première fois.

Coefficient de majoration ou minoration

Cf § 3.3 Coefficient de majoration ou minoratio.

De plus, lorsqu'une exécution supplémentaire est à effectuer des coefficients de réutilisation variant entre 0,5 et 0,9 doivent être utilisés.

Modalité de réception par le CNRS

La recette porte, outre le constat de remise des livrables, sur :

- sur le respect des normes techniques de programmation, de documentation et d'exploitation,
- sur la conformité aux documents d'entrée,
- sur l'exhaustivité et la pertinence du contenu de la documentation livrée.

Les non-conformités éventuelles détectées par le CNRS sont priorisées en fonction de leur gravité[24] et leur correction doit être prise en compte dans le cadre d'une relivraison conformément aux dispositions précisées dans le Livret 1 du CCTP.

4.10 PRESTATION 10 : RÉVERSIBILITÉ ET TRANSFERT DE COMPÉTENCES

4.10.1 Présentation de la prestation

Cette prestation permet de sortir une application existante ou un projet du périmètre du marché au cours de l'exécution de l'accord cadre ou/et en fin de l'accord cadre.

La capacité de réversibilité est une responsabilité permanente du titulaire dans le cadre de l'exécution de l'ensemble des prestations de l'accord cadre.

La réversibilité, ou phase de réversibilité, est établie et conduite par le titulaire de manière à :

- minimiser les risques de transition sur le futur accord cadre succédant au présent accord cadre ;
- poursuivre les prestations objet du présent accord cadre jusqu'à leur terme sans aucune interruption ni dégradation de service.

Le titulaire est responsable de la gestion et de la conduite des prestations de réversibilité. Il réalise toutes les activités nécessaires en vue de réaliser le transfert de responsabilité de l'ensemble des prestations du présent accord cadre

Cette prestation est commandée si le titulaire de l'accord-cadre est différent du titulaire attributaire du marché ou de l'accord-cadre qui sera lancé pour succéder au présent accord-cadre.

Dans ce cadre de cette prestation, le titulaire a pour obligation :

- d'exécuter les prestations convenues pendant toute la réversibilité ;
- de maintenir à jour la documentation dont il a la charge jusqu'au terme du présent accord cadre, y compris pendant la phase de réversibilité ;
- de mettre à disposition la logistique minimum, permettant au CNRS et/ou aux titulaires désignés de venir dans ses locaux si nécessaire, pour acquérir la connaissance sous réserve du respect des conditions et normes de sécurité du titulaire ;
- de présenter au CNRS et/ou aux titulaires désignés les prestations, organisations et outils en place
- de mettre à disposition du CNRS et/ou des titulaires désignés les éléments nécessaires au transfert de connaissances
- de préparer et conduire la formation théorique du CNRS et/ou des titulaires désignés
- de préparer et conduire la formation pratique du CNRS et/ou des titulaires désignés
- de préparer et conduire l'exécution des prestations objet de l'accord cadre, en binôme avec le CNRS et/ou les titulaires désignés ;

[24] La typologie des anomalies est décrite dans le glossaire au § 5.1.2.

- d'autoriser l'exécution des prestations objet de l'accord cadre par le CNRS et/ou les titulaires désignés sous son contrôle dans la mesure où les engagements de niveaux de service peuvent être respectés ;
- de restituer tous les actifs du CNRS ;
- d'assurer l'assistance post-réversibilité.

Ce transfert de compétence doit permettre à l'équipe projet CNRS et/ou à un tiers désigné par le CNRS d'acquérir la connaissance suffisante de l'application ou du projet concerné et utilisé par le CNRS ainsi que de son environnement, afin de pouvoir prendre en compte et corriger l'ensemble des anomalies applicatives qui se présentent ainsi que de répondre à toutes les demandes d'évolutions émises.

Cette équipe peut soit faire partie de la DSI, soit être un partenaire de la DSI (autre EPST…), soit être une société de service.

Une fois hors du périmètre, l'application ne pourra pas faire l'objet d'une prestation avant d'avoir été concernée par une prestation de prise en main dans le cadre du MCO.

4.10.2 Unité d'œuvre SORTIE_REV – Sortie du périmètre du marché avec réversibilité

Présentation

Cette unité d'œuvre permet de sortir une application existante ou un projet du périmètre du marché. Dans cette UO il s'agit d'une sortie bénéficiant d'un encadrement auprès de l'équipe reprenant la charge de l'application.

Eléments en entrée

- Identification de l'application concernée.
- Plan de réversibilité.

Actions à réaliser

Les actions à réaliser par le titulaire sont les suivantes :

- Fournir au CNRS au lancement de la prestation, les informations listées ci-après. Pendant toute la durée de la réversibilité le titulaire met ces informations à jour de manière hebdomadaire, sauf en cas d'accord avec le CNRS sur une fréquence différente :
    - l'environnement technique, les logiciels et outils ;
    - les processus et procédures ;
    - le référentiel de tests ;
    - la compilation des documentations (technique et fonctionnelle) complètes sur support électronique ;
    - toute information pertinente pour avoir une « photographie » fidèle du système
    - les paramétrages et codes source des applications (propriétés du CNRS) y compris l'intégralité du code de l'application non encore soumis ;
    - les volumes et statistiques ;
    - les inventaires d'actifs ;
    - les demandes en cours ainsi que leurs statuts (évolutions, développements, études, etc.) ;
    - toute autre information pertinente demandée par le CNRS nécessaire au transfert de responsabilité vers le CNRS et/ou les titulaires désignés et à la continuité des prestations.
- Composer l'équipe :
    - Décrire la composition et les coordonnées de l'équipe chargée d'effectuer la réversibilité ;
    - Désigner les chefs de projet de réversibilité identifiés : vingt jours ouvrés au minimum avant le début de la réversibilité, chacune des parties impliquées désigne un chef de projet de réversibilité suffisamment qualifié et expérimenté pour être l'unique point de contact pour l'ensemble de la prestation de réversibilité et en notifie les autres parties.
    - Dès la demande d'exécution de la prestation de réversibilité, le titulaire, le CNRS et/ou les titulaires désignés discutent et se mettent d'accord sur la liste des personnels clés du titulaire (a minima le

responsable technique et le responsable fonctionnel) nécessaires à la continuité des prestations. Sauf cas de force majeure, le titulaire s'engage à conserver ces personnels clés pendant toute la durée de la réversibilité.

- Le titulaire consulte le CNRS avant d'effectuer, pour un personnel clé, tout changement de poste ou de position, ou d'affectation sur un autre projet ou client du titulaire pendant la réversibilité.

- Organiser et planifier le transfert de compétences avec le « repreneur » de l'application ;
- Réaliser un transfert de connaissances vers le CNRS et/ou les titulaires désignés. Ce transfert inclut notamment les activités suivantes :
    - la participation aux ateliers, réunions et activités de prise en main pour lesquels la participation du titulaire est demandée par le CNRS et/ou les titulaires désignés ;
    - la fourniture au CNRS et/ou aux titulaires désignés des informations nécessaires à l'exécution du plan de réversibilité ;
    - la fourniture au CNRS et/ou aux titulaires désignés des informations et de la documentation nécessaires à la continuité des prestations (y compris les détails et les accès aux points de contact du personnel du titulaire et des tiers durant la réversibilité) ;
    - l'autorisation au CNRS et/ou aux titulaires désignés d'amener et d'utiliser des ordinateurs, des systèmes d'enregistrement ainsi que d'autres équipements et connexions sur les sites du titulaire pour faciliter le transfert de connaissance, en accord avec la politique de sécurité du titulaire portée à la connaissance du CNRS ;
    - la formation du personnel du CNRS et/ou des titulaires désignés sur l'exécution des prestations à transférer ;
    - l'autorisation au CNRS et/ou aux titulaires désignés de travailler avec les équipes du titulaire pour faciliter le transfert de connaissance du titulaire vers le CNRS et/ou les titulaires désignés (y compris le transfert de connaissance par compagnonnage et par l'exécution des prestations sous contrôle du titulaire - minimum de 3 corrections/évolutions mineures ;
    - la fourniture des accès au personnel du titulaire ayant travaillé ou travaillant actuellement sur le présent accord cadre du CNRS (même si ces ressources ne sont pas dédiées au projet) ;
    - l'explication au CNRS et/ou aux titulaires désignés de toute documentation, organisation, méthode, standard, processus et procédure pertinente.
- Décommissionner les environnements correspondant à l'application chez le titulaire
- Supprimer tous les documents et toutes les données liés à l'application chez le titulaire.

Durée de la réversibilité

- Le titulaire exécute la prestation de réversibilité jusqu'à la complétude du plan de réversibilité.
- A l'issue de la réversibilité, le titulaire fournit au CNRS et/ou aux titulaires désignés un support téléphonique ou par e-mail pendant six mois. Ce support consiste à fournir au CNRS et/ou aux titulaires désignés les réponses aux questions résiduelles relatives à la réversibilité et qui pourraient survenir à l'issue. Ce support est inclus dans le prix de la prestation de réversibilité.
- La durée de la réversibilité ne doit en aucun cas être inférieure à la durée nécessaire à un transfert effectif des prestations vers le CNRS et/ou les titulaires désignés et ce, sans impacter les activités du CNRS.

Livrables

Les livrables pour cette unité d'œuvre sont :

- Planning de réversibilité ;
- Plan de réversibilité mis à jour et entièrement exécuté ;
- Ensemble des éléments documentaires (informations fonctionnelles et techniques) sur le système d'information à jour et cohérentes avec la version en exploitation :
    - spécifications fonctionnelle et technique détaillées (cf. la description du contenu aux § 5.2.2.1 et 5.2.2.2),
    - modèle de la base de données et description détaillée des tables et des rubriques,
    - composants logiciels sources et exécutables des applicatifs,

- état des versions des progiciels, outils et utilitaires en exploitation.

- L'intégralité du code de l'application non encore soumis dans la forge du CNRS
- Mise à niveau fonctionnelle et technique de l'équipe projet CNRS et/ou de l'équipe d'un autre titulaire ;
- Documents support de la mise à niveau;
- Rapport indiquant l'ensemble des documents supprimés
- Procès-verbal de sortie du périmètre de l'application attestant
  - des actions réalisées ;
  - de la destruction des documents, des codes-sources et des données du CNRS, copies de sauvegarde comprises.

Complexité

La complexité résulte de la complexité de l'application.

Modalité de réception par le CNRS

La réception de la prestation est prononcée après :

- la réalisation du transfert de compétences ;
- la mise à niveau des équipes.

Le titulaire ne peut considérer l'exécution de la prestation de réversibilité comme achevée sans l'accord explicite du CNRS prononcé au terme de la recette de la réversibilité, par la signature du PV de recette.

5 ANNEXES

5.1 ANNEXE 1 : ABRÉVIATIONS ET GLOSSAIRE

5.1.1 Abréviations

Le tableau suivant présente la liste des abréviations utilisées dans ce document.

Sigle	Description
BPU	Bordereau de prix unitaires
CCAP	Cahier des Clauses Administratives
CCT	Cadre de Cohérence Technique
CCTP	Cahier des Clauses Techniques Particulières
CNRS	Centre National de la Recherche Scientifique
CTD	Conception technique détaillée
DAL	Dossier d'architecture logicielle
DSI	Direction des Systèmes d'Information
EPST	Etablissement public à caractère scientifique et technique
MCO	Maintien en Condition Opérationnelle
MCS	Maintien en Condition de Sécurité
MEX	Manuel d'exploitation
MINS	Manuel d'installation
MOA	Maîtrise d'ouvrage
MOE	Maîtrise d'œuvre
MUT	Manuel utilisateur
MVP	Minimum Viable Product : produit minimum viable
PAQ	Plan Assurance Qualité
PAS	Plan Assurance Sécurité
SFD	Spécifications fonctionnelles détaillées
SI	Système d'Information
STD	Spécifications techniques détaillées

5.1.2 Glossaire

Le tableau suivant présente le glossaire des termes utilisés dans ce document.

Terme	Description
Anomalie	La notion d'anomalie signifie : • tout événement qui ne fait pas partie du fonctionnement normal, provoquant ou pouvant provoquer une interruption ou une dégradation du service ou une altération de sa qualité ou de la sécurité ; • ou tout écart par rapport à un fonctionnement préalablement défini par une référence faisant autorité (documentations fonctionnelles et techniques ou exigences du CNRS).

Terme	Description
Anomalie bloquante	Anomalie pour laquelle il n'existe pas de solution de contournement possible et qui présente au moins l'une des caractéristiques suivantes : • elle endommage des données de la base, • elle rend inopérante l'utilisation d'un ou plusieurs actes de gestion ou d'une ou plusieurs procédures d'exploitation, • elle restitue à l'utilisateur des données obsolètes ou erronées, • elle concerne un incident de sécurité (événement non prévu qui affecte la disponibilité, l'intégrité ou la confidentialité de la donnée, ou qui met en péril les fonctions de traçabilité ou journalisation légale des actions utilisateurs ou administrateurs).
Anomalie majeure	Anomalie présentant les mêmes caractéristiques qu'une anomalie bloquante mais pour laquelle il existe des solutions de contournement temporaires acceptées par le CNRS
Anomalie mineure	Anomalie qui n'a pas d'impact sur le déroulement de la fonctionnalité ou bien anomalie qui ne rentre pas dans une des deux catégories précédentes
Audit de sécurité	Audit d'une application, d'une architecture, d'une infrastructure, d'un site ou d'une organisation. L'objectif est de valider la conformité au cahier des charges, à une politique, des bonnes pratiques ou de l'état de l'art
Règle de gestion	Une règle métier ou règle de gestion est une déclaration explicite qui définit ou contrôle une contrainte, une exigence ou une logique opérationnelle qui doit être respectée. Une règle de gestion doit être : • **Non-ambiguë** : la règle doit être formulée de manière claire et compréhensible, sans ambigüité. Elle doit être interprétable de la même manière par toutes les parties concernées. • **Non-redondance** : les règles ne doivent pas se contredire ni se répéter. Chaque règle doit apporter une information unique et spécifiques. • **Obligatoire** : la règle doit être contraignante. Cela signifie qu'elle doit définir une exigence ou une contrainte qui doit être respectée sans exception. • **Stabilité** : idéalement, une règle de gestion ne devrait pas changer fréquemment. Elle doit rester stable pour garantir la cohérence des opérations dans le temps.
Test de non régression	Test, dans le cas de changement de version permettant de vérifier que les modifications apportées n'ont pas entraîné d'effets de bord non prévus qui pourraient dégrader le comportement du logiciel antérieurement validé. Ils portent sur l'exécution de tests de bout en bout des différentes procédures métier au travers des fonctionnalités de l'application et déjà joués afin de s'assurer que le système répond toujours aux exigences spécifiées.
Test de charge	Test permettant de mesurer la performance d'un système en fonction de la charge d'utilisateurs simultanées. Hormis la simulation sur le nombre de connexions simultanées, un test de charge permet également de tester le temps de réponse du système, sa robustesse, ou encore de dimensionner des serveurs etc.
Test de performance	Test permettant de vérifier les performances d'une ou plusieurs fonctionnalités de l'application en charge ; nécessite un déploiement complet de l'application (serveur d'application, base de données ...).
Test de performance unitaire	Test de la performance unitaire d'un fragment de l'application, d'un composant, d'une fonctionnalité
Test de validation	Test d'une fonctionnalité de l'application qui peut impliquer plusieurs composants ; nécessite un déploiement complet de l'application (serveur d'application, base de données, ...)
Test d'intégration	Test d'un composant de l'application, c'est à dire d'un ensemble de classes ou de procédures/fonctions ; nécessite parfois la présence d'une base de données ou autre

Terme	Description
Test d'intrusion	Test d'intrusion d'une application. L'objectif est de mesurer le risque associé à un système d'information en simulant des conditions d'attaque réalistes.
Test unitaire	Test d'un fragment applicatif ; cela correspond à tester le fonctionnement d'une seule classe ou d'une procédure/fonction

5.2 ANNEXE 2 : DESCRIPTION DES PRINCIPAUX PLANS ET LIVRABLES DOCUMENTAIRES

Les éléments attendus à minima dans les principaux livrables documentaires sont décrits ci-dessous.

5.2.1 Plans de gouvernance

5.2.1.1 Plan d'Assurance Qualité

Le Plan d'Assurance Qualité (PAQ) est :

- instancié partiellement par le titulaire à la signature du présent accord-cadre et selon les instructions du CNRS
- complété et livré en totalité par le titulaire à la fin de la prestation de lancement
- mis à jour en fonction des besoins et livré par le titulaire pour présentation en comité de pilotage.

Le Plan d'Assurance Qualité porte notamment sur :

- la documentation de référence applicable ;
- l'organisation générale ;
- les structures de coordination et de suivi, processus associés ;
- les processus de gouvernance (y compris la description détaillée de la comitologie et de son mode de fonctionnement) ;
- les processus généraux et les processus opérationnels relatifs aux prestations.

Le Plan d'Assurance Qualité est par ailleurs porté par des personnels du titulaire pluridisciplinaires, qualifiés et expérimentés en conséquence (profils managériaux / experts méthodes, processus et qualité).

Les processus et procédures prévus au Plan d'Assurance Qualité doivent être appliqués par le titulaire. Il lui appartient toutefois d'alerter le CNRS dans le cas où l'un d'eux lui semble inadapté ou inefficace et de proposer en tant que professionnel les améliorations idoines.

5.2.1.2 Plan d'Assurance Sécurité

Le Plan d'Assurance Sécurité (PAS) est :

- instancié partiellement par le titulaire à la signature du présent accord-cadre et selon les instructions du CNRS
- complété et livré en totalité par le titulaire à la fin de la prestation de lancement
- mis à jour en fonction des besoins et livré trimestriellement par le titulaire pour présentation en comité de sécurité
- en cas de modifications importantes des process ou des modalités d'exécution des prestations, le plan est mis à jour immédiatement et communiqué au CNRS. Si besoin, un comité de sécurité extraordinaire est convoqué par le CNRS pour l'approuver.

Le Plan d'Assurance Sécurité décrit l'ensemble des moyens techniques, organisationnels, matériels, logiciels et humains qui doivent être mis en place dans le cadre des prestations pour conserver, rétablir, garantir la sécurité du système d'Information telle que définie dans le Plan de Sécurité des Systèmes d'Information du CNRS et tout autre document de même nature applicable.

Le Plan d'Assurance Sécurité comprend notamment les dispositions qui doivent être maintenues, appliquées et vérifiées en matière de :

- disponibilité ;
- intégrité ;
- confidentialité ;
- traçabilité ;
- authentification ;
- non répudiation et imputation ;
- etc.

Le titulaire précise également dans ce document les références de ses propres Plan d'Assurance Sécurité et de Plan de Reprise d'Activité (PRA) qui doivent pouvoir être consultés sans réserve à la demande du CNRS.

Le Plan d'Assurance Sécurité est par ailleurs porté par des personnels du titulaire qualifiés et expérimentés en conséquence (profils managériaux / experts sécurité, continuité, disponibilité).

5.2.1.3 Plan de mesure

Le Plan de mesure explicite les indicateurs, leurs modalités de calculs (formule, période retenue, outillage), leurs objectifs à atteindre et leurs modalités de suivi.

Le Plan de mesure est :

- défini et livré en totalité par le titulaire à la fin du lancement du premier projet
- mis à jour en fonction des besoins et livré par le titulaire pour présentation en comité de pilotage.

5.2.1.4 Plan de réversibilité

Le Plan de réversibilité décrit les modalités du processus de réversibilité.

Le Plan de réversibilité est :

- instancié partiellement par le titulaire à la signature du présent accord-cadre
- complété et livré en totalité par le titulaire à la fin de la prestation de lancement. Ce plan de réversibilité est un livrable de la prestation de MCO/MCS. S'il n'est pas livré aux échéances requises, la prestation ne pourra être admise
- mis à jour au minimum une fois par an par le titulaire, ou en fonction de l'évolution des prestations fournies et de leur périmètre, pour présentation en comité de pilotage
- finalisé au début de la prestation de réversibilité.

5.2.1.5 Rapport d'analyse de l'obsolescence technique et de vulnérabilités de sécurité

Le rapport d'analyse de l'obsolescence technique et de vulnérabilités de sécurité liste les composants globaux impactés et dépréciés pour chacune des applications de la prestation de MCO/MCS, en particulier il sera indiqué :

- les sous composants d'un Framework utilisés, le niveau d'obsolescence en regard du CCT,
- l'impact d'une mise à niveau sur chaque application en particulier,
- le rapport de scan de vulnérabilités (de type owasp zap), les préconisations de corrections et les charges,
- le rapport Sonar avec au minimum les problèmes marqués bloquants, critiques, majeurs, les vulnérabilités, les « security hotspot », la dette technique, le taux de couverture du code par les tests et le pourcentage de duplication de code
- défini et livré en totalité par le titulaire dès le passage en mode MCO/MCS
- mis à jour semestriellement et livré par le titulaire pour présentation en comité de pilotage. Ce rapport est un livrable de la prestation de MCO/MCS.

5.2.2 Livrables documentaires relatifs à la maintenance de la solution applicative

5.2.2.1 Spécifications fonctionnelles détaillées (SFD)

Les spécifications fonctionnelles détaillées décrivent l'ensemble des cas d'utilisation de l'application avec leurs règles de gestion ainsi qu'une description détaillée du comportement métier tant pour les éléments graphiques (écrans, …) que pour les services.

Les spécifications fonctionnelles détaillées peuvent par exemple comprendre ces éléments, s'ils sont pertinents :

- pour les éléments graphiques sont décrits :
  - le rendu visuel : les règles d'ergonomie et les éléments graphiques et visuels,
  - le comportement dynamique avec une description précise de l'ensemble des écrans avec leurs règles de gestion, les traitements associés et leur enchaînement, ainsi que les éléments ;

- pour les services sont décrits :
  - les interfaces et les échanges de données, les règles de gestion, les traitements associés et la gestion des erreurs.
- le modèle conceptuel des données ou schémas d'entités Drupal (exemple : Node, User, Taxonomy Term, Custom Entity) avec
  - Liste des types de contenu (exemple: Article, Publication) et leurs champs (ex: title, body, field_author).
  - Relations entre entités (exemple: référence entre Node:Article et Taxonomy Term:Category via Entity Reference).
  - Validations (exemple: champ field_pdf limité à 10 Mo, format PDF uniquement).
- spécification des endpoints REST/GraphQL (si API custom), des formulaires Drupal (via Form API), et des intégrations avec des modules contribués (ex: Feeds pour les imports, Rules pour les workflows),
- rôles, permissions Drupal (granularité via modules comme Permissions by Term ou Group), et workflows de modération (ex: Content Moderation).
- Configuration des types de contenu (champs, formats, validations).
- Règles de workflow (ex: transitions de states via Workflow ou State Machine).
- Intégrations avec des modules existants (exemple:
  - Views : Liste des publications avec filtres (catégorie, date).
  - Webform : Formulaire de contact avec validation côté serveur.
  - Feeds : Import automatique de données depuis un CSV/JSON.
  - Rules : Actions automatisées (ex: envoyer un email après soumission d'un formulaire).
- Personnalisations de thèmes:
  - Templates Twig (exemple: node--article.html.twig pour le rendu d'un article).
  - Libraries (CSS/JS personnalisés via libraries.yml).
- Preprocess hooks (exemple: hook_preprocess_node() pour ajouter des variables aux templates).

5.2.2.2 Spécifications techniques détaillées (STD)

Ce document permet d'avoir une vision de l'architecture applicative du projet. Il a pour objectif de faciliter les opérations de maintenance.

Les spécifications techniques détaillées peuvent par exemple comprendre ces éléments, s'ils sont pertinents :

- Architecture modulaire :
  - Liste des modules custom (avec leur rôle : ex: custom_article pour la gestion des articles).
  - Liste des modules contribués utilisés (ex: Views, Rules, Webform, JSON:API).
  - Intégration avec Symfony (services, events, dependency injection).
- Description de l'architecture modulaire:
  - Modules custom : Rôle, dépendances, hooks/services utilisés (exemple: custom_article gère les articles via hook_node_presave()).
  - Modules contribués : Version, configuration, personnalisations (exemple: Views avec des displays custom).
  - Thème : Structure (Twig, libraries), héritage (exemple: sous-thème de Olivero).
- Description de la logique métier dans des services Symfony ou des hooks Drupal, diagramme des modules et leurs dépendances (via composer.json), flux de données entre entités (ex: comment un Node de type "Article" est lié à un Taxonomy Term "Catégorie").
- Gestion des données :
  - Schéma des entités custom (via hook_entity_type_build() ou annotations).
  - Configuration des champs (types, formats, validations).

- Migration des données (via Migrate API si import depuis un ancien système).

- Sécurité :
  - Permissions (rôles, hook_permission()).
  - Protection des formulaires (CSRF, validation côté serveur).
  - Gestion des sessions (via SessionHandler Symfony ou modules comme Session Limit).
- Performance :
  - Cache (Drupal Cache API, cache_contexts, cache_tags).
  - Indexation (via Search API + Solr/Elasticsearch).
- Multilingue/Responsive/Accessibilité :
  - Configuration des langues (via Language et Content Language).
  - Thèmes adaptatifs (ex: Olivero pour l'accessibilité, Bootstrap pour le responsive).
  - Vérification WCAG (outils comme Pa11y ou modules Accessibility).
- Description des flux de données
  - API REST : Module JSON:API ou RESTful Web Services (endpoints, méthodes autorisées).
  - Imports/Exports : Module Feeds (pour les imports CSV/JSON) ou Migrate (pour les migrations depuis un ancien système).
  - Webservices externes : Intégration via Guzzle (HTTP client) ou modules comme SOAP
- Description détaillée de l'implémentation technique de l'authentification et des sessions,
  - SSO : Intégration avec Shibboleth (module SimpleSAMLphp) ou OAuth2 (module OAuth2).
  - Gestion des sessions : Configuration de session.handler (ex: database ou redis), module Session Limit pour limiter les sessions simultanées.
  - Sécurité : CSRF (protection native Drupal), Password Policy (pour les mots de passe complexes).
- Configuration Drupal :
  - Fichiers settings.php/ settings.local.php et services.yml (variables d'environnement, services custom).
  - Configuration Management (drush cex/drush cim) pour exporter/importer la configuration entre environnements.
  - Variables spécifiques (ex: clés API, URLs de services externes) stockées dans settings.php/ settings.local.php ou via le module Config Ignore pour les exclure des exports.
- Mises en place pour la qualité de code et le paramétrage
  - Coding Standards : Respect des Drupal Coding Standards (outils : PHP_CodeSniffer + drupal/coder).
  - Tests :
    - PHPUnit (tests unitaires pour les services Symfony).
    - Behat (tests fonctionnels avec DrupalDriver).
    - Cypress (tests E2E pour l'UI).
  - Analyse statique : PHPStan ou Psalm pour détecter les erreurs de typage.
- Gestion des statistiques de fréquentations
  - Matomo pour le tracking.
  - Logs personnalisés : Utilisation de Watchdog (\Drupal::logger()) ou intégration avec un outil externe (exemple: ELK Stack).
- Conformité RGPD
  - Module GDPR : Gestion des consentements, anonymisation des données.
  - Droit à l'oubli : Implémentation via hook_entity_delete() pour supprimer les données personnelles.

- Cookies : Module Cookie Consent pour la gestion des cookies.

- Outils transverses (antivirus sur serveur via le “webservice Antivirus” du CNRS, GED…).

5.2.2.3 Conception technique détaillée (CTD)

La conception technique détaillée peut par exemple comprendre ces éléments, s’ils sont pertinents :

- la description de l’architecture technique et applicative,
- Diagramme des entités (par exemple: Node, User, Custom Entity) et leurs relations.
- Diagramme des modules et leurs interactions (par exemple : comment custom_article utilise Views pour afficher une liste).
- Description des contraintes métiers et techniques :
    - Contraintes sur les entités :
        - Validations des champs (ex: field_pdf limité à 10 Mo, format PDF uniquement).
        - Règles de cardinalité (ex: un Article doit avoir un et un seul Auteur).
        - Hooks de validation (ex: hook_entity_presave() pour vérifier que tous les champs obligatoires sont remplis).
    - Index et performances :
        - Index automatiques gérés par Drupal (via Entity API et Database API).
        - Index custom (si nécessaire) : Définis via hook_schema() ou dans le fichier .install du module.
        - Cache : Utilisation de cache_tags et cache_contexts pour optimiser les requêtes.
    - Référentiels :
        - Taxonomy : Pour les référentiels hiérarchiques (ex: Catégories → Sous-catégories).
        - Custom Entities : Pour les référentiels complexes (ex: Laboratoire avec des champs spécifiques).
- Modules contribués : Options pour les listes statiques (ex: Liste des statuts : Brouillon, Validé, Publié).
- Dictionnaire des entités et champs :
    - Liste des entités :
        - Entités natives : Node, User, Taxonomy Term, Media.
        - Entités custom : CustomEntityX (définies via hook_entity_type_build() ou annotations).
    - Description des champs :
        - Pour chaque entité, lister ses champs avec :
            - Nom machine (ex: field_author).
            - Type de champ (ex: entity_reference pour une référence à User).
            - Cardinalité (ex: 1 pour un auteur unique, -1 pour plusieurs).
            - Validations (ex: required, max_length: 255).
            - Format de stockage (ex: text, text_long, file).
    - Configurations associées :
        - Fichiers de configuration YAML (par exemple: config/install/field.storage.node.field_author.yml).
        - Exportable via drush cex.
- Architecture des modules custom :
    - Services Symfony (fichiers *.services.yml).
    - Hooks Drupal utilisés (ex: hook_node_presave(), hook_form_alter()).

- Plugins (ex: BlockPlugin, FieldFormatter, ActionPlugin).
- Events Symfony (ex: KernelEvents::REQUEST, HookEventDispatcher).

- Intégrations avec le SI CNRS :
  - Authentification : SSO via Shibboleth ou OAuth2 (module SimpleSAMLphp).
  - Flux de données : API REST/GraphQL (modules JSON:API, GraphQL), ou imports via Feeds/Migrate.
- Tests :
  - PHPUnit (pour les services Symfony).
  - Behat (pour les tests fonctionnels).
  - Cypress ou Selenium (pour les tests E2E).
- Gestion des configurations :
  - Fichiers YAML dans /config/install ou /config/optional.
  - Export/Import : Utilisation de drush cex/drush cim pour synchroniser les configurations entre environnements.
  - Variables d'environnement : Stockage des clés sensibles (ex: clés API) dans settings.php ou via le module Config Ignore.

5.2.3 Livrables documentaires relatifs à l'exploitation de la solution applicative

5.2.3.1 Dossier d'architecture logicielle (DAL)

Ce dossier décrit de manière exhaustive les utilisateurs, les interactions avec la solution, et les caractéristiques techniques particulières. Il détaille notamment l'architecture logicielle

- Description de l'empilement Drupal + Symfony (par exemple : "Drupal 10 + Symfony 6 + PHP 8.2"),
- Modules clés : Liste des modules contribués/custom et leur rôle.

Mais également son intégration dans le SI du CNRS :

- Flux ou services web exposés ou exploités

Il doit également détailler les éléments de dimensionnement (nombre d'utilisateurs, capacité des serveurs, volumétrie des données).

Un point particulier est apporté sur l'aspect sécurité par rapport aux droits des utilisateurs, l'authentification, et les mesures de sécurité (chiffrement, protection réseau, gestion des traces, …).

5.2.3.2 Manuel d'exploitation (MEX)

Ce manuel vise à donner toutes les informations nécessaires au responsable de l'exploitation pour administrer d'un point de vue technique la nouvelle application. Il décrit l'ensemble des processus permettant de gérer la nouvelle application, notamment, les procédures d'arrêt / redémarrage des différents services composant l'application, les informations nécessaires à la définition des procédures de sauvegarde / restauration, les procédures de gestion et de rotation des logs, la gestion des batch et des flux inter-applicatifs …

Le document doit être conçu de façon à ce qu'un exploitant qui n'a pas participé au déploiement initial et qui n'a pas de compétence Drupal soit capable d'assurer ce travail (notamment les commandes drush pour gérer les scripts custom, pour vider les caches, la configuration, les crons…

Le Manuel d'exploitation est :

- défini et livré en totalité par le titulaire à la fin de la phase de transition
- mis à jour en fonction des besoins et livré par le titulaire pour présentation en comité de pilotage.

5.2.3.3 Manuel d'installation (MINS)

On distingue :

- **Le manuel d'installation complet de l'application**

Ce manuel permet au CNRS d'être totalement autonome vis-à-vis du titulaire pour (ré)installer la solution en environnements non productif et productif. Ce document détaille l'intégralité de la procédure d'installation pour tous les éléments constitutifs de l'application ainsi que les actions et paramétrages à réaliser pour mettre en production.

Le document doit être conçu de façon à ce qu'un exploitant qui n'a pas participé au déploiement initial et qui n'a pas de compétence Drupal soit capable d'assurer ce travail. Chaque commande à dérouler est à détailler précisément (Composer, Drush etc…).

Le manuel d'installation sera mis à jour lors d'une évolution majeure de configuration et livré par le titulaire pour présentation en comité de projet.

- **Le manuel d'installation d'une version**

Dans ce document, les évolutions seront tracées d'une version à l'autre, en conservant l'historique des versions. Le manuel d'installation doit être identifié avec le même numéro de version que l'application livrée.

La procédure d'installation doit être identique quel que soit l'environnement cible (recette, production, …).
Ce document peut servir de base pour automatiser les déploiements avec Ansible.