

Nicolas
BUKOWSKI



Cyril
PIAUD
c open

**JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS**

COMMENT LA SYSTEM TEAM « TESTING » EST
DEVENUE UNE SOLUTION TEAM « QUALITÉ » ?



11 JUIN 2024
BEFFROI DE MONTROUGE



Fondamentaux du programme SI Travaux



JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS

Les outils du programme SI Travaux apportent un **gain de temps, confort** et **sécurité** pour plus de temps utile de production sur les chantiers effectués sur les voies.

Ce potentiel en termes de bénéfices passe par une évolution des outils et des pratiques :



Remplacement des dépêches papier et des appels téléphoniques par des **échanges digitalisés**



Remplacement de la nécessité de collationnement par le partage de **données digitalisées communes et fiables**



Mise en place **de bonnes pratiques adaptées** aux nouveaux outils pour tirer pleinement profit des gains

Le 365 Travaux s'inscrit dans la continuité du programme SI Travaux

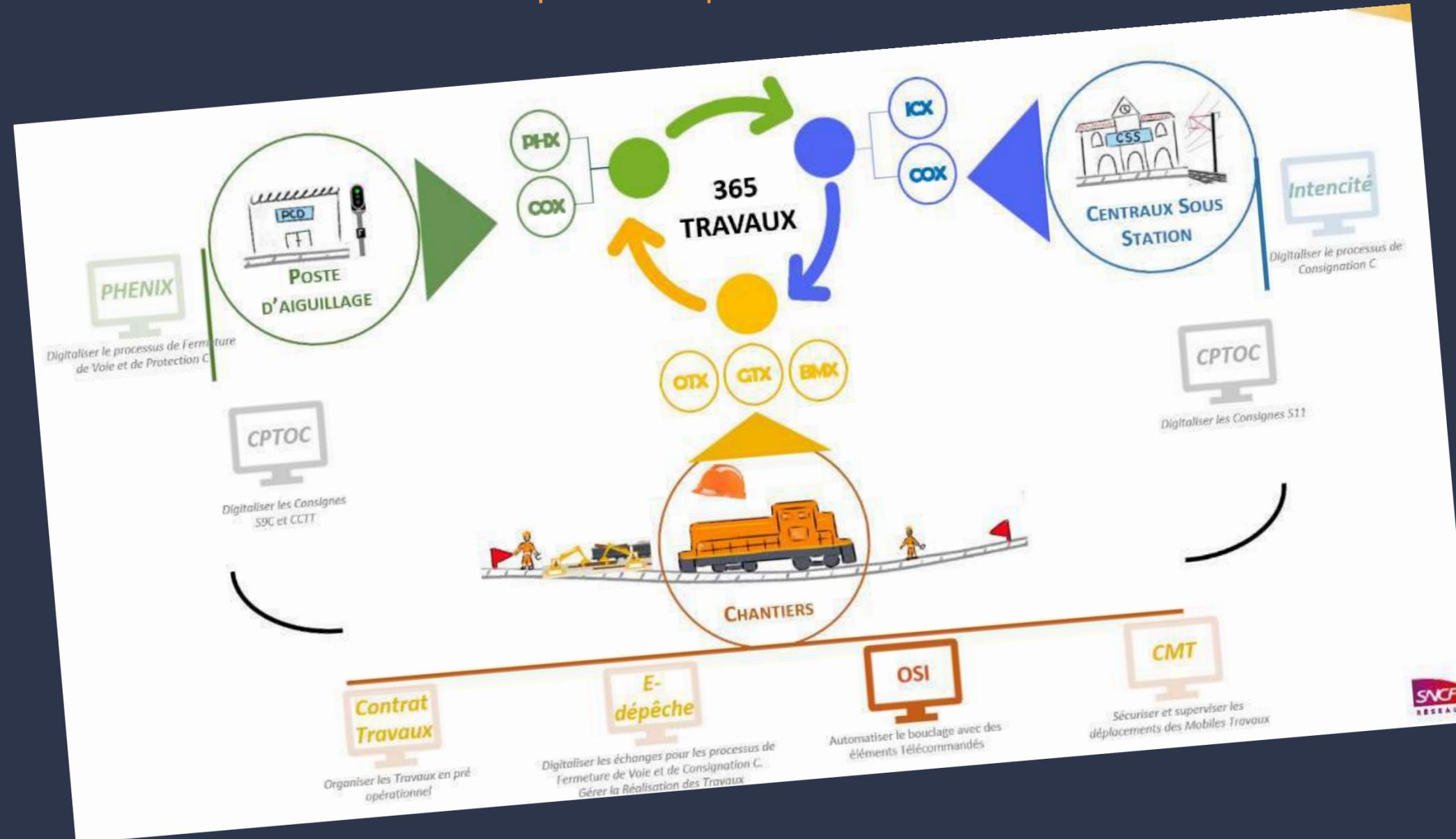
JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS

OBJECTIFS

- Mises en sécurité des chantiers accélérées
- Restitution rapide des chantiers pour favoriser la robustesse des circulations
- Désaturation et amélioration de la capacité des postes

Productivité

Maintenance
et Travaux



Vision système



User centric

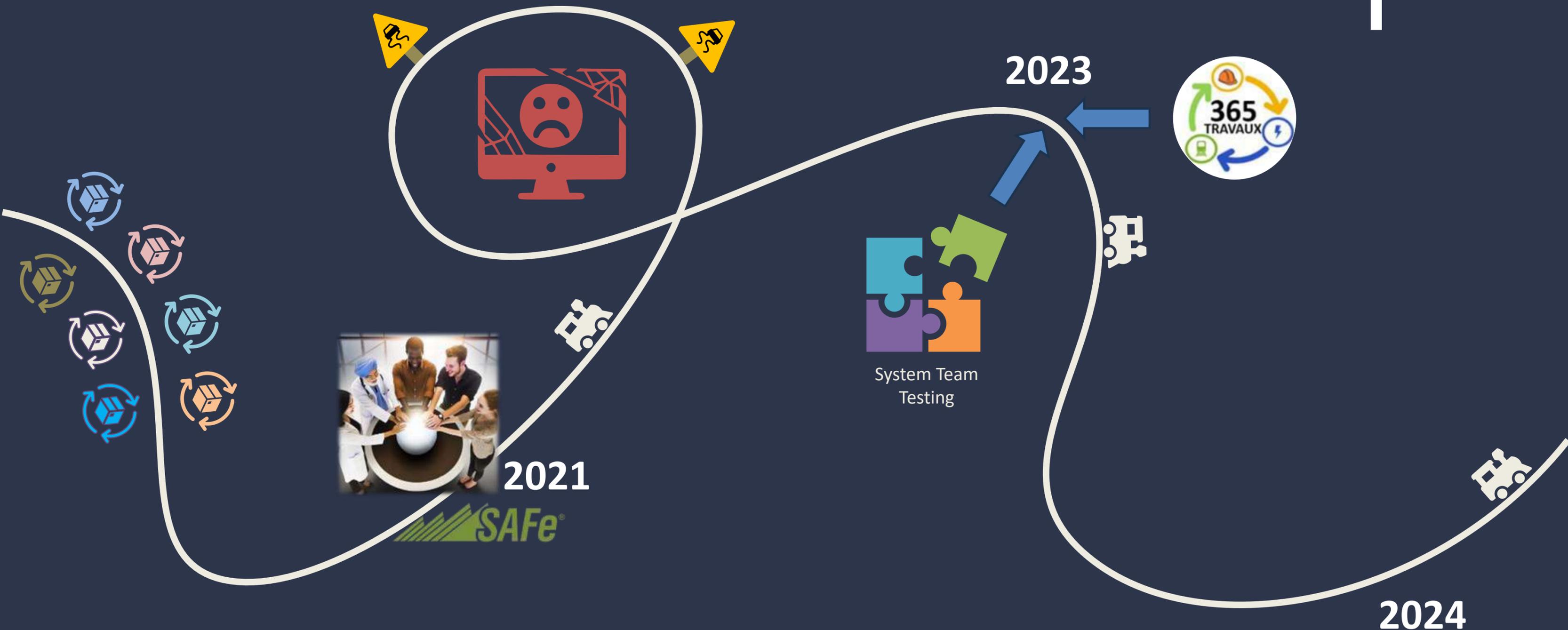


Modulaire

Complexité des tests bout en bout car processus global regroupant plusieurs processus métiers.

Problématiques initiales rencontrées

7 équipes Agile interdépendantes (environ 70 personnes)
Interconnexions entrantes et sortantes avec d'autres applications du SI SNCF Réseau

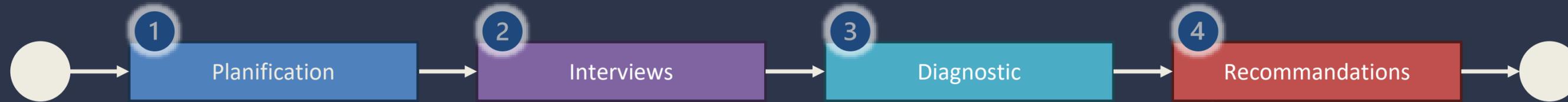


Maturation et construction d'une solution



2022

Etat des lieux des pratiques de test



- Organisation
- Processus
- Pratiques
- Outils



2023



APPROUVÉ



System Team

Mais d'abord, c'est quoi une System Team (au sens Safe) ?

System Team : Equipe Agile au service des autres Equipes du Train



Responsabilités Clés :

- Assurer **l'intégration continue**
- Soutenir les **environnements** de développement et de test
- Aider à coordonner le **processus de livraison**
- Fournir une **assistance technique**



Impacts attendus sur l'ART (Agile Release Train) :

- **Accélération** du processus de **livraison**
- Permet aux équipes de **se consacrer au développement**



N'empêche pas le **partage des responsabilités** avec les équipes agiles **pour une livraison de valeur de bout en bout**



UNE SYSTEM TEAM AU SEIN DU SI TRAVAUX

LE CONCEPT :

transverse
automatisation
support vision
equipe
solution
expertise pilotage
pluridisciplinaire

POURQUOI ?

Incarner,
Organiser et
Industrialiser la
vision globale de
la solution

POUR QUI ?

Toutes les
équipes projets
et la direction
de programme

QUELLES SOLUTIONS PEUT-ELLE APPORTER ?

Vision d'ensemble

- Pilotage global de la qualité du système
- Synchronisation des dépendances
- Gestion des adhérences
- Couverture des recommandations MSC

Désilotage des équipes

- Mutualisation des compétences
- Backupage des connaissances
- Homogénéisation des pratiques

- Intégration des solutions Industrielles
- Amélioration continue
- Suivi de la satisfaction utilisateur

Activités dévolues

- Pilotage global du Testing
- Recette de bout en bout
- Management des tests
- Gestion des adhérences (Internes et Externes)
- Automatisation des tests

Mesures du succès

- Respect de la trajectoire
- Baisse du nombre d'anomalie
- Baisse des régressions
- Amélioration de la qualité (produite et perçue)

Raisons d'échecs

- Résistance au changement
- Difficulté d'intégration aux équipes en place
- Sous dimensionnement de la System Team (ST)
- Manque de polyvalence des membres de la ST

MEMBRES DE LA SYSTEM TEAM SI TRAVAUX



LES ETAPES POUR LA MISE EN OEUVRE



Fonctionnement au quotidien de la System Team



Cérémonies

- Daily Programme
- Daily Equipe
- Démo Dev
- PI Planning



Activités

- Validation des US (Grooming)
- Rédaction et exécution des tests (fonctionnels et métiers)
- Conception des jeux de données
- Validation environnements post-déploiement (tests de recevabilité)
- Reporting (Indicateurs, Avancement, Anomalie, bilan de test)
- Contrôle du SONAR
- Automatisation



Organisation (10 personnes)

- Scrum Master (Test Manager)
- Testeurs
- Intégrateur
- Automaticien
- Jeux de données (Gestionnaire)
- Expert

Fondamentaux

Définition d'une politique de test, d'une stratégie de test basée sur la gestion des risques et d'une stratégie d'automatisation pour l'ensemble du programme qui s'inscrit dans les pratiques de l'usine logicielle de la SNCF.



1. Politique de test

2. Stratégie de tests (basée sur la gestion des risques)
3. Processus de test
4. Automatisation des tests



Fondamentaux

Définition d'une politique de test, d'une stratégie de test basée sur la gestion des risques et d'une stratégie d'automatisation pour l'ensemble du programme qui s'inscrit dans les pratiques de l'usine logicielle de la SNCF.

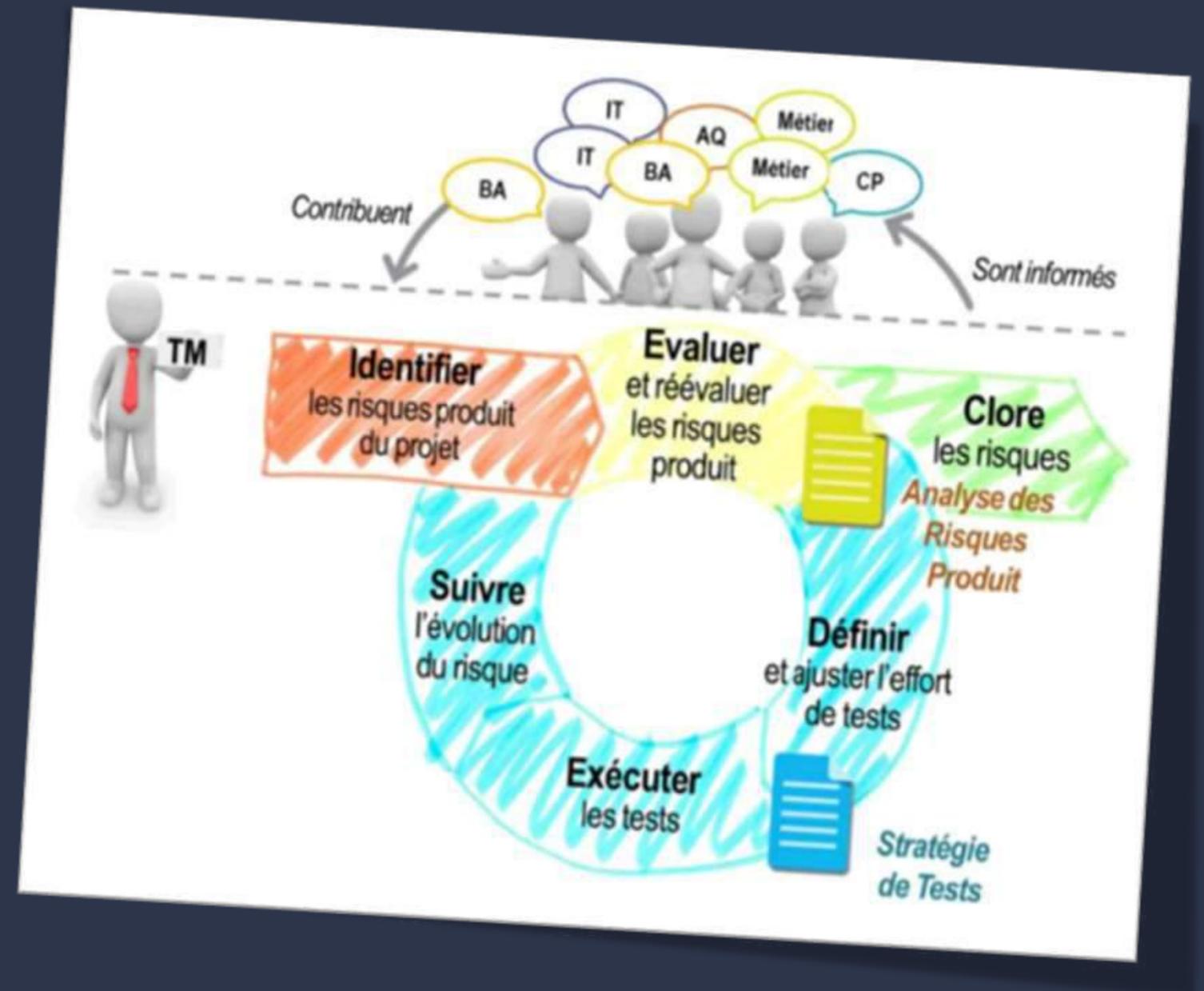
1. Politique de test



2. **Stratégie de tests** (basée sur la gestion des risques)

3. Processus de test

4. Automatisation des tests



Fondamentaux

Définition d'une politique de test, d'une stratégie de test basée sur la gestion des risques et d'une stratégie d'automatisation pour l'ensemble du programme qui s'inscrit dans les pratiques de l'usine logicielle de la SNCF.

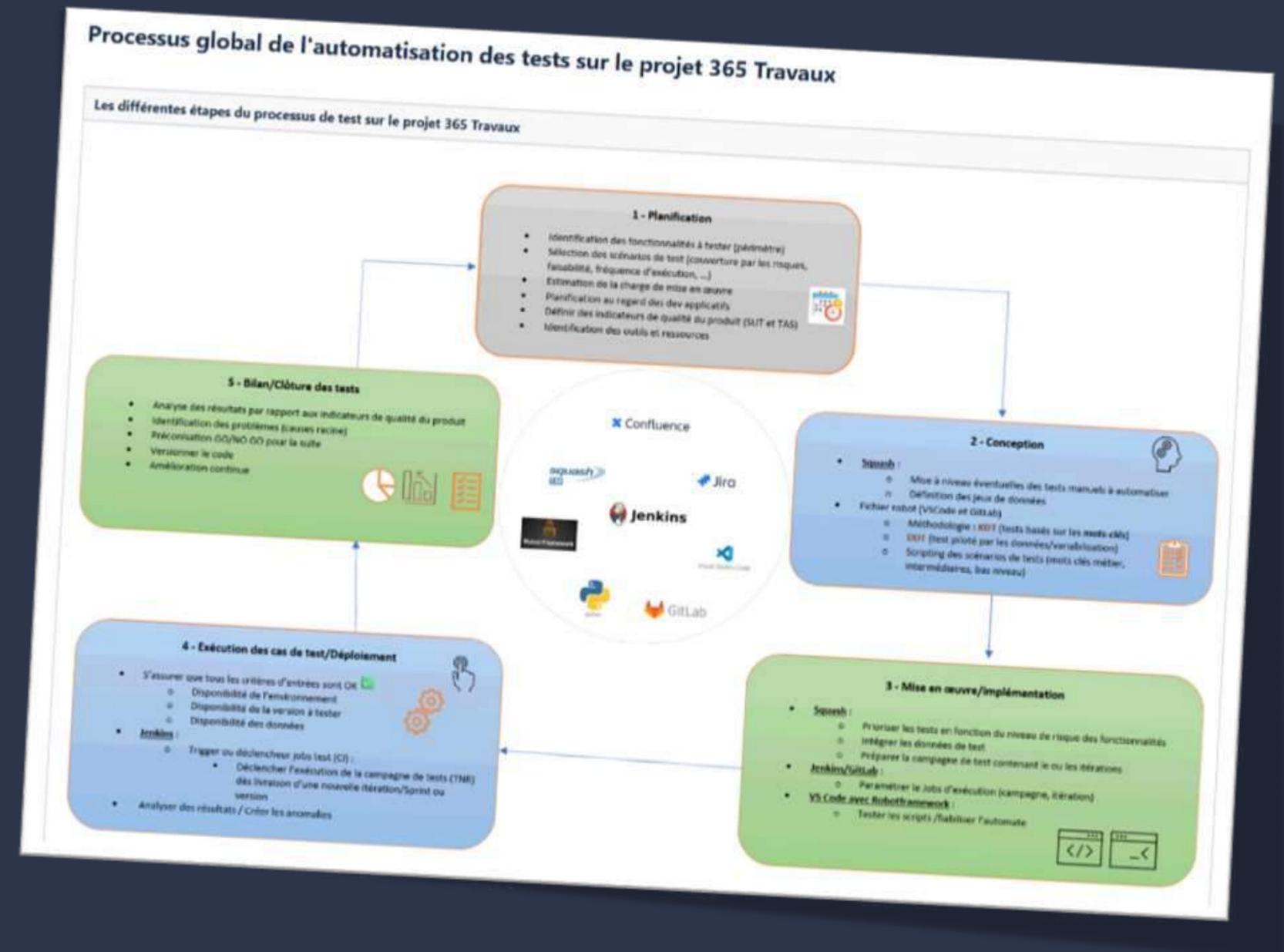
1. Politique de test
2. Stratégie de tests (basée sur la gestion des risques)
-  3. **Processus de test**
4. Automatisation des tests

Processus de test		
<i>Une campagne de test c'est beaucoup plus qu'exécuter des cas de test. C'est un processus qu'il faut penser, suivre et clôturer.</i>		
	Qui	Quand
Phase d'analyse des exigences		
<ul style="list-style-type: none">Analyse des risques.Analyse des User Stories.Discuter avec les différents acteurs des points d'incompréhension et d'imprécision.	Coordinateur de l'équipe Projet Testeur Fonctionnel / Testeur Mobile / Dev QA (QA) / Testeur Fonctionnel / Testeur Mobile / Dev	Cadrage sur de la conception préliminaire Après la phase de conception architecturale et de développement Cadrage des cas de test sur les parties les plus critiques du développement
Phase de Planification		
<ul style="list-style-type: none">Evaluation de la capacité de test de l'équipeEstimation de l'effort de test pour chaque USAffectation des US à tester.Répartition des tâches récurrentes de test (Analyse Sonar, Revue par les pairs...)	Equipe (Scrum Team) Testeur Fonctionnel, Testeur Mobile [Scrum/Agile Team] / QA [Scrum/Agile Team] / QA	Examen (Scrum Planning) Test Lors de la rétrospective et d'Adaptage (Scruming) Scrum Planning Test Examen de Scrum Planning Test
Phase de conception des tests :	Qui	Quand

Fondamentaux

Définition d'une politique de test, d'une stratégie de test basée sur la gestion des risques et d'une stratégie d'automatisation pour l'ensemble du programme qui s'inscrit dans les pratiques de l'usine logicielle de la SNCF.

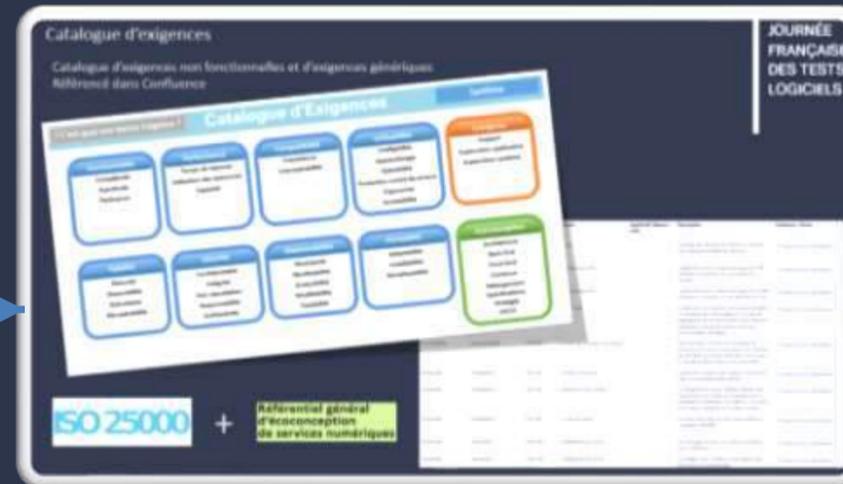
1. Politique de test
2. Stratégie de tests (basée sur la gestion des risques)
3. Processus de test
4. Automatisation des tests



Gestion des risques

Comment l'analyse des risques a permis de définir une approche de test

Catalogue d'exigences



Exemples :

- **Utilisabilité** – Tri des listes
- **Compatibilité** – Standard contrats d'interfaces
- **Performance** – Temps affichage page web
- **Sécurité** – Principe de moindre privilège

Priorisation des risques



Approche de mitigation des risques



Catalogue d'exigences

Catalogue d'exigences non fonctionnelles et d'exigences génériques
Référéncé dans Confluence

JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS



ISO 25000

+

Référentiel général
d'écoconception
de services numériques

Libellé	Applicatif/ Réseau / Infra	Description	Précisions / Notes		
		L'usage des données est prévu et conforme aux exigences de confidentialité.	> Cliquez ici pour développer...		
		L'application est en mesure de supporter 200 utilisateurs simultanés sur une période de 3 minutes.	> Cliquez ici pour développer...		
		L'application est en mesure de supporter 20 000 utilisateurs connectés sur une période de 30 minutes.	> Cliquez ici pour développer...		
		L'application est amenée à fonctionner pendant 10 semaines sans être arrêtée il n'y a pas de dégradation des fonctionnalités (hors maintenance, dégradation des performances, perte de fonctionnalités, dérapage).	> Cliquez ici pour développer...		
Compatibilité	Interopérabilité	Coln-001	Format des échanges atomiques	Les échanges entre les données et les services atomiques sont réalisés via API REST au format JSON selon les normes et standards définis dans le document 001	> Cliquez ici pour développer...
Utilisabilité	Intelligibilité	Utin-001	Facilité d'intégration	L'application propose des moyens d'intégration clairs et compréhensibles (UX/UI)	> Cliquez ici pour développer...
Utilisabilité	Intelligibilité	Utin-002	Adaptation du vocabulaire	Les désignations (noms, étiquettes, libellés) dans l'application sont claires et adaptées pour la population d'utilisateurs. Par ailleurs, une notation spécifique désigne de manière unifiée.	> Cliquez ici pour développer...
Utilisabilité	Intelligibilité	Utin-003	Et dans les listes	Les listes sont triées sur des critères définis en conception détaillée.	> Cliquez ici pour développer...
Utilisabilité	Opérabilité	Utop-001	Intelligibilité de l'erreur	Les messages d'erreur sont clairs et explicites pour l'utilisateur.	> Cliquez ici pour développer...
Utilisabilité	Opérabilité	Utop-002	Intelligibilité de l'erreur	Les libellés sont validés et externalisés dans des fichiers de paramètres.	> Cliquez ici pour développer...

Priorisation des risques

Analyse de risques

Aide			Impacts Métier (Imet)				Probabilité de réalisation du risque (Preal)					Niveau de risque
ID	Type de risque	Description du Risque	Criticité	Visibilité	Fréquence	Coût de la défaillance	Imet Total	Complexité	Nouveauté	Nombre de défauts	Preal Total	Global
EDI-01	Sécurité	...	Majeur	Majeur	Modéré	Majeur	11	Moyenne	Moyenne	Basse	5	55
EDI-02	Fiabilité	...	Modéré	Modéré	Modéré	Majeur	9	Haute	Moyenne	Basse	6	54
EDI-03	Maintenabilité	...	Modéré	Modéré	Modéré	Modéré	8	Moyenne	Haute	Basse	6	48
EDI-04	Utilisabilité	...	Modéré	Majeur	Modéré	Modéré	9	Moyenne	Moyenne	Basse	5	45
EDI-05	Performance	...	Modéré	Modéré	Modéré	Non Significatif	7	Moyenne	Haute	Basse	6	42
EDI-06	Fonctionnel/Métier	...	Majeur	Majeur	Modéré	Modéré	10	Basse	Moyenne	Basse	4	40
EDI-07	Fiabilité	...	Non Significatif	Modéré	Majeur	Non Significatif	7	Moyenne	Moyenne	Basse	5	35
EDI-08	Compatibilité	...	Modéré	Non Significatif	Non Significatif	Modéré	6	Moyenne	Moyenne	Basse	5	30
EDI-09	Fonctionnel/Métier	...	Majeur	Majeur	Modéré	Modéré	10	Basse	Basse	Basse	3	30
EDI-10	Sécurité	...	Majeur	Majeur	Non Significatif	Majeur	10	Basse	Basse	Basse	3	30
EDI-11	Fonctionnel/Métier	...	Majeur	Majeur	Non Significatif	Modéré	9	Basse	Basse	Basse	3	27
EDI-12	Utilisabilité	...	Modéré	Modéré	Modéré	Modéré	8	Basse	Basse	Basse	3	24
EDI-13	Fonctionnel/Métier	...	Modéré	Modéré	Majeur	Non Significatif	8	Basse	Basse	Basse	3	24



Bénéfices de la démarche :

- Anomalies prises en compte sans débat
- Partage et acceptation par tous
- Evite la surcharge et la redondance dans les US
- S'applique à l'existant et aux futures évolutions

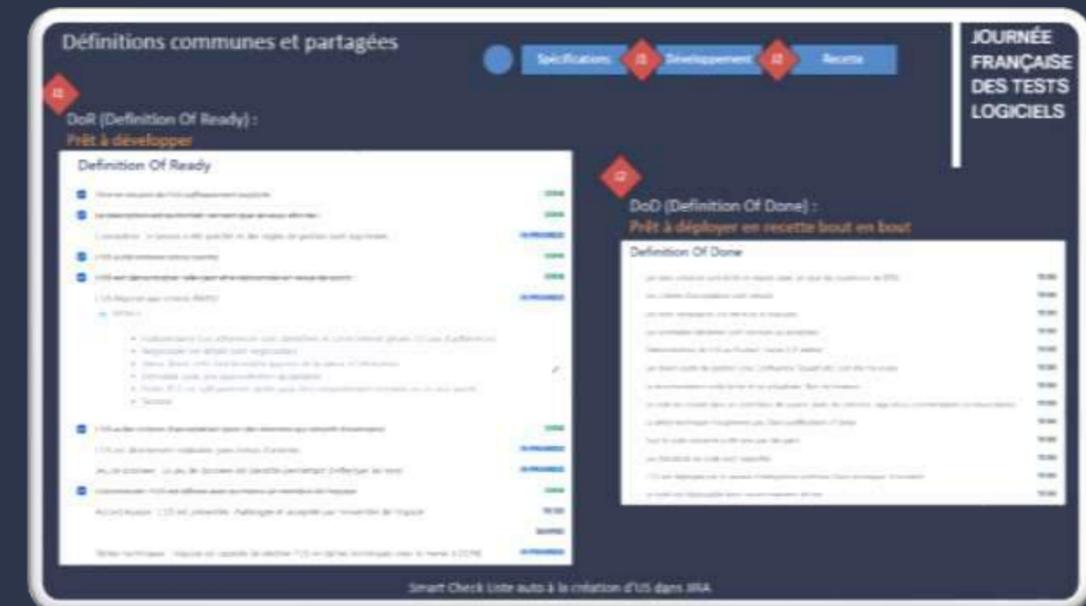


Si l'exigence de respect du tri alphabétique des listes déroulantes n'est pas respectée, cela donne lieu à la déclaration d'une anomalie

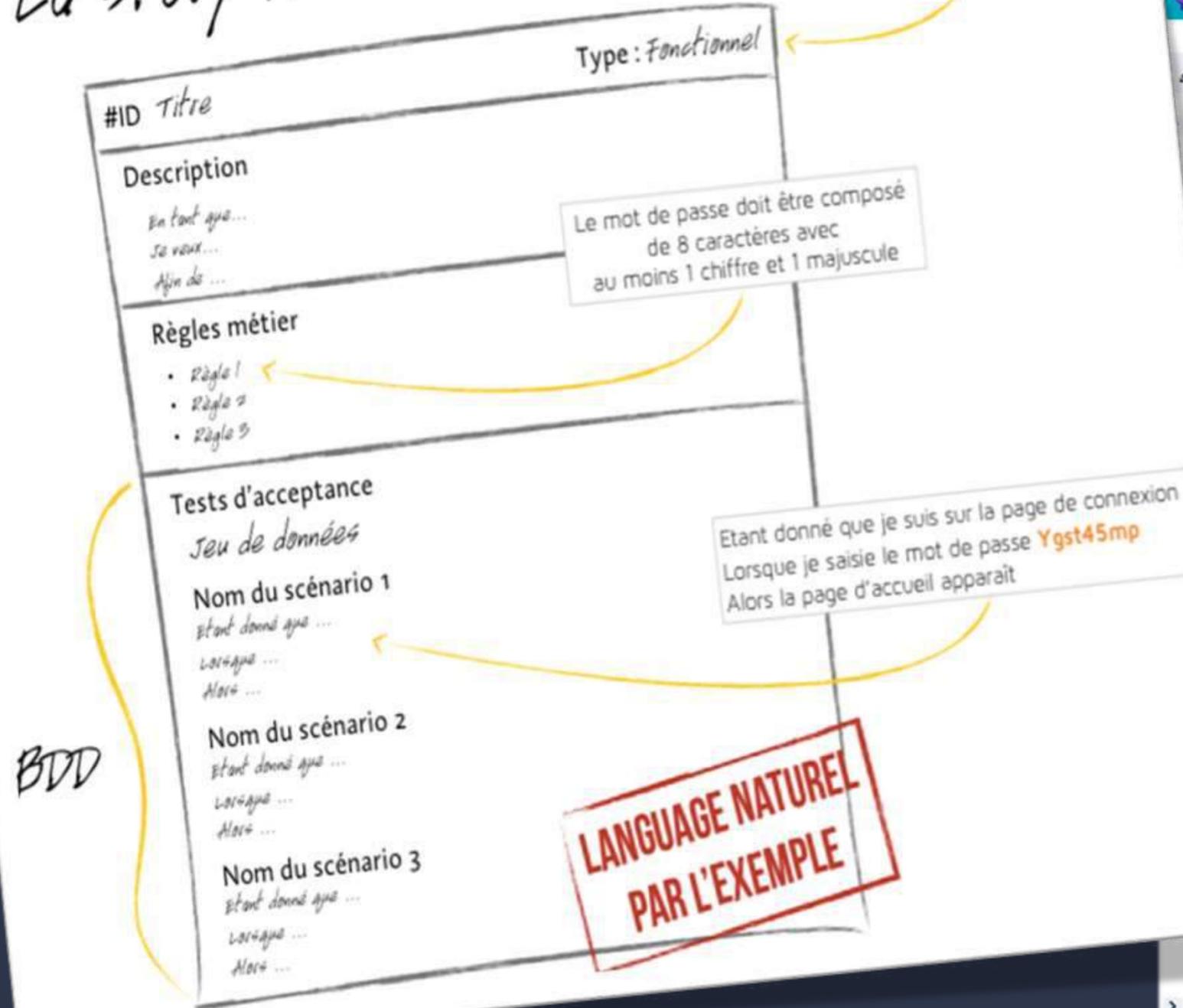
Industrialisation de la Spécification des User Story



Industrialisation de la définition : - Prêt à développer - Prêt à déployer



La story A4



Modifier | Ajouter un commentaire | Attribuer | Plus | Ouverte

Informations

Type: Story
Priorité: Moyenne
Affecte la/les version(s): Aucune
Composants: Aucune
Étiquettes: Aucune
Template: [lien]

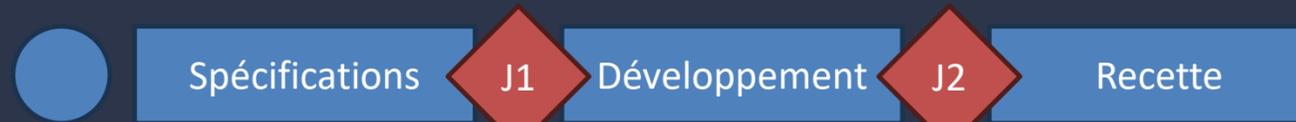
Écrit Utilisateur:
- En tant que [utilisateur/système]
Je souhaite [objectif]
Afin de [bénéfice]

Critères d'acceptation:
- TVX2-xxx-CA-xxx:
Etant donné que [situation initiale, contexte]
Lorsque [les actions à réaliser]
Alors [le résultat attendu]
...

Description
Description :
Métier :
xx-RG-xxx:
être non ambiguës, bien définies, complètes et précises.
le doit être concise et simple (pas de si, alors, sinon dans une seule et même règle métier)
s verbes modaux inutiles : le système « peut », « doit », « pourrait »
plutôt la forme active que passive

> Smart Checklist

Définitions communes et partagées



J1

DoR (Definition Of Ready) :
Prêt à développer

Definition Of Ready

- Titre et résumé de l'US suffisamment explicite DONE
- La description est au format : en tant que, je veux, afin de... DONE
- Conception : le besoin a été spécifié et des règles de gestion sont exprimées IN PROGRESS
- L'US a été estimée (story points) DONE
- L'US est démontrable : elle peut être démontrée en revue de sprint DONE
- L'US Répond aux critères INVEST IN PROGRESS
 - DETAILS
 - Indépendante (Les adhérences sont identifiées et correctement gérées OU pas d'adhérences)
 - Négociable (les détails sont négociables)
 - Valeur (livrer cette fonctionnalité apporte de la valeur à l'entreprise)
 - Estimable (avec une approximation acceptable)
 - Petite (l'US est suffisamment petite pour être complètement terminée en un seul sprint)
 - Testable
- L'US a des critères d'acceptation (avec des données qui servent d'exemples) DONE
- L'US est directement réalisable (sans temps d'attente) IN PROGRESS
- Jeu de données : un jeu de données est identifié permettant d'effectuer les tests IN PROGRESS
- Coconstruite : l'US est affinée avec au moins un membre de l'équipe DONE
- Accord équipe : L'US est présentée, challengée et acceptée par l'ensemble de l'équipe TO DO
- Des maquettes d'interface utilisateur sont réalisées (Uniquement pour IHM) SKIPPED
- Tâches techniques : l'équipe est capable de décliner l'US en tâches techniques pour la mener à DONE IN PROGRESS

J2

DoD (Definition Of Done) :
Prêt à déployer en recette bout en bout

Definition Of Done

- Les tests unitaires sont écrits et réussis (avec un taux de couverture de 80%) TO DO
- Les critères d'acceptation sont remplis TO DO
- Les tests nécessaires ont été écrits et exécutés TO DO
- Les anomalies déclarées sont résolues ou acceptées TO DO
- Démonstration de l'US au Product Owner (CP métier) TO DO
- Les divers outils de gestion (Jira, Confluence, Squash etc.) ont été mis à jour TO DO
- La documentation a été écrite et ou actualisée : Bon de livraison TO DO
- Le code est stocké dans un contrôleur de source :(avec les commits, tags et/ou commentaires correspondants) TO DO
- La dette technique n'augmente pas (Sans justification) cf Sonar TO DO
- Tout le code concerné a été revu par des pairs TO DO
- Les standards de code sont respectés TO DO
- L'US est déployée par le serveur d'intégration continue (Sans provoquer d'incident) TO DO
- Le build est déployable dans l'environnement de test TO DO



Bénéfices de la démarche :

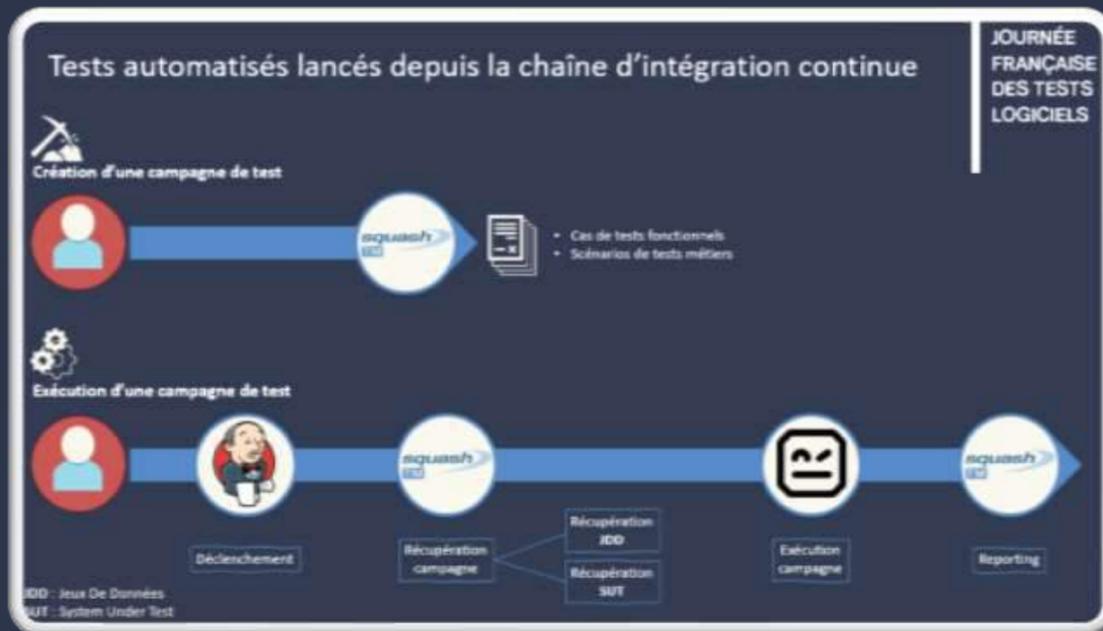
- Standardisation des expressions de besoin
- Complétude des exigences
- Capacité à tester
- Facilite l'évaluation de la complexité



L'obligation de respect du formalisme En tant que, Je veux, Afin de, conduit les PO à se questionner sur l'objectif précis de la fonctionnalité

L'ingénierie du DevTestOps

Automatisation des tests



Stratégie d'utilisation des environnements



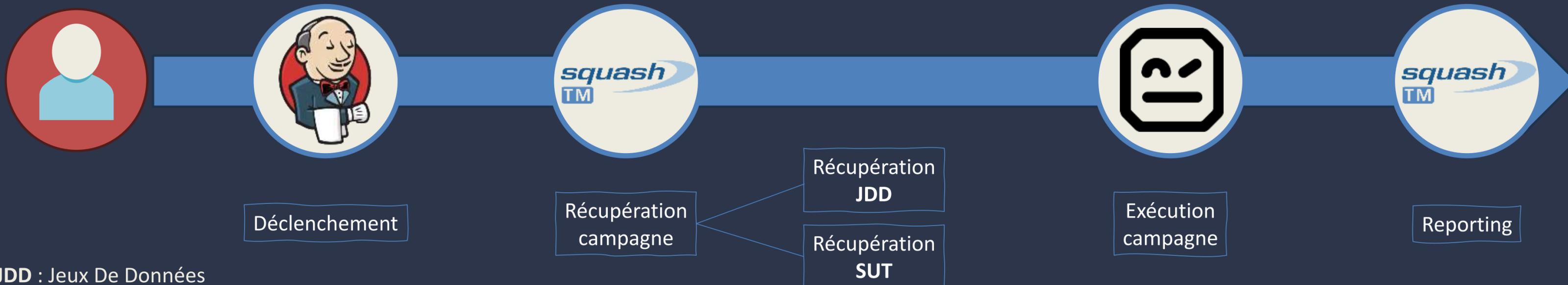
Tests automatisés lancés depuis la chaîne d'intégration continue



Création d'une campagne de test



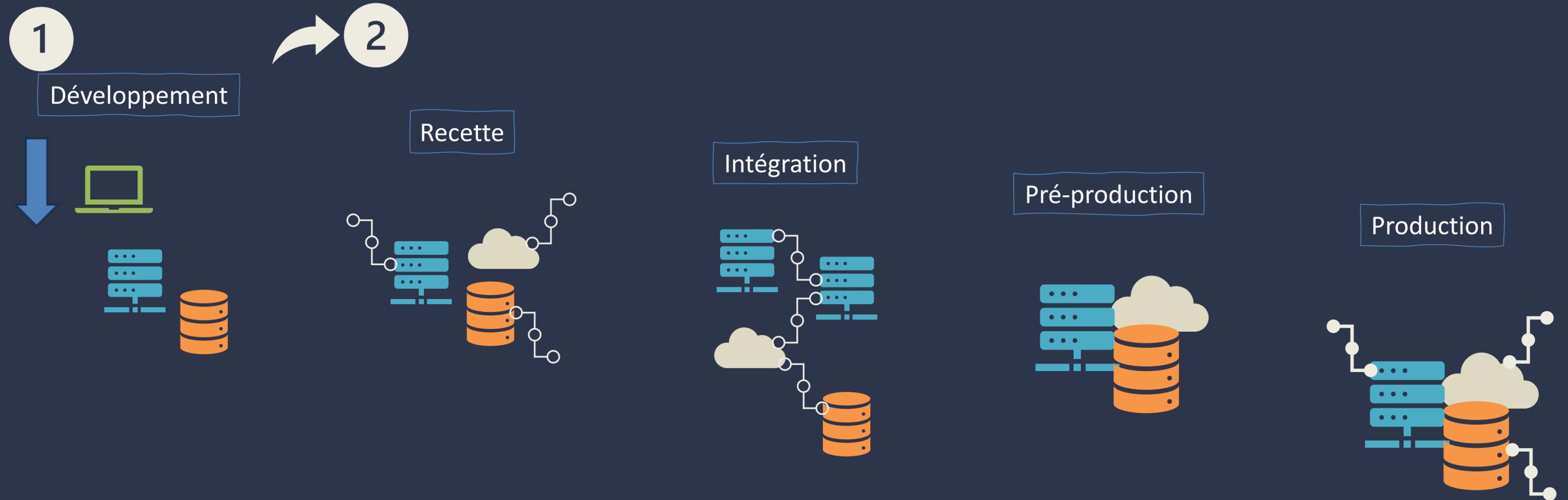
Exécution d'une campagne de test



JDD : Jeux De Données
SUT : System Under Test

Maitrise des environnements de test

Avoir des **intégrateurs dans l'équipe** permet une **meilleure communication**, un suivi et la **prise en compte des besoins** du projet et des **exigences de test** par les équipes Infra





Bénéfices de la démarche :

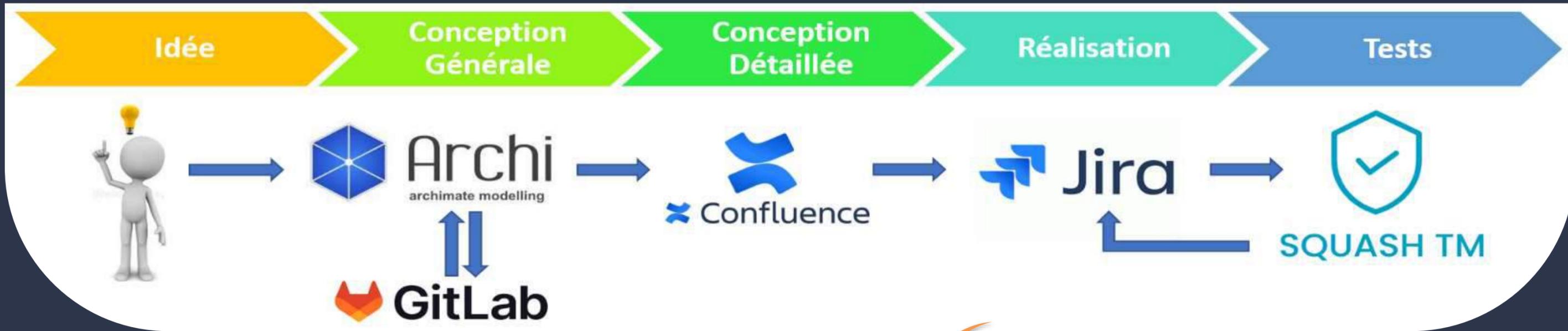
- Gain de temps sur les déploiements (recevabilité)
- Les environnements répondent aux exigences de test
- La System Team maîtrise le contenu des environnements de test
- Maîtrise de jeux de données cohérents
- Amélioration de la couverture de test



L'apport des échanges entre développeurs, testeurs et opérationnels apporte une compréhension mutuelle des besoins et contraintes de chacun. Le focus est mis sur l'environnement de test et le service qu'il doit rendre.

Un outillage cohérent

La chaîne outillée et industrialisée porte le processus





Bénéfices de la démarche :

- Fluidité dans la chaîne de production
- Facilité de mise en place d'indicateurs basés sur des données partagées
- Chaque étape du processus est cadrée par les outils
- Processus permettant une homogénéisation de ce qui est produit
- Chaîne intégrée dans l'usine logicielle (Préconisations SNCF)



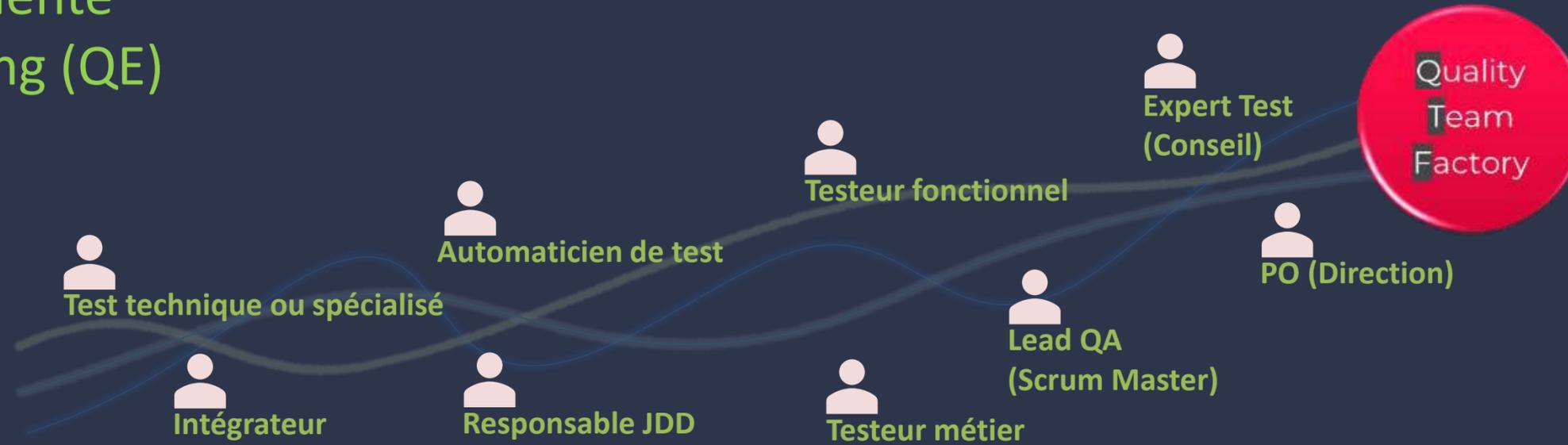
La System Team Testing s'appuie sur l'usine logicielle pour construire une chaîne outillée. Elle bénéficie de son support et apporte également une expérience terrain que l'usine logicielle s'approprie pour un large partage

Points de vigilance et difficultés rencontrées

-  ➤ Résistance au **changement**
-  ➤ Le respect des **règles** partagées
-  ➤ **Pilotage**
-  ➤ **Engagement** de la direction
-  ➤ **Reporting**
-  ➤ **Responsabilités** de la System Team
-  ➤ Gestion de la **connaissance**
-  ➤ **Directives** contradictoires

En quoi la System Team Safe est devenue une Quality Team Factory

Un dispositif orienté Quality Engineering (QE)



Convictions Open

- Fournisseur de solutions en support des Feature Teams et de la direction de programme
- Efficacité du dispositif immédiatement perçue
- Permet au Test d'avoir une voix et d'être intelligible et entendu
- Légitimité
- Modèle adaptable à tous types d'organisation



Constats SNCF

- Formalisation des processus
- Harmonisation des processus
- Capitalisation
- Transversalité
- Réactivité et adaptabilité



Le dispositif est cité dans les 4 points positifs du PI suivant sa mise en place.

Questions / Réponses

JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS



Nicolas
BUKOWSKI



Cyril
PIAUD
open

**JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS**

COMMENT LA SYSTEM TEAM « TESTING » EST
DEVENUE UNE SOLUTION TEAM « QUALITÉ » ?



**MERCI DE VOTRE ÉCOUTE
N'OUBLIEZ PAS DE VOTER**



JOURNÉE
FRANÇAISE
DES TESTS
LOGICIELS

VOTEZ POUR LA
MEILLEURE
PRÉSENTATION

