

# Cadre de Référence de la méthodologie transverse de la démarche de tests au MJ

Date d'application : 15/04/2020

Version : V2.0

---

**Retrouvez-nous sur :**  
[justice.gouv.fr](http://justice.gouv.fr)

## Circuit de validation

<b>Date application</b>	<b>Version</b>	<b>Objet</b>	<b>Rédaction</b>	<b>Vérification</b>	<b>Approbation</b>
15/04/2020	V2.0	Prise en compte des impacts de l'agilité sur la démarche	Le 15/04/2020 P. RONCOLI	Le 03/04/2020 S. MAUREY C. ROCHEDEREUX	Le 15/04/2020 O. MARTEL

## Diffusion

<b>Pour action</b>	SNUM
<b>Pour information</b>	

## Historique des modifications

<b>Date application</b>	<b>Version</b>	<b>Objet</b>	<b>Rédaction</b>	<b>Vérification</b>	<b>Approbation</b>
14/08/2018	1.0VP05	Intégration fiches de relecture PMO et JVT	FBU		
12/09/2018	1.0VP06	Intégration fiches de relecture JVT P05 et retours atelier revue du 4 sept. 2018	FBU		
03/10/2018	1.0VP07	Pour validation de la première version	FBU		
04/10/2018	1.0VP08	Modification	DFO		
15/11/2018	1.0VP09	Prise en compte remarque COPIL 3, pdf annoté et fiche retour VP08 et gestion des exigences	JEW		
16/11/2018	1.0	Prise en compte des remarques SMA Définition VPA Définition VA	FBU		
04/02/2019	V1.1	Prise en compte des remarques sur le COSUIT	JEW		
31/01/2020	V1.2	Prise en compte SNUM	P. RONCOLI		
19/03/2020	V2.0 VP0.1	Prise en compte des impacts de l'agilité sur la démarche	P. RONCOLI		
27/03/2020	V2.0 VP0.2	Remarques de C. ROCEDEREUX (hors chapitre « activité »)	P. RONCOLI		
30/03/2020	V2.0 VP0.3	Prise en compte agilité pour le chapitre sur les activités	P. RONCOLI		
15/04/2020	V2.0	Prise en compte des remarques de S. MAUREY	P. RONCOLI		

# Sommaire

1	Objet du document .....	3
1.1	Domaine d'application .....	3
1.2	Définition.....	3
2	Le Processus de Test au Ministère de la Justice .....	4
2.1	Processus de test.....	4
2.1.1	Niveaux de test.....	5
2.1.2	Activités.....	6
2.2	Insertion des niveaux et activités de tests dans le Cycle de vie des projets.....	7
2.2.1	Cycle en V.....	7
2.2.2	Agile .....	7
2.3	Détail des niveaux de tests .....	8
2.3.1	Test de Développement .....	8
2.3.2	Test de Vérification.....	10
2.3.3	Test de validation .....	13
2.4	Activités de test.....	17
2.4.1	Activité Organiser.....	17
2.4.2	Activité Analyser les TS.....	18
2.4.3	Activité Analyser les TIA.....	19
2.4.4	Activité Analyser les TAO .....	20
2.4.5	Activité Analyser les RFU .....	21
2.4.6	Activité Spécifier .....	22
2.4.7	Activité Implémenter .....	23
2.4.8	Activité Exécuter et rendre compte.....	24
2.4.9	Activité Conclure.....	25
2.4.10	Rapport de Test (RT).....	26
3	La stratégie de test.....	27
3.1	Introduction .....	27
3.2	Démarche de construction.....	28
3.2.1	Cadrage - Analyser .....	28
3.2.2	Cadrage - Périmètre et Moyens des tests .....	28
3.2.3	Cadrage - Organisation .....	29
3.3	La dynamique de construction pour un projet Cycle en V .....	29
3.3.1	Stratégie de tests V1 .....	30
3.3.2	Stratégie de tests V2 .....	30
3.4	La dynamique de construction pour un projet en mode Agile .....	31
3.5	Focus sur l'effort de Test .....	31
4	Les Moyens .....	33
4.1	Les environnements de test.....	33
4.1.1	Environnement de Développement .....	34
4.1.2	Environnement de Recette Usine.....	34
4.1.3	Environnement de Recette Fonctionnelle Utilisateur .....	35
4.1.4	Environnement de tests techniques.....	35
4.1.5	Environnement de Pré-production .....	35
4.1.6	Environnement de Performance (optionnel) .....	36
4.2	Les outils standards utilisées dans le cadre des tests MJ.....	36
5	La gouvernance .....	37
6	Annexes.....	38
6.1	Rôles/Profils : Abréviations utilisées dans les RACI .....	38

# 1 Objet du document

## 1.1 Domaine d'application

Ce document est le cadre de référence de la méthodologie transverse de la démarche de tests telle qu'elle doit se pratiquer au Ministère de la Justice.

Pour que sa mise en œuvre soit la plus efficace et pertinente possible, il a fallu trouver le juste équilibre entre mise en œuvre de théories éprouvées (TMMI/ISTQB) et capitalisation de retours d'expérience et de bonnes pratiques.

La maîtrise de l'activité de Test est un levier permettant de garantir le niveau de qualité convenu conformément aux exigences dans des délais et des coûts contrôlés.

Les points clés :

- Anticiper la préparation,
- Organiser au mieux le processus,
- Optimiser les efforts de test,
- Assurer la couverture des tests en regard des besoins exprimés.

Cela se traduit par la mise en œuvre d'une Stratégie de Test Globale, d'une organisation client-fournisseur optimisée, d'outils et d'indicateurs utiles.

Ce cadre de référence s'adresse à tous les contributeurs d'un projet d'évolution du Système d'Information.

Les bonnes pratiques de test décrites dans ce cadre de référence s'intègrent à la méthodologie projet en vigueur.

Ce document décrit quoi, comment et par qui doivent être appliquées les bonnes pratiques, notamment sur :

- La définition des niveaux et typologies de test,
- La définition des activités de test,
- Les rôles et responsabilités de chacun,
- Les livrables à fournir.

## 1.2 Définition

Dans tout le document, les RACI sont définis comme suit :

**Rappel de la signification de RACI :**

- **R : Réalisateur**, celui qui réalise principalement l'activité et est garant de sa qualité ; Un seul réalisateur par activité.
- **A : Autorité**, celui qui supervise et qui réfère à la hiérarchie ; Une seule autorité par activité.
- **C : Contributeur**, celui qui conseille le réalisateur, mais aussi qui contribue à la réalisation de l'activité.
- **I : Informé**, celui qui doit être informé.

## 2 Le Processus de Test au Ministère de la Justice

### 2.1 Processus de test

Le processus de test est un ensemble de niveaux, d'activités et de moyens qui doivent être mis en œuvre pour vérifier et valider une solution développée par rapport aux besoins exprimés par le métier et aux contraintes techniques de production.

Il doit permettre de détecter toutes les anomalies, le plus tôt possible dans le déroulement du processus projet.

Il couvre à la fois les aspects de développement, de vérification et de validation.

On appelle test de **développement** tous les tests liés au développement, et qui ont pour objectifs de :

- Contrôler de manière unitaire les composants développés
- Contrôler l'assemblage des composants entre eux

On appelle **vérification** tous les tests qui permettent de confirmer que la solution est construite correctement, à savoir conforme à ce qui a été spécifié.

La vérification a pour objectif de vérifier que la solution :

- Est conforme à ses spécifications fonctionnelles détaillées ou aux User Stories et Enablers enrichis et aux documents descriptifs de la solution
- S'intègre bien avec les autres applications du SI
- C'est en général à partir des niveaux de vérification que les équipes fonctionnent en mode gestion du changement : anomalies, évolutions, livraisons contrôlées...

On appelle **validation** tous les tests qui permettent de confirmer que le produit est le bon, à savoir qu'il répond aux besoins exprimés.

La validation a pour objectif de valider que le produit :

- Est conforme aux besoins exprimés ou spécifications fonctionnelles générales ou aux EPICS, processus métiers
- S'intègre de bout en bout dans le processus métier et dans l'environnement de travail de l'utilisateur final,
- S'intègre correctement dans la production informatique

La démarche de tests du Ministère de la Justice, s'appuie sur 6 niveaux de tests :

- TU : Test unitaire
- TA : Test d'Assemblage
- TS : Test Système
- TIA : Test d'Intégration Applicative
- RFU : Recette Fonctionnelle Utilisateur
- TAO : Tests d'Acceptation Opérationnel

L'instanciation de la démarche de tests sur les projets est basée sur 6 activités permettant de construire, réaliser et suivre chaque niveau de tests.

### 2.1.1 Niveaux de test

Le niveau de test est une étape du processus de test :

- Avec un objectif précis et différent par niveau de test
- Avec différentes typologies de test

Il adresse le périmètre de responsabilités des tests.

Le respect de l'enchaînement des niveaux de test (suivant le mode de gestion du projet : Cycle en V ou Agile), de leurs objectifs et des conditions de passage d'un niveau à un autre permet d'assurer la qualité du produit livré. Certains niveaux de test font l'objet d'une synthèse spécifiée dans le Rapport de Test.

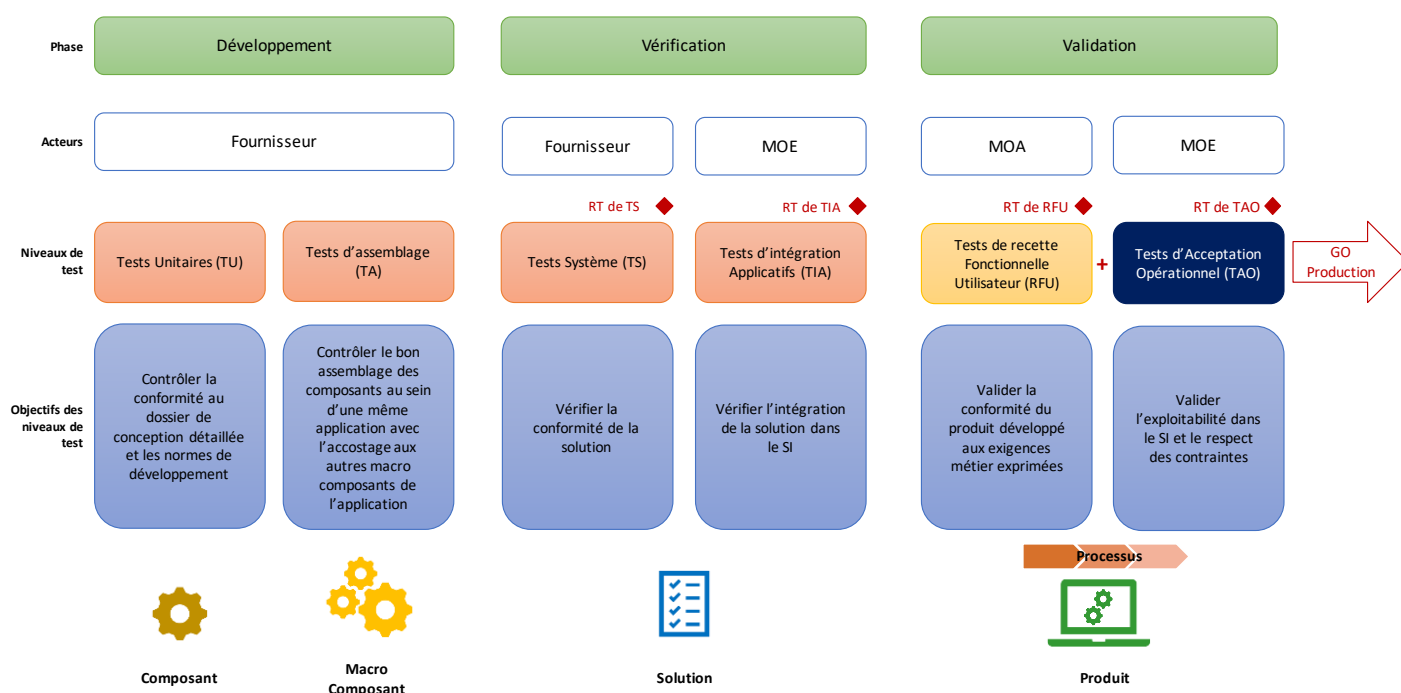
Ce paragraphe décrit les niveaux de tests, et pour chacun d'eux on définira :

- Les objectifs du niveau de test,
- La phase projet dans laquelle il s'insère,
- Le responsable du niveau de test,
- Les entrants,
- Les sortants,
- La typologie de test et l'acteur MJ associé,
- Le RACI par niveau de tests.

Dans la suite du document, nous distinguons :

- La **Solution** : ensemble des éléments informatiques (techniques et fonctionnels) qui répondent au besoin exprimé par les Métiers
- Le **Produit** : le produit est la solution mise en condition opérationnelle de fonctionnement Métier et de Production

Le schéma présente manière synthétique chaque niveau de tests :



### 2.1.2 Activités

L'organisation du processus de tests en activités permet d'optimiser les tests en se fondant sur les principes suivants :

- Anticiper les préoccupations de test, (stratégie, organisation, estimation des charges, planification, analyse) dès la phase de Spécification Fonctionnelle,
- Répartir les activités d'exécution des différents Niveaux de test dans la démarche projet (cycle en V ou Agile),
- Capitaliser les éléments réutilisables dans les Référentiels de test par domaine (MOE) / processus (MOA et PROD),
- Maîtriser la qualité du produit ainsi que les coûts et les délais des tests.

Les activités de tests sont au nombre de 6 :

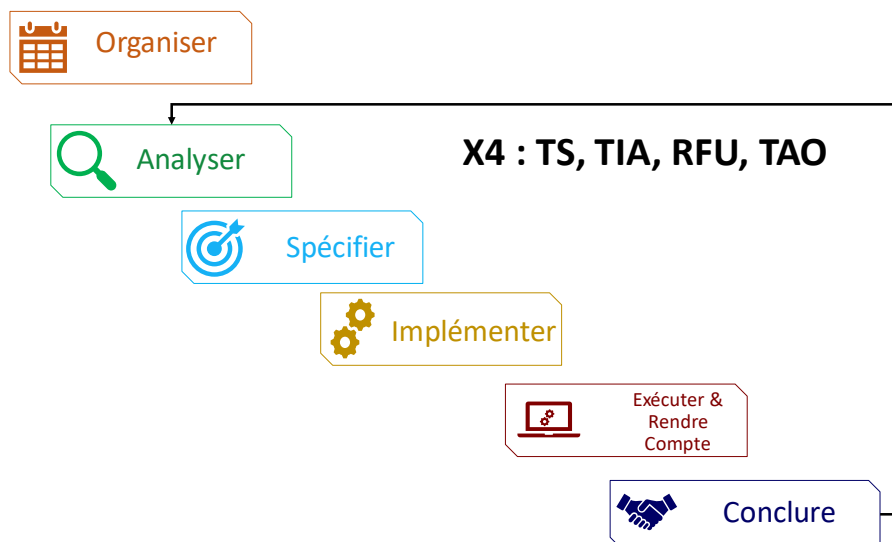
- Organiser,
- Analyser,
- Spécifier,
- Implémenter,
- Exécuter,
- Conclure.

Le pilotage des tests est assuré dans le cadre du pilotage du projet. En conséquence, le « sous-projet » de test s'intègre dans le plan projet. Tout au long du cycle projet, les activités de test sont effectuées par niveau de test par différents acteurs participants au projet.

Très tôt dans le cycle projet, les objectifs de tests du projet doivent être partagés entre les acteurs et une collaboration étroite doit être instituée. C'est l'objet de la stratégie de test abordée dans le chapitre 3.

L'enchaînement des activités est représenté par le schéma ci-dessous,

L'activité Organiser est faite pour tous les niveaux de tests. L'enchaînement des activités de « Analyser à Conclure » doit être répété pour chacun des niveaux de test, voir à chaque incrément en mode agile.







## 2.3 Détail des niveaux de tests

Dans ce chapitre les profils identifiés dans les RACI ont un code couleur, permettant d'identifier l'acteur suivant le mode projet :

- Profil : écrit en noir ; Profil commun au mode de gestion de projet cycle en V et Agile
- **Profil** : écrit en vert ; Profil particulier pour un projet en mode Cycle en V
- **Profil** : écrit en orange ; Profil particulier pour un projet en mode Agile

### 2.3.1 Test de Développement

Les tests de Développement comme leur nom l'indique correspondent aux tests réalisés par les développeurs des composants et macro composants qu'ils programment.

En général, ils ne font pas l'objet de gestion du changement (ouverture d'anomalies ou demande d'évolution). Cependant, leur contenu peut être décrit et des traces fournies par les outils de développement ou des outils tiers comme un outil d'analyse de code.

## Niveau Test Unitaire (TU)

Niveau : Test Unitaire		Responsable : Fournisseur		
Objectifs :				
<ul style="list-style-type: none"><li>• Contrôler la conformité du code (normes de sécurité et standards de développement)</li><li>• Contrôler les règles de saisie (syntaxe)</li><li>• Contrôler les règles de gestion</li><li>• Contrôler la couverture des conditions et des chemins de traitement</li><li>• Réaliser des tests de performance unitaire (selon typologie de développement)</li><li>• Réaliser une revue de code par les pairs</li></ul>				
Entrants				
<u>Projet cycle en V :</u> <ul style="list-style-type: none"><li>• Normes de développement</li><li>• Dossier d'Architecture Technique</li><li>• SFD</li></ul>		<u>Projet Agile :</u> <ul style="list-style-type: none"><li>• Normes de développement</li><li>• Dossier d'Architecture Technique</li><li>• User Story / Enabler</li></ul>		
Typologie de test	R	A	C	I
Test de composant	Fournisseur	Fournisseur		
Revue de code par les pairs	Fournisseur.	Fournisseur		
Performance Unitaire	Fournisseur	Fournisseur		
Sortants	R	A	C	I
Compte rendu issu des outils d'analyse de code utilisés et éventuellement des revues de pair formalisées	Fournisseur	Fournisseur	Fournisseur	CPMoe, RTE
Compte rendu de test de performances	Fournisseur	Fournisseur.	Fournisseur	CPMoe, RTE
Compte rendu de test unitaire	Fournisseur	Fournisseur	Fournisseur	CPMoe, RTE

## Niveau Test d'Assemblage (TA)

<b>Niveau</b> : Test d'Assemblage		<b>Responsable</b> : Fournisseur		
Objectifs :				
<ul style="list-style-type: none"><li>• Contrôler l'assemblage des composants deux à deux</li><li>• Contrôler l'accostage du macro-composant avec les macro-composants connexes</li><li>• Réaliser des tests de performance unitaire du macro-composants (selon typologie de développement)</li><li>• Réaliser une revue de code par les pairs</li></ul>				
Entrants				
<b>Projet cycle en V :</b> <ul style="list-style-type: none"><li>• Normes de développement</li><li>• Dossier d'Architecture Technique</li><li>• SFD</li></ul>		<b>Projet Agile :</b> <ul style="list-style-type: none"><li>• Normes de développement</li><li>• Dossier d'Architecture Technique</li><li>• User Story / Enabler</li></ul>		
<b>Typologie de test</b>	<b>R</b>	<b>A</b>	<b>C</b>	<b>I</b>
Revue de code automatique	Fournisseur	Fournisseur		
Performance unitaire	Fournisseur	Fournisseur		
Assemblage composant / macro-composant 2 à 2	Fournisseur	Fournisseur		
Revue de code par les pairs	Fournisseur	Fournisseur		
<b>Sortants</b>	<b>R</b>	<b>A</b>	<b>C</b>	<b>I</b>
Compte rendu issu des outils d'analyse de code utilisés	Fournisseur	Fournisseur	Fournisseur	CPMoe, RTE
Compte rendu de test de performances	Fournisseur	Fournisseur	Fournisseur	CPMoe, RTE
Compte rendu de test d'assemblage 2 à 2	Fournisseur	Fournisseur	Fournisseur	CPMoe, RTE

## 2.3.2 Test de Vérification

Les tests de vérification ont pour objectif de vérifier que la solution est conforme aux spécifications fonctionnelles détaillées et autres documents décrivant la solution.

On répond donc à la question « Réalisons nous la solution correctement ? »

Nous dénombrons 2 niveaux de tests contribuant à cette vérification :

- Les Tests Systèmes,
- Les Tests d'Intégration Applicatif.

### Niveau Test Système (TS) – Projet développé interne MJ

Niveau : Test Système (développement interne MJ)		Responsable : CPMOE / RTE		
Objectifs :				
<ul style="list-style-type: none"><li>• Vérifier que la solution est conforme aux spécifications techniques et fonctionnelles</li><li>• Vérifier que la documentation technique est conforme</li><li>• Prendre en compte la performance</li></ul>				
Entrants				
<b>Projet cycle en V :</b> <ul style="list-style-type: none"><li>• Stratégie V2 (finalisé)</li><li>• Manuel d'installation</li><li>• Protocole Packaging et livraison applicatifs</li><li>• CCT</li><li>• SFD (IHM, Batch)</li></ul>		<b>Projet Agile :</b> <ul style="list-style-type: none"><li>• Stratégie de l'incrément (finalisée)</li><li>• Manuel d'installation</li><li>• Protocole Packaging et livraison applicatifs</li><li>• CCT</li><li>• Users Story / Enabler</li></ul>		
Typologie de test	R	A	C	I
Test fonctionnel des règles de gestion	Fournisseur	CPT	BA	
Sortants	R	A	C	I
Cahier de test exécuté	Fournisseur	CPT		CPMoe, PO
Référentiel de test MJ mise à jour	Fournisseur	CPT		CPMoe, PO
Anomalies résiduelles	CPT	CPMoe, RTE	Fournisseur / BA	PO, LSI
Rapport de Test (RT) de TS	CPT	CPMoe, RTE	Fournisseur / BA	PM, PO, LSI

## Niveau Test Système (TS) – Projets développés externe MJ

Niveau : Test Système (développement externe MJ)		Responsable : CPMOE / RTE		
Objectifs :				
<ul style="list-style-type: none"><li>• Vérifier que la solution est conforme aux spécifications techniques et fonctionnelles</li><li>• Vérifier la construction du livrable et la documentation associée</li><li>• Vérifier que le packaging et l'installation sont conformes aux attendus</li><li>• Prendre en compte la performance</li><li>• Vérifier que la documentation technique est conforme</li></ul>				
Entrants				
<b>Projet cycle en V :</b> <ul style="list-style-type: none"><li>• Stratégie de tests (finalisé)</li><li>• Manuel d'installation</li><li>• Protocole Packaging et livraison applicatifs</li><li>• CCT</li><li>• SFD (IHM, Batch)</li><li>• Scénarios MOA pour une potentielle VPA</li></ul>		<b>Projet Agile :</b> <ul style="list-style-type: none"><li>• Stratégie de l'incrément (finalisée)</li><li>• Manuel d'installation</li><li>• Protocole Packaging et livraison applicatifs</li><li>• CCT</li><li>• Users Story / Enabler</li><li>• Scénarios MOA pour une potentielle VPA</li></ul>		
Typologie de test	R	A	C	I
Test fonctionnel des règles de gestion	Fournisseur	CPT	BA	
Test technique	Fournisseur	CPT		LSI
Test Installation & Packaging	Fournisseur	CPT	INT	LSI
Sortants	R	A	C	I
Cahier de test exécuté	Fournisseur	CPT		CPMoe, PO
Référentiel de test MJ mise à jour	Fournisseur	CPT		CPMoe, PO
Anomalies résiduelles	RPrest / CPT	CPMoe, RTE	Fournisseur / BA	PO, LSI
Rapport de Test (RT) de TS	RPrest / CPT	CPMoe, RTE	Fournisseur / BA	PM, PO, LSI

## Condition d'Acceptation Livraison (VPA)

La VPA se déroule deux étapes, une partie technique et une partie fonctionnelle.

Elle est précédée de la fourniture de prérequis déclencheurs de la VPA.

### 1 – Les prérequis

Pour permettre au MJ de préparer la recette usine, le fournisseur livre au préalable (au plus tard trois jours avant la date) au MJ les éléments qui seront nécessaires lors de cette VPA, dont :

- DAT,
- Manuel d'Installation,
- Guide d'Intégration,
- Plans de tests et les résultats associés y compris les tirs de performances
- Package de l'application,
- Tout autre document qui peut être utile.

La non-réception de ces éléments, dans les délais, entraîne le report de VPA.

### 2 - La partie technique

Elle consiste à vérifier le bon fonctionnement de la procédure d'installation dans l'environnement du Fournisseur. Elle est assurée par un intégrateur. La présence de l'Architecte est souhaitée mais pas obligatoire.

Elle peut être exigée :

- Avant une première livraison
- Après chaque changement significatif d'architecture
- Ou sur demande du MJ, suite à la dégradation de la qualité du livrable.

Lorsque le fournisseur indique qu'il est prêt pour livrer, la date de recette usine est convenue d'un commun accord.

La recette usine consiste à dérouler la procédure d'installation dans l'environnement du fournisseur.

L'objectif de cette phase est de qualifier l'implémentation de l'application en tenant compte de :

- La conformité de l'implémentation par rapport aux exigences du DAT. En d'autres termes l'environnement utilisé pour la recette usine doit être une instanciation du DAT
- La conformité des procédures d'installation et de la documentation associée avec les normes, et exigences du MJ.
- La mise en œuvre des bonnes pratiques.
- L'adéquation de l'implémentation aux contraintes du projet et notamment par rapport à la version déployée actuellement en production.

Elle est aussi l'occasion de vérifier la bonne gestion des sources de l'application et la construction de manière automatique du livrable

### 3 - La partie fonctionnelle

Elle consiste à vérifier le bon fonctionnement de l'application au travers d'un échantillon représentatif de scénarios fonctionnels fournis par la MOA.

## Niveau Test Intégration Applicative (TIA)

Niveau : Test Intégration Applicative		Responsable : CPMOE / RTE		
Objectifs :				
<ul style="list-style-type: none"><li>Vérifier l'intégration des applications de bout en bout</li><li>Vérifier les règles d'interface</li></ul>				
Entrants				
Projet cycle en V :		Projet Agile :		
<ul style="list-style-type: none"><li>Stratégie de tests (finalisée)</li><li>Rapport de la VPA</li><li>DAF</li><li>SFG</li><li>SFD (IHM, Batch)</li><li>Processus métier</li><li>Contrats d'interfaces</li></ul>		<ul style="list-style-type: none"><li>Stratégie de tests de l'incrément (finalisée)</li><li>Rapport de la VPA (si VPA effectuée)</li><li>User Stories</li><li>Enablers</li><li>Processus métier</li><li>Contrats d'interfaces</li></ul>		
Typologie de test	R	A	C	I
Test de règle d'interface	Fournisseur / Test	CPT	Archi	LSI, SCM
Test d'intégration de bout en bout	Test	CPT	Archi	LSI, SCM
Sortants	R	A	C	I
Cahier de test exécuté	Fournisseur / Test	CPT		CPMoe, PO
Référentiel de test mise à jour	Fournisseur / Test	CPT		CPMoe, PO
Anomalies résiduelles	CPT	CPMoe, RTE	Fournisseur / Test	PO, LSI, SCM
Rapport de Test (RT) de TIA	CPT	CPMoe, RTE	Fournisseur / Test	PM, PO, LSI, SCM

### 2.3.3 Test de validation

Les tests de validation ont pour objectif de valider que le produit (évolution ou projet) répond aux besoins du métier et aux exigences de la production MJ.

On répond donc à la question : « Faisons-nous le bon produit ? »

Nous dénombrons 2 niveaux de tests contribuant à cette validation :

- La Recette Fonctionnelle Utilisateur,
- Les Tests d'Acceptation Opérationnels.

## Niveau Recette Fonctionnel Utilisateur (RFU)

Niveau : Recette Fonctionnelle Utilisateur		Responsable : CPMOA / PM		
Objectifs :				
<ul style="list-style-type: none"><li>Valider que le produit est conforme aux besoins décrits dans les spécifications générales</li><li>Valider la bonne intégration du produit dans les processus métier (bout en bout)</li></ul>				
Entrants				
Projet cycle en V :		Projet Agile :		
<ul style="list-style-type: none"><li>Stratégie de tests (finalisée)</li><li>Rapport de la VPA</li><li>Rapport de tests niveau TS et niveau TIA</li><li>Jeu de données métier</li><li>Processus métier</li><li>Manuel utilisateur</li><li>SFG</li></ul>		<ul style="list-style-type: none"><li>Stratégie de tests de l'incrément (finalisée)</li><li>Rapport de la VPA (si VPA effectuée)</li><li>Rapport de tests niveau TS et niveau TIA</li><li>Jeu de données métier</li><li>Processus métier</li><li>Manuel utilisateur</li><li>User Story / Feature / EPIC</li></ul>		
Typologie de test	R	A	C	I
Test fonctionnel de bout en bout	MOA	CPMOa, PM	CPT /BA / Archi	CPMoe, RTE, LSI, SCM
Test Fonctionnel des règles métiers (cycle en V) ou critères d'acceptation (Agile)	MOA	CPMOa, PO	CPT / BA	CPMoe, RTE, LSI, SCM
Sortants	R	A	C	I
Cahier de test exécuté	MOA	CPMOa, PM		CPT
Référentiel de test mise à jour	MOA	CPMOa, PM		CPT
Anomalies résiduelles	CPT	CPMOa, PM	MOA /BA / Archi	CPMoe, RTE, LSI, SCM
Rapport de Test (RT) de RFU	CPT	CPMOa, PM	MOA /BA / Archi	CPMoe, RTE, LSI, SCM

## Niveau Test d'Acceptation Opérationnel (TAO)

<b>Niveau</b> : Test Acceptation Opérationnel	<b>Responsables</b> : <ul style="list-style-type: none"><li>CPMOE / RTE</li><li>CPMOA / PM</li></ul>			
Objectifs : <ul style="list-style-type: none"><li>Valider les exigences techniques</li><li>Valider l'installation de la solution</li><li>Valider l'exploitabilité de la solution</li><li>Valider la performance attendue</li><li>Valider les exigences de sécurité</li></ul>				
Entrants				
<b>Projet cycle en V</b> : <ul style="list-style-type: none"><li>Stratégie de test (finalisée)</li><li>Rapport de la VPA</li><li>DAT</li><li>CDC performance</li><li>Dossier d'homologation de sécurité</li><li>Guide d'intégration</li></ul>	<b>Projet Agile</b> : <ul style="list-style-type: none"><li>Stratégie de tests de l'incrément (finalisée)</li><li>Rapport de la VPA (si VPA effectuée)</li><li>DAT</li><li>CDC performance</li><li>Dossier d'homologation de sécurité</li><li>Guide d'intégration</li></ul>			
<b>Typologie de test</b>	<b>R</b>	<b>A</b>	<b>C</b>	<b>I</b>
Test de performance	Perf	CPT		CPMoe, RTE, LSI, SCM
Test technique	Test	CPT		CPMoe, RTE, LSI, SCM
Test d'installation et de packaging	INT	CPT		CPMoe, RTE, LSI, SCM
Test d'exploitabilité	INT	CPT		CPMoe, RTE, LSI, SCM
Test de sécurité	Sous-traitance	CPMOa, PM	Archi / RSSI	CPMoe, RTE, LSI, SCM
<b>Sortants</b>	<b>R</b>	<b>A</b>	<b>C</b>	<b>I</b>
Cahier de test exécuté	INT	CPT		
Référentiel de test mise à jour	INT	CPT		
Anomalies résiduelles	CPT	CPMoe, RTE / CPMOa, PM	INT / Archi / RSSI	PO, LSI, SCM
Rapport de Test (RT) de TAO (2 porteurs / BAPT et EXP)	CPT	CPMoe, RTE / CPMOa, PM	INT / Archi / RSSI	PO, LSI, SCM



## Vérification d'Aptitude (VA)

La vérification d'aptitude est l'ensemble des tests effectués par l'administration débutant à la Mise en Ordre de Marche\* (MOM) et se terminant par une décision de "GO/NO GO" de mise en production.

Ces tests sont constitués de trois niveaux :

- Le niveau Test Intégration Applicative (TIA)
- Le niveau Recette Fonctionnelle Utilisateur (RFU)
- Le niveau Test d'Acceptation Opérationnelle (TAO).

\* La Mise en Ordre de Marche est caractérisée par un document émanant du titulaire certifiant qu'il a effectué la livraison répondant à ses obligations de vérification (\*\*) livraison devant être validée par l'administration.

\*\* Les obligations de vérification portent d'une part sur la conformité du contenu de la livraison correspondante à la description de la ou des prestations d'autre part sur la qualité requise. A ce titre les livraisons logicielles doivent faire l'objet de tests de niveau système (Qualimétrie, conformité des développements conformément aux besoins techniques et fonctionnels, tests de performance).

## 2.4 Activités de test

Comme dans le chapitre précédent, les profils identifiés dans les RACI ont un code couleur, permettant d'identifier l'acteur suivant le mode projet :

- Profil : écrit en noir ; Profil commun au mode de gestion de projet cycle en V et Agile
- **Profil** : écrit en vert ; Profil particulier pour un projet en mode Cycle en V
- **Profil** : écrit en orange ; Profil particulier pour un projet en mode Agile

### 2.4.1 Activité Organiser

Activité : Organiser				
<p>Objectifs :</p> <ul style="list-style-type: none"><li>• Définir les priorités et risques</li><li>• Définir le périmètre des tests de chaque niveau de test et les conditions de succès associées</li><li>• Identifier les contributeurs et de leurs responsabilités</li><li>• Définir les moyens (environnements/données, outils d'automatisation ou de simulations d'applications, postes de travail, modèle de l'outil de gestion des tests à utiliser...)</li><li>• Intégrer l'organisation générale de la conduite des activités des tests (macro-planning, charges, Condition d'Acceptation Livraison, gouvernance...) dans le plan Projet</li></ul>				
Entrants				
<b>Projet cycle en V :</b> <ul style="list-style-type: none"><li>• Pour Stratégie V1 : CCTP, CCAP, SFG, plan projet, rapport d'études de faisabilité, DAT initialisé pour la partie fonctionnelle, Dossier de sécurité, Processus métier</li><li>• Pour Stratégie V2 : Stratégie V1, DAT, SFD, Protocole de packaging et de livraison applicatif, Contrat d'interface, Guide d'intégration, plan de sauvegarde, plan de continuité, cahier de supervision</li></ul>		<b>Projet Agile :</b> <ul style="list-style-type: none"><li>• Pour la Stratégie : CCTP, CCAP, plan projet, rapport d'études de faisabilité, Dossier d'Intention d'Architecture, processus métier, dossier de sécurité, story map</li><li>• Pour les annexes incrément : backlog des users stories et enablers (statut ready), exigences techniques et fonctionnelles, DAT</li></ul>		
Tâches				
<ul style="list-style-type: none"><li>• Rédiger la stratégie de test</li><li>• Réaliser les ateliers de travail nécessaires</li><li>• Estimer la charge de test par niveau de test</li><li>• Faire valider par les contributeurs</li></ul>				
Livrables	R	A	C	I
Stratégie de tests	CPT	CPMoe, RTE	Archi, CPMoa, PM, PO, SCM, LSI, RSSI	Tous
Macro - Estimation de la charge des tests	CPT	CPMoe, RTE	Archi, CPMoa, PM, PO, SCM, LSI, RSSI	Tous

## 2.4.2 Activité Analyser les TS

Activité : Analyser les TS				
<p>Objectifs :</p> <ul style="list-style-type: none"><li>Détailler les exigences produit en exigences à tester</li><li>Identifier les scénarios</li><li>Formaliser les étapes de traitement des IHM</li><li>Formaliser les étapes de traitement batch</li><li>Formaliser les étapes de traitement des éditions</li></ul>				
Entrants				
<p><b>Projet cycle en V :</b></p> <ul style="list-style-type: none"><li>Stratégie de tests (finalisé)</li><li>Manuel d'installation</li><li>Protocole Packaging et livraison applicatifs</li><li>CCT</li><li>SFD (IHM, Batch)</li></ul>		<p><b>Projet Agile :</b></p> <ul style="list-style-type: none"><li>Stratégie de l'incrément (finalisée)</li><li>Manuel d'installation</li><li>Protocole Packaging et livraison applicatifs</li><li>CCT</li><li>Users Story / Enabler</li></ul>		
Tâches				
<ul style="list-style-type: none"><li>Créer/Mettre à jour les exigences à tester et les étapes de traitement</li><li>Estimer les charges de façon détaillées</li><li>Organiser et Animer les ateliers de travail</li><li>Analyser et préparer les travaux de mise à disposition des moyens</li><li>Identifier les scénarios de test</li><li>Analyser le provisionnement des environnements de tests nécessaire</li></ul>				
Livrables (développement interne)	R	A	C	I
Charge et planning détaillé du niveau de test	CPT	CPMoe, RTE	Tous	
Description des exigences et leur arborescence	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI
Description des scénarios	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI
Demandes d'environnement	CPT	LSI	CPMoe, RTE	
Livrables (développement externe)	R	A	C	I
Charge et planning détaillé du niveau de test	Fournisseur	PResp	CPT	CPMoe, RTE, LSI
Description des exigences et leur arborescence	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI
Description des scénarios	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI
Gestion des d'environnements	Fournisseur	PResp		

## 2.4.3 Activité Analyser les TIA

Activité : Analyser les TIA				
<p>Objectifs :</p> <ul style="list-style-type: none"><li>Détailler les exigences produit en exigences à tester</li><li>Formaliser les étapes d'intégration applicatives</li><li>Identifier les scénarios</li></ul>				
Entrants				
<p><b>Projet cycle en V :</b></p> <ul style="list-style-type: none"><li>Stratégie de tests (finalisée)</li><li>DAF</li><li>SFG</li><li>SFD (IHM, Batch)</li><li>Processus métier</li><li>Contrats d'interfaces</li></ul>		<p><b>Projet Agile :</b></p> <ul style="list-style-type: none"><li>Stratégie de tests de l'incrément (finalisée)</li><li>DAF</li><li>User Stories</li><li>Enablers</li><li>Processus métier</li><li>Contrats d'interfaces</li></ul>		
Tâches				
<ul style="list-style-type: none"><li>Créer/Mettre à jour les exigences à tester et les étapes de traitement</li><li>Estimer les charges de façon détaillées</li><li>Organiser et Animer les ateliers de travail</li><li>Analyser et préparer les travaux de mise à disposition des moyens</li><li>Identifier les scénarios de test</li><li>Analyser le provisionnement des environnements de tests nécessaire</li></ul>				
Livrables	R	A	C	I
Charge et planning détaillé du niveau de test	CPT	CPMoe, RTE	Tous	
Description des exigences et leur arborescence	Fournisseur / Test	CPT	Archi	CPMoe, RTE
Description des scénarios	Fournisseur / Test	CPT	Archi	CPMoe, RTE
Demandes d'environnement	CPT	LSI	CPMoe, RTE	

## 2.4.4 Activité Analyser les TAO

Activité : Analyser les TAO				
Objectifs :				
<ul style="list-style-type: none"><li>Détailler les exigences techniques en exigences à tester</li><li>Identifier les scénarios</li></ul>				
Entrants				
<b>Projet cycle en V :</b>		<b>Projet Agile :</b>		
<ul style="list-style-type: none"><li>Stratégie de test (finalisée)</li><li>DAT</li><li>CDC performance</li><li>Dossier d'homologation de sécurité</li><li>Manuel d'installation, Protocole de packaging et de livraison applicatif, Manuel d'exploitation, plan de sauvegarde, plan de continuité, cahier d'ordonnancement, cahier de supervision</li></ul>		<ul style="list-style-type: none"><li>Stratégie de tests de l'incrément (finalisée)</li><li>DAT</li><li>CDC performance</li><li>Dossier d'homologation de sécurité</li><li>Manuel d'installation, Protocole de packaging et de livraison applicatif, Manuel d'exploitation, plan de sauvegarde, plan de continuité, cahier d'ordonnancement, cahier de supervision</li></ul>		
Tâches				
<ul style="list-style-type: none"><li>Créer/Mettre à jour les exigences à tester et les étapes de traitement</li><li>Estimer les charges de façon détaillées</li><li>Organiser et Animer les ateliers de travail</li><li>Analyser et préparer les travaux de mise à disposition des moyens</li><li>Identifier les scénarios de test</li><li>Analyser le provisionnement des environnements de tests nécessaire</li></ul>				
Livrables	R	A	C	I
Charge et planning détaillé du niveau de test	CPT	CPMoe, RTE	Tous	
Description des exigences et leur arborescence	Perf, INT, Test	CPT	Archi, RSSI	CPMoe, RTE, LSI, SCM
Description des scénarios	Perf, INT, Test	CPT	Archi, RSSI	
Demandes d'environnement	CPT	LSI	CPMoe, RTE	

## 2.4.5 Activité Analyser les RFU

Activité : Analyser les RFU				
<p>Objectifs :</p> <ul style="list-style-type: none"><li>Détailler les exigences métier en exigences à tester</li><li>Formaliser les étapes des processus impactés</li><li>Identifier les scénarios</li></ul>				
Entrants				
<p><b>Projet cycle en V :</b></p> <ul style="list-style-type: none"><li>Stratégie de tests (finalisée)</li><li>Processus métier</li><li>Manuel utilisateur</li><li>SFG</li></ul>		<p><b>Projet Agile :</b></p> <ul style="list-style-type: none"><li>Stratégie de tests de l'incrément</li><li>Processus métier</li><li>Manuel utilisateur</li><li>User Story / Feature / EPIC</li></ul>		
Tâches				
<ul style="list-style-type: none"><li>Créer/Mettre à jour les exigences à tester et les étapes de traitement</li><li>Estimer les charges de façon détaillées</li><li>Organiser et Animer les ateliers de travail</li><li>Analyser et préparer les travaux de mise à disposition des moyens</li><li>Identifier les scénarios de test</li><li>Analyser le provisionnement des environnements de tests nécessaire</li></ul>				
Livrables	R	A	C	I
Charge et planning détaillé du niveau de test	CPT	CPMoa, PM	Tous	
Description des exigences et leur arborescence	MOA	CPMoa, PM	CPT /BA / Archi / PO	CPMoe, RTE, LSI, SCM
Description des scénarios	MOA	CPMoa, PM	CPT /BA / Archi / PO	CPMoe, RTE, LSI, SCM
Demandes d'environnement	CPT	LSI	CPMoe, RTE	CPMoa, PM, PO

## 2.4.6 Activité Spécifier

Activité : Spécifier				
<p>Objectifs :</p> <ul style="list-style-type: none"> <li>Intégrer dans l'outil de gestion de tests l'ensemble des éléments décrits dans la phase d'analyse</li> <li>Définir les cas de tests</li> <li>Construire la couverture des exigences par les cas</li> <li>Définir la valorisation logique des données</li> <li>Définir /Mettre à jour les scénarios et organiser les campagnes de test</li> </ul>				
Entrants				
<ul style="list-style-type: none"> <li>Stratégie de test</li> <li>Exigences à tester</li> <li>Entrants de l'activité Analyser</li> </ul>				
Tâches				
<ul style="list-style-type: none"> <li>Saisir dans l'outil de gestion des tests les exigences dans le module exigences</li> <li>Créer et saisir dans l'outil de gestion des tests les cas de tests</li> <li>Saisir dans l'outil de gestion des tests les scénarios</li> <li>Définir les campagnes</li> </ul>				
Livrables	R	A	C	I
TS (dev Interne) : Référentiel de test complété	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI, SCM
TS (dev externe) : Référentiel de test complété	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI, SCM
TIA : Référentiel de test complété	Fournisseur / Test	CPT	Archi / BA	CPMoe, RTE, LSI, SCM
TAO : Référentiel de test complété	Perf, INT, Test	CPT	Archi	CPMoe, RTE, LSI, SCM
RFU : Référentiel de test complété	MOA	CPT	PO	CPMoa, PM, CPMoe, RTE, LSI, SCM

## 2.4.7 Activité Implémenter

Activité : Implémenter				
Objectifs :				
<ul style="list-style-type: none"> <li>Valoriser les données physiques</li> <li>Vérifier la mise à disposition des moyens nécessaires en environnement, traitements et données de test</li> </ul>				
Entrants				
<ul style="list-style-type: none"> <li>Livrables de Spécifier</li> <li>Environnements de test (infrastructure, environnement installé, bouchons, données de test...)</li> </ul>				
Tâches				
<ul style="list-style-type: none"> <li>Valoriser les données des scénarios dans l'outil de gestion des tests</li> <li>S'assurer de la mise en place des moyens nécessaires : bouchons, enchaînement des traitements, données de test</li> </ul>				
Livrables	R	A	C	I
TS (dev interne) : cahier de test valorisé	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI, SCM
TS (dev externe) : cahier de test valorisé	Fournisseur	CPT	BA, Archi, RSSI	CPMoe, RTE, LSI, SCM
TIA : cahier de test valorisé	Fournisseur / Test	CPT	Archi / BA	CPMoe, RTE, LSI, SCM
TAO : cahier de test valorisé	Perf, INT, Test	CPT	Archi	CPMoe, RTE, LSI, SCM
RFU : cahier de test valorisé	MOA	CPT	PO	CPMoa, PM, CPMoe, RTE, LSI, SCM
Mise en place de l'infrastructure des tests : Environnements, JDD physiques, pilotes/bouchons...	Perf, INT, Test, Fournisseur	CPT	LSI, Archi, PO, MOA	CPMoa, PM, CPMoe, RTE, LSI, SCM



## 2.4.8 Activité Exécuter et rendre compte

Activité : Exécuter et Rendre Compte				
Objectifs : <ul style="list-style-type: none"> <li>Exécuter les cas de tests</li> <li>Identifier, déclarer et vérifier les anomalies</li> <li>Etablir un tableau de bord de suivi hebdomadaire (cycle en V)</li> <li>Informer quotidiennement et mettre à jour un tableau de bord (mode Agile à confirmer)</li> </ul>				
Entrants				
<ul style="list-style-type: none"> <li>Livrables phase Implémenter</li> <li>Stratégie de tests (finalisée)</li> <li>Les moyens prévus dans la stratégie</li> <li>Les composants applicatifs livrés dans le respect des conditions de succès et du planning</li> </ul>				
Tâches				
<ul style="list-style-type: none"> <li>Dérouler les cas de test conformément à la stratégie de test et à la campagne de définie (enchaînement des scénarios valorisés)</li> <li>Contrôler les résultats observés par rapport aux résultats attendus</li> <li>Sauvegarder les preuves de l'exécution dans l'outil de gestion des tests</li> <li>Répertorier les anomalies et suivre le Workflow</li> <li>Produire et diffuser le reporting</li> </ul>				
Livrables	R	A	C	I
La liste des anomalies TS	Fournisseur	CPT		CPMoe, RTE, LSI, SCM
La liste des anomalies TIA	Fournisseur / Test	CPT		CPMoe, RTE, LSI, SCM
Liste des anomalies TAO	Perf, INT, Test	CPT		CPMoe, RTE, LSI, SCM
La liste des anomalies RFU	MOA	CPT		CPMoa, PM, PO, CPMoe, RTE, LSI, SCM
Le cahier de tests exécuté (TS)	Fournisseur	CPT		CPMoe, RTE, LSI, SCM
Le cahier de tests exécuté (TIA)	Fournisseur / Test	CPT		CPMoe, RTE, LSI, SCM
Le cahier de tests exécuté (TAO)	Perf, INT, Test	CPT		CPMoe, RTE, LSI, SCM
Le cahier de tests exécuté (RFU)	MOA	CPT		CPMoa, PM, PO, CPMoe, RTE, LSI, SCM
Le reporting d'avancement des tests et de la qualité constatée	CPT	CPT	Tous	Tous

## 2.4.9 Activité Conclure

Activité : Conclure				
Objectifs : <ul style="list-style-type: none"> <li>Conclure le niveau de test par un rapport de test (RT)</li> <li>Faire un bilan du niveau de test</li> <li>Capitaliser dans les référentiels de test</li> </ul>				
Entrants				
Livrables de l'activité Executer / Rendre Compte				
Tâches				
<ul style="list-style-type: none"> <li>Rédiger le rapport de test et le soumettre à validation</li> <li>Rédiger le bilan qualitatif et quantitatif du niveau de test, mettre en évidence les éléments capitalisables</li> <li>Réconcilier les exigences/cas de test / scénario éligibles à la capitalisation dans les référentiels par Domaine fonctionnel / Processus</li> </ul>				
Livrables	R	A	C	I
RT de TS	CPT	CPMoe, RTE	Fournisseur	Tous
RT de TIA	CPT	CPMoe, RTE	Fournisseur / Test	Tous
RT de TAO	CPT	CPMoe, RTE	Perf, INT, Test	Tous
RT de RFU	CPT	CPMoa, PM	MOA	Tous
TS : Référentiels de tests mis à jour	Fournisseur	CPT		CPMoe, RTE, LSI, SCM
TIA : Référentiels de tests mis à jour	Fournisseur / Test	CPT		CPMoe, RTE, LSI, SCM
TAO : Référentiels de tests mis à jour	Perf, INT, Test	CPT		CPMoe, RTE, LSI, SCM
RFU : Référentiels de tests mis à jour	MOA	CPT		CPMoa, PM, PO, CPMoe, RTE, LSI, SCM
Bilan	CPT	CPMoe / CP Moa	Tous	Tous

## 2.4.10 Rapport de Test (RT)

Le Rapport de Test est un élément clé dans le processus de test.

Il certifie la fin de certains niveaux de test et donc l'acceptation du niveau de qualité atteint.

Deux rapports de test sont indispensables à la mise en Production : le RT de RFU et le RT de TAO. C'est à partir des éléments de ces deux RT que sera décidée la mise en Production.

Chaque rapport de test est porté par le responsable du niveau de Test

Le rapport de test doit :

- Donner une description du projet
  - La version développée qui a fait l'objet des tests et qui monte en production
  - Le périmètre du projet
  - Les impacts du projet
- Préciser la couverture des tests
  - Ce qui était prévu de tester et de ne pas tester
- Donner des précisions sur les campagnes de tests
  - Lister les dates clés des campagnes
  - Préciser ce qui a réellement été testé
  - Rappeler les changements de périmètre (arbitrages, rajout...) et leur justification
  - Rappeler l'historique des aléas (indisponibilités de moyens, retard de livraisons...)
- Présenter précisément le résultat des campagnes de tests
  - Dresser le bilan des anomalies
  - Présenter l'évaluation en risque métier / technique des anomalies résiduelles
  - Donner les préconisations pour le GO / NO GO en production (RT TAO et RFU)
  - Lister les réserves
  - Présenter un plan de résolution

## 3 La stratégie de test

### 3.1 Introduction

La stratégie de test est un document clé du processus de tests.

- Il doit obligatoirement être **initialisé** lors du lancement du projet de façon collaborative par tous les acteurs du projet.
- Il doit être vu comme le contrat scellant l'accord entre toutes les parties prenantes du processus de test du projet.

Ce document est le fruit d'une réflexion commune, il doit décrire les points suivants. :

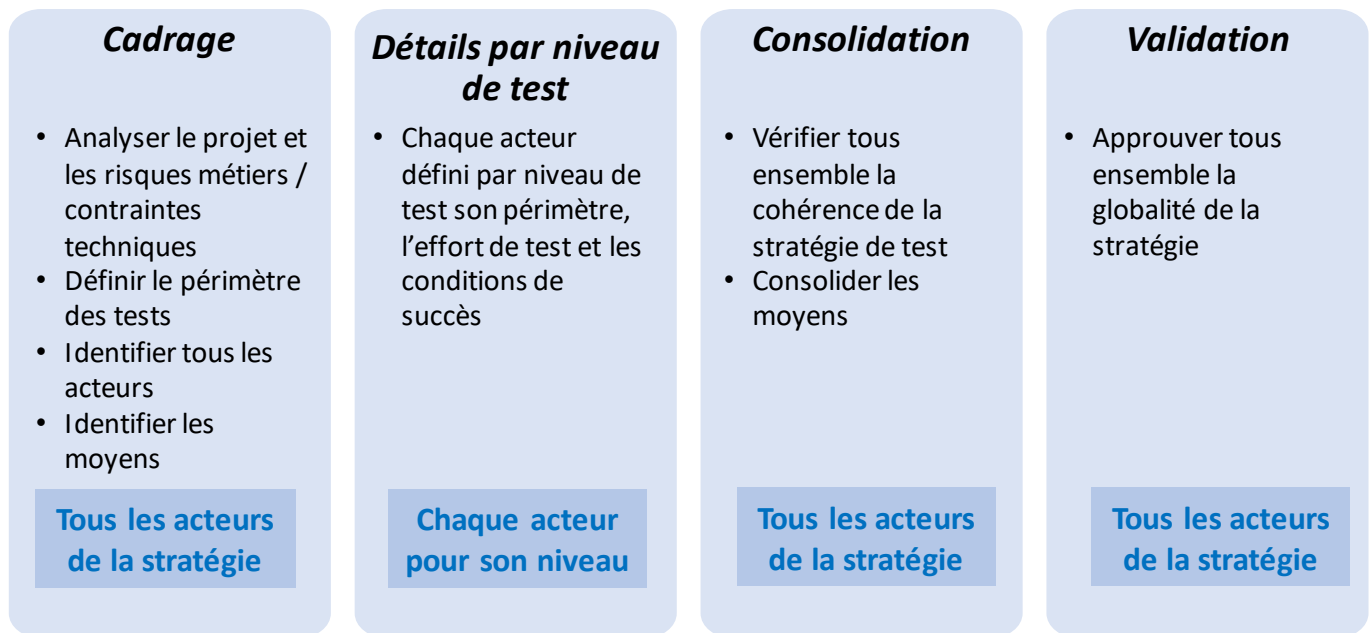
- Le 'quoi' tester : les macro-exigences,
- Le 'comment' tester : les moyens à mettre en œuvre,
- Le 'par qui' sont fait les tests : le RACI détaillé,
- Comment les risques métier et technique, sont assumés ensemble,
- Le juste effort de test, prenant en compte les évaluations de risque (cf chapitre 4.2.4),
- La stratégie de TNR et d'automatisation des tests,
- Le périmètre par niveau de tests,
- Le périmètre d'exclusion,
- Les critères de transition, de qualité, de couverture par les tests,
- Les besoins : les moyens de toutes sortes.

Ce document a les caractéristiques suivantes :

- C'est la résultante d'une analyse des livrables projets,
- Il est dynamique :
  - Cycle en V : une version V1 puis V2 (s'affine au fur et à mesure),
  - Agile : un tronc commun et des compléments par incrément
- Il est validé par tous les acteurs,
- Il fait le lien entre les activités de tests et les autres activités du projet.

## 3.2 Démarche de construction

Les étapes de construction de la stratégie de test pour un projet géré en cycle en V ou en mode Agile sont les mêmes ; C'est la dynamique d'enchaînement des étapes qui est différente. Le schéma ci-dessous présente les grandes étapes de construction d'une stratégie de tests :



### 3.2.1 Cadrage - Analyser

Les acteurs du projet procèdent à l'analyse des livrables du projet afin de définir ou d'identifier les spécificités du projet.

L'analyse portera principalement sur l'étude de l'architecture fonctionnelle et technique, des processus métiers, des risques métier et des contraintes.

Il conviendra de

- S'assurer que les contraintes et objectifs du projet sont bien identifiés
- Vérifier la disponibilité des éléments clés de l'architecture technique et fonctionnelle afin d'identifier toutes les applications connexes
- Vérifier que toutes les interfaces sont bien identifiées
- S'assurer que tous les acteurs sont identifiés et intégrer à la construction de la stratégie
- Lister les risques métiers et techniques qui s'appliquent

Selon les besoins l'étude de la documentation sera complétée par des ateliers spécifiques.

### 3.2.2 Cadrage - Périmètre et Moyens des tests

En fonction des résultats de l'analyse, les acteurs projets définissent la criticité des macro-exigences, élaborent en commun le périmètre des tests ainsi que l'effort de test à produire pour chacune des macro-exigences.

Cet effort de test est la résultante de l'analyse des risques (métiers, techniques)

Les acteurs projets vont identifier tous les moyens nécessaires à la prise en charge des tests tels que définis. On identifiera tous les moyens, qu'ils soient humains, matériels, logiciels, formations, ...

### 3.2.3 Cadrage - Organisation

Cette dernière étape consiste à quantifier, planifier et organiser les contours des tests tels que définis.

Sous la responsabilité du chef de projet,

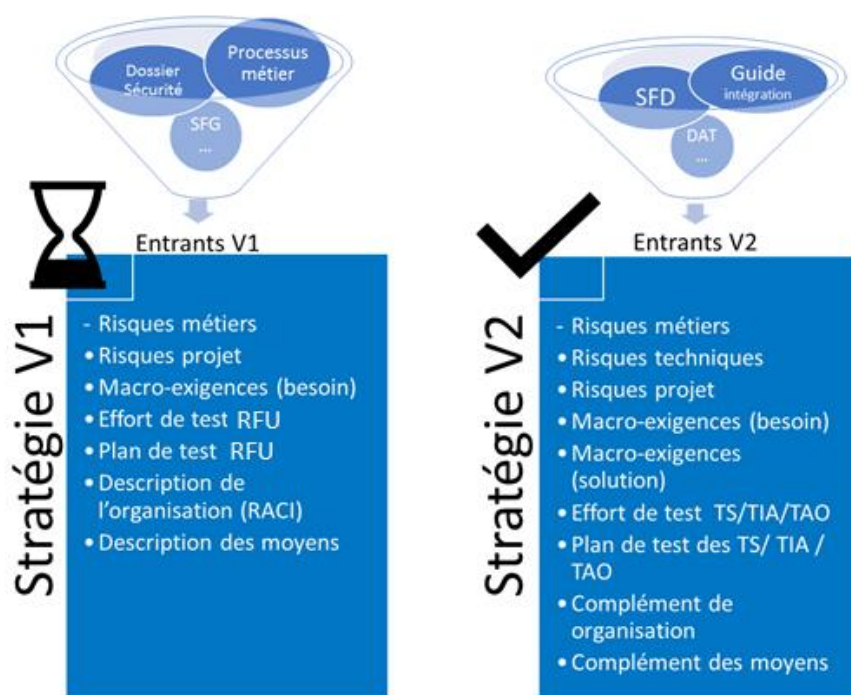
- Identification et description précise de la répartition des travaux acteur par acteur,
- Planification détaillée des activités en précisant les acteurs responsables,
- Identification et description des critères qualités, et des conditions d'acceptation de passage d'un acteur à un autre,
- Chiffrage et planification des moyens identifiés.

### 3.3 La dynamique de construction pour un projet Cycle en V

En Cycle en V, c'est le chef de projet MOE qui pilote la « construction » de la stratégie de tests.

La stratégie de tests est un document qui se construit durant les premières phases du projet :

- Une première version : V1 est diffusée à la fin de la phase Spécifications Fonctionnelles,
- Une Seconde version : V2 est diffusée à la fin de la phase Conception Technique.



### 3.3.1 Stratégie de tests V1

Cette première version du document, qui est initialisée au lancement du projet, et finalisée durant la phase projet Spécifications fonctionnelles, traite des points suivants :

- Identification et présentation des risques métiers et projets
- Identification des contraintes techniques à partir du DAT
- Identification des contraintes de sécurité à partir du dossier d'homologation de sécurité
- Identification des processus métiers impactés
- Identification des acteurs à partir des processus métiers impactés
- Définition de la liste des macro-exigences de type **besoin** (métier / sécurité / technique)
- Identification et description du périmètre de test pour le niveau de test RFU (incluant l'effort de tests)
- Initialisation de l'organisation et du RACI
- Initialisation des moyens

Cette première version est élaborée à partir des livrables projets et produit suivants : Plan projet, SFG, Processus métier, Dossier Sécurité, DAT.

### 3.3.2 Stratégie de tests V2

Cette seconde version du document, qui est élaborée durant la phase projet 'Conception technique', traite des points suivants :

- Identification et présentation des risques techniques
- Consolidation des risques
- Consolidation des processus métiers impactés
- Identifications des applications et des infrastructures impactées
- Consolidation des acteurs impliqués
- Définition de la liste des macro-exigences de type **produit**
- Identification et description du périmètre de test pour les niveaux de test TS/TIA/TAO (incluant l'effort de tests)
- Finalisation du périmètre de test RFU
- Finalisation et description des exigences qualité et des CAL
- Finalisation de l'organisation et du RACI
- Finalisation des moyens

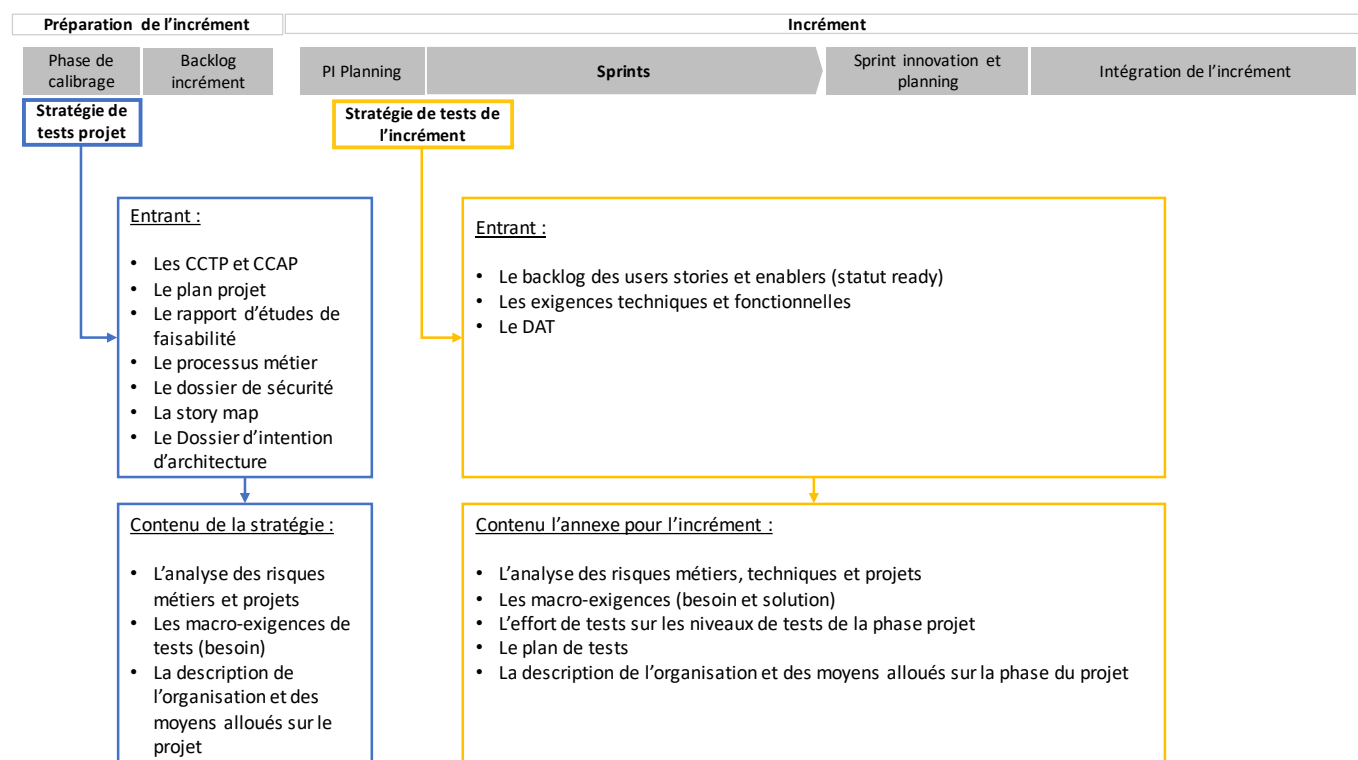
Cette version est élaborée à partir des livrables projets et produit suivants : SFD, DAT, DAF, protocole packaging et livraison applicatif, contrats interface, guide intégration, plan de sauvegarde, plan de continuité, cahier de supervision.

### 3.4 La dynamique de construction pour un projet en mode Agile

En Agile, c'est le RTE qui pilote la « construction » de la stratégie de tests, mais la rédaction est portée par le CP Tests.

La stratégie de tests est un dossier qui est composée :

- Un document principal, rédigé au début du projet et valable pour l'ensemble des incréments.
- Une annexe par incrément.

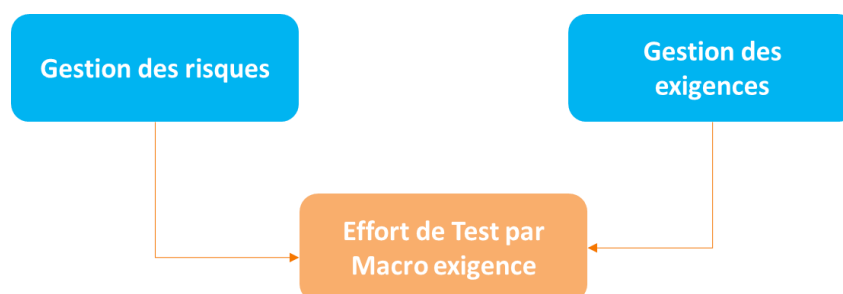


### 3.5 Focus sur l'effort de Test

Dans l'activité de test, identifier où doit se porter l'effort de test est clé pour s'assurer de la qualité de la solution livrée et pour piloter les tests à réaliser. C'est dans la stratégie de test que va être estimé l'effort de test.

Cet effort est la résultante de deux composantes qui sont des entrants de la stratégie de test :

- Les facteurs de risque produit et leur criticité,
- Les (macro)exigences et leur criticité.





Pour chaque macro-exigence, on utilise cette grille pour déterminer l'effort en fonction de la criticité du facteur de risque associé et de la criticité de la macro-exigence.

Par défaut, s'il n'y a pas de risque associé à la macro-exigence alors considérer que le risque est faible.

Macro Exigence	Macro Exigence Criticité « Forte »	Macro Exigence Criticité « Moyenne »	Macro Exigence Criticité « Faible »
Facteur de risque			
Criticité - Critique	Maximal	Maximal	Faible
Criticité - Elevée	Maximal	Elevé	Faible
Criticité – Moyenne	Elevé	Moyen	Faible
Criticité - Faible	Moyen	Faible	Faible

L'effort de test dépend du contexte du projet, cependant on donne une définition générique de chaque niveau d'effort. Pour cela, on peut considérer que l'effort de test est composé de :

- La profondeur des tests de l'exigence, c'est à dire le nombre de cas de test pour couvrir l'exigence.
- Le panel de données mis en œuvre lors des tests

L'effort de test doit être évalué pour chaque macro-exigence de chaque niveau de test.

Illustration de l'effort de test :

Effort	Définition	Exemple
Maximal	-Profondeur maximale toutes les combinaisons possibles pour couvrir l'exigence -Multiples jeux de données	Pour un use case Métier - Toutes les variantes (nominale, alternatives, exceptions) - Toutes les types de panel de données associées au use case
Elevée	-Profondeur élevée par une couverture importante (au moins 50% de la combinatoire) de l'exigence par des cas de test -Jeux données standards	Pour un batch -Tests de toutes les étapes de manière unitaire cas passant et cas non passant et des Règles de gestion associées -Jeux de données dédiés pour tester chaque règle de gestion associée à une étape du batch
Moyen	-Profondeur moyenne par la prise en compte de certain test et pas d'autres -Jeux de données unique	Pour des tests de performance -Test de montée en charge -Pas de Test de pallier -Pas de test aux limites
Faible	-Profondeur faible en ne réalisant que le cas passant/nominal pour couvrir l'exigence -Jeux de données simple	Pour une IHM - Cas passant sur le contrôle des zones d'un écran

## 4 Les Moyens

L'anticipation des besoins dès la stratégie de test est un facteur de réussite du projet.

L'objectif est de disposer du bon environnement, des bonnes applications, des bons jeux de données et des bonnes connectivités pour pouvoir dérouler les tests prévus dans chacun des niveaux de test.

La préparation et la mise en place des moyens nécessaires aux tests doivent être planifiées et chiffrées dès la stratégie de recette.

Les moyens mis à disposition sont :

- Les environnements de tests administrés,
- Les outils,
- Le catalogue de services du Centre de Compétences de Tests.

### 4.1 Les environnements de test

Les environnements ont des caractéristiques spécifiques qui permettent de répondre aux besoins des différents niveaux de test.

Les environnements sont utilisés par différentes équipes, cela implique la mise à disposition de moyens et des contraintes pour les utilisateurs :

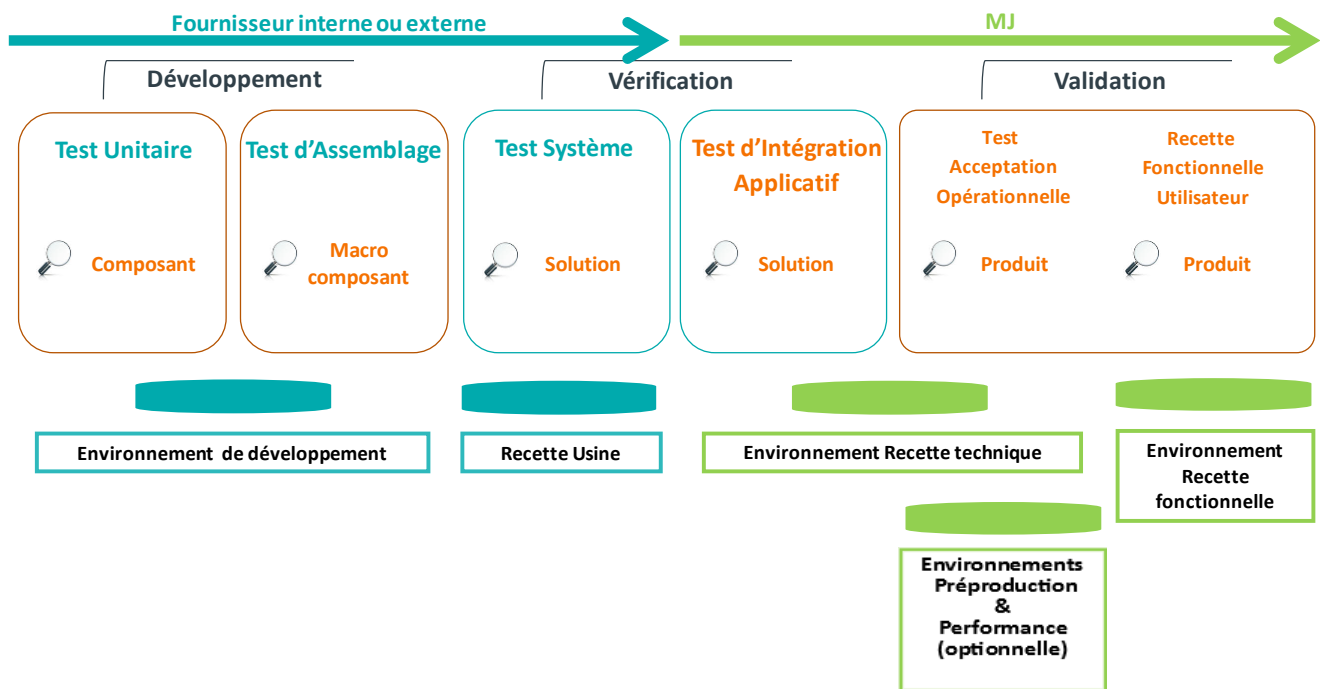
- Ils bénéficient de moyens mutualisés et administrés
- Un certain nombre de contraintes est à prendre en compte lors de leurs utilisations
- Un certain nombre de règles est à appliquer sur la mise à jour des données et des composants applicatifs

Chaque environnement a son niveau de cohérence, tant au niveau fonctionnel que technique. La différenciation entre les environnements repose sur :

- La complétude de l'infrastructure
- Le volume de données (d'une image de la production à un volume réduit de données ou des données construites),
- Les traitements batch exécutés, (lancés à la demande, lancement automatique partiel)
- Les connexions avec d'autres services transverses, d'autres applications ou d'autres services fournis par des partenaires, a contrario le bouchonnage de certaines interfaces.

Ce chapitre vise à décrire les différents environnements techniques. Il a pour principale vocation de définir l'ensemble des plates-formes utiles aux opérations.

Le schéma présente une vue d'ensemble des environnements de tests :



#### 4.1.1 Environnement de Développement

L'environnement de développement est nécessaire au développement, tests unitaires et tests d'assemblage.

L'environnement de développement est hébergé par le fournisseur (développement interne = MJ, développement externe = prestataire).

Il peut être subdivisé en plusieurs sous-environnements pour gérer les versions livrées et les différentes branches en cours de réalisation.

La mise en œuvre de cet environnement et son maintien en condition opérationnelle est à la charge du fournisseur (développement interne = MJ, développement externe = prestataire).

#### 4.1.2 Environnement de Recette Usine

L'environnement de recette usine est nécessaire à la réalisation des tests systèmes, d'installation et des tests techniques (conformité d'architecture).

Il est hébergé par le fournisseur (développement interne = MJ, développement externe = prestataire).

La mise en œuvre de cet environnement et son maintien en condition opérationnelle est à la charge du fournisseur.

Le Ministère n'exige pas que ses environnements de recette aient des architectures physiques identiques à ceux de production :

- Les ressources physiques telles que CPU, RAM et disque des machines virtuelles (ou conteneurs) n'ont pas être identiques à celles de production mais devront être suffisamment adaptées afin de permettre un fonctionnement non dégradé de l'application.
- Il n'est pas non plus exigé que l'architecture physique du réseau soient respectée ni que tous les équipements la définissant soient présents. En revanche, concernant le cas particulier du boîtier de répartition de charge (F5 BigIP) présent en entrée site (internet et intranet) ainsi qu'en zone applicative, sa présence n'est pas exigée sur l'environnement de recette usine mais il devra être remplacé à minima par une solution logicielle permettant de réaliser de la répartition de charge (Apache HTTPD par exemple).

Pour autant, l'environnement de recette usine doit être suffisamment représentatif, et les architectures logiques doivent

être identiques :

- L'environnement de recette usine devra respecter la répartition des services telle que définie sur l'environnement de production. La mutualisation de machines (ou conteneurs) et le regroupement de services sur celles-ci ne sont pas acceptés.
- Si pour des raisons de disponibilité et de gestion améliorée du fail-over certains services sont redondés sur l'environnement de production (selon un schéma actif-actif) alors ce principe doit être également reconduit sur l'environnement de recette usine. En revanche il n'est pas nécessaire que le nombre d'instances d'un même service soit identique entre la production et la recette usine. Sauf contrainte, un maximum de deux instances est suffisant afin de pouvoir réaliser des tests de répartition de charge et de reprise sur erreur.

Il est également exigé que le système d'exploitation des machines virtuelles (ou du système hôte dans le cas d'un déploiement par conteneurs) soit identique entre les environnements de recette usine et de production (tel que défini dans le Dossier d'Architecture Technique de l'application).

#### 4.1.3 Environnement de Recette Fonctionnelle Utilisateur

L'environnement de recette fonctionnelle permet de valider que l'application est fonctionnellement en tout point conforme aux spécifications fonctionnelles générales.

L'environnement de de recette fonctionnelle est hébergé par le MJ.

L'administration de l'environnement de recette technique et son maintien en condition opérationnelle est à la charge de la Squad d'intégration de la tribu (ou de l'équipe technique EXP dans le cas, ou le projet est rattaché à un domaine ou un bureau).

En cas d'utilisation de données de production dans le cadre de la recette fonctionnelle, elles doivent respecter la démarche de sécurité liée aux données de production, pour certaines données, être anonymisées.

Contrairement aux environnements de recette technique et de recette usine, la redondance des services n'est pas exigée. En revanche, l'architecture logique et la répartition des services sur les machines (ou sur les conteneurs) doivent être respectées afin de ne pas multiplier les procédures d'installation.

#### 4.1.4 Environnement de tests techniques

L'environnement de tests permet de réaliser les tests TIA, mais également un ensemble de tests techniques issus du niveau TAO tels que les tests d'installation/packaging, et tous autres tests techniques de validation définis dans la stratégie de tests du projet

L'environnement de tests techniques est hébergé par le MJ.

L'administration de l'environnement de tests techniques et son maintien en condition opérationnelle est à la charge des équipes d'exploitation.

Les droits d'accès à cet environnement sont limités.

#### 4.1.5 Environnement de Pré-production

L'environnement de pré-production permet de réaliser les tests d'Acceptation Opérationnelles ciblant l'exploitation du produit (sauvegarde, ordonnanceur et supervision).

Il permet de re-jouer les procédures d'installation dans un environnement cible.

C'est également dans cet environnement que sont réalisés, le cas échéant, les tests techniques de l'architecture physique :

- Test de haute disponibilité (cluster),
- Test de bascule (PRA).

L'environnement de pré-production est hébergé par le MJ.

L'administration de l'environnement de pré-production et son maintien en condition opérationnelle est à la charge des équipes d'exploitation.

Les droits d'accès à cet environnement sont limités.

#### 4.1.6 Environnement de Performance (optionnel)

L'environnement de performance peut être nécessaire en fonction du contexte du projet et des exigences de performance requises.

Il est sous la responsabilité conjointe des équipes d'exploitation (pour la partie infrastructure) et de la MOE pour l'installation et l'organisation des tests de performance.

Il est hébergé dans le datacenter du ministère

## 4.2 Les outils standards utilisés dans le cadre des tests MJ

Le document de référence des outils du SNUM est le Cadre de Cohérence Technique.

Le tableau ci-dessous présente les principaux outils utilisés pendant la phase de tests :

Solution	Logiciel
Tests unitaires	J Unit / PHP Unit
Tests fonctionnels automatisés	Sélénium
Gestion des anomalies	Mantis
Gestion des référentiels de test	Squash TM
Orchestration d'Intégration continue	Jenkins/Maven
Qualimétrie	SonarQube (pmd, findbugs, checkstyle) Check Marx CAST AIP
Anonymisation	REGDATA
Tests de charge	JMeter/DBUnit

## 5 La gouvernance

La gouvernance des tests permet de couvrir deux grands items :

- La gestion des anomalies détectées lors des phases de tests
- Le suivi de l'avancement des étapes de tests.

Ces deux items sont suivis différemment suivant le mode de gestion du projet ; le schéma suivant présente la comitologie ou cérémonie suivant le mode de gestion :

Objectifs	Comités pour les projets cycle agile	Comités pour les projets cycle en V
Suivre les anomalies de façon quotidienne au sein des squads / sur les projets	▪ Daily stand-up (partie dédiée aux tests)	▪ Comités anomalies
Suivre les anomalies de façon régulière au niveau des tribus	▪ Comités des anomalies tribal	
Suivre l'avancement des tests au sein des squads / tribus sur les projets	▪ Weekly testing synchro	
Présenter l'avancement de l'incrément, en montrant le fonctionnement des dernières fonctionnalités implémentées	▪ Démo système	
Partager avec le métier les tests réalisés en cohérence avec le périmètre fonctionnel attendu	▪ Sprint review	▪ Comités de suivi des tests
Faire le bilan de la conduite des tests et évaluer l'efficacité des actions menées pour déterminer les axes d'amélioration	▪ Sprint rétrospective	▪ Comités de suivi des tests
Faire le bilan de la conduite des tests avec l'ensemble des acteurs et déterminer les axes d'amélioration	▪ Rétrospective de l'incrément	▪ Comités de suivi des tests

**Nota : Pour l'agilité, la gestion et le suivi des tests peuvent être inclus dans des cérémonies standards de l'agilité à l'échelle. Dans ce cas, les informations présentant la cérémonie, viennent en complément et ne couvre que la partie « Tests ».**

La description détaillée de cette gouvernance est présentée dans le processus « Gestion de programmes et projets » (Référentiel méthodologique SNUM).

## 6 Annexes

### 6.1 Rôles/Profils : Abréviations utilisées dans les RACI

Profil / Rôle	Abréviation	Organisation
Equipe de développement (*)	Fournisseur	Ministère de la Justice – MOE / Prestataire
Responsable Prestataire	RPrest	Prestataire
Release Train Engineer	RTE	Ministère de la Justice - MOE
Product Manager	PM	Ministère de la Justice - MOA
Chef de Projet MOE	CPMoe	Ministère de la Justice - MOE
Chef de Projet Test	CPT	Ministère de la Justice - MOE <i>(Ce rôle peut être porté par le CP MOE)</i>
Leader Squad Intégration	LSI	Ministère de la Justice - MOE
Testeur	Test	Ministère de la Justice - MOE
Business analyste (ou analyste fonctionnel)	BA	Ministère de la Justice - MOE
Intégrateur / testeur système	INT	Ministère de la Justice – MOE
Scrummaster	SCM	Ministère de la Justice – MOE / Prestataire
Chef de Projet MOA	CPMoa	Ministère de la Justice - MOA
Product Owner	PO	Ministère de la Justice - MOA
MOA (analyse et exécution)	MOA	Ministère de la Justice - MOA
Architecte Solution / Technique	Archi	Ministère de la Justice – MOE
Expert Performance	Perf	Ministère de la Justice – MOE
RSSI	RSSI	Ministère de la Justice – MOE/MOA

(\*) : équipe de développement : comprend l'intégralité des rôle et profils d'une Squad