



🕒 14 Septembre 2021

13^{ème} édition

JOURNÉE FRANÇAISE
DES TESTS LOGICIELS

LE CHEMIN VERS L'AGILITÉ PASSE AUSSI PAR LE TEST DESIGN OUTILLÉ

RETOUR D'EXPÉRIENCE

**LE TEST LOGICIEL, MAÎTRISER L'ÉTAT
DE L'ART DU TEST AVEC LE CFTL !**



Agenda

Nous ne parlerons pas...



Nous verrons...

L'histoire d'une jeune équipe de test qui doit faire évoluer ses pratiques vers du test agile dans un contexte où le mot « **Agile** » peut être « **tabou** »

ADIS : Qui sommes-nous ?

Société de courtage d'assurances créée en 1980

Filiale d'Axa France

**Centre de gestion historique de l'association
AGIPI**

+ Gestion de contrats pour ANPERE

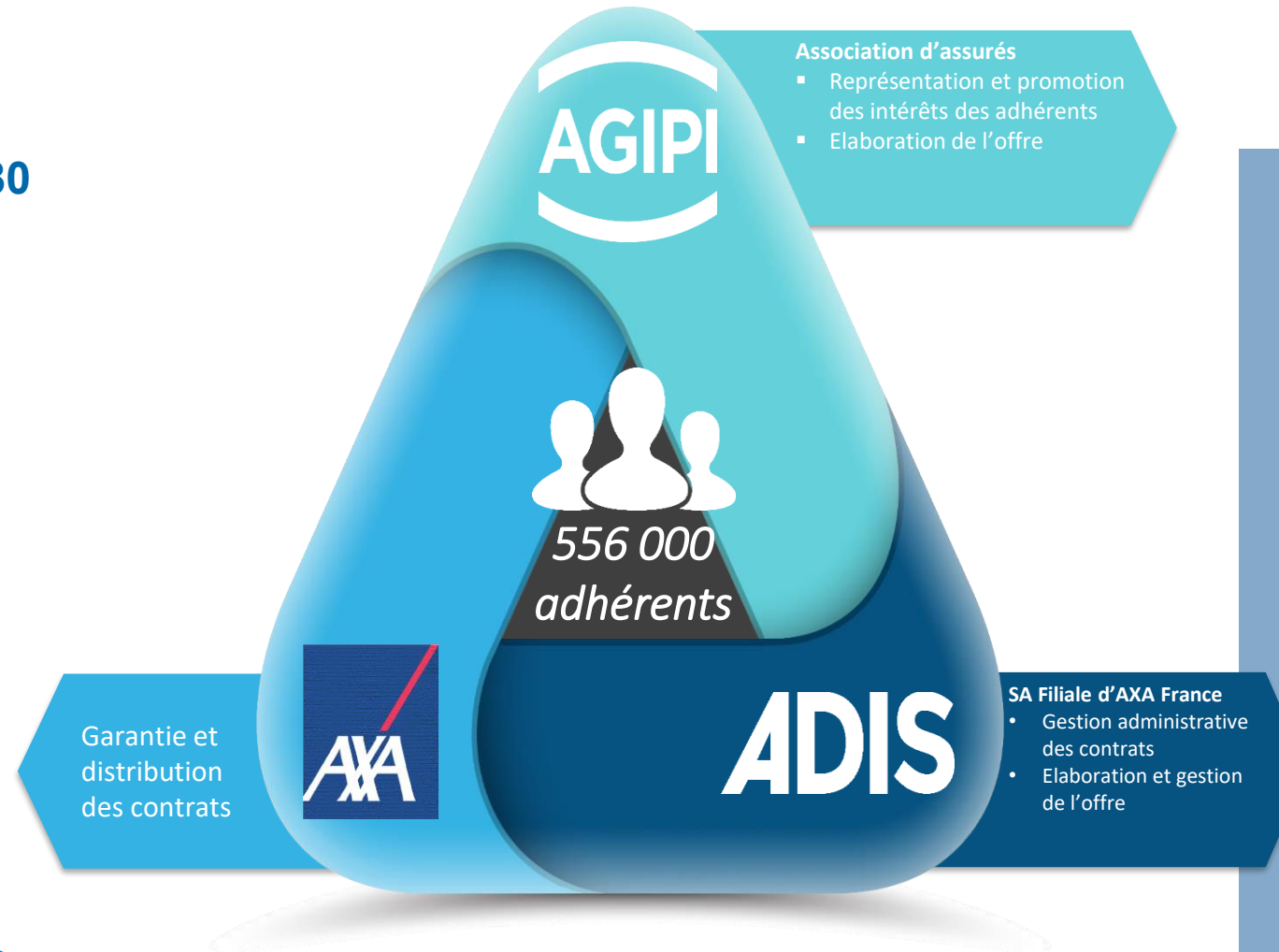
Experts en protection sociale

- Elaboration et gestion de contrats d'assurance de personnes (Epargne, Retraite, Prévoyance et santé)
- Services et relations personnalisés

40 ans
d'expérience

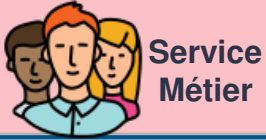
600+
collaborateurs

**Près de
850 000**
adhésions
gérées



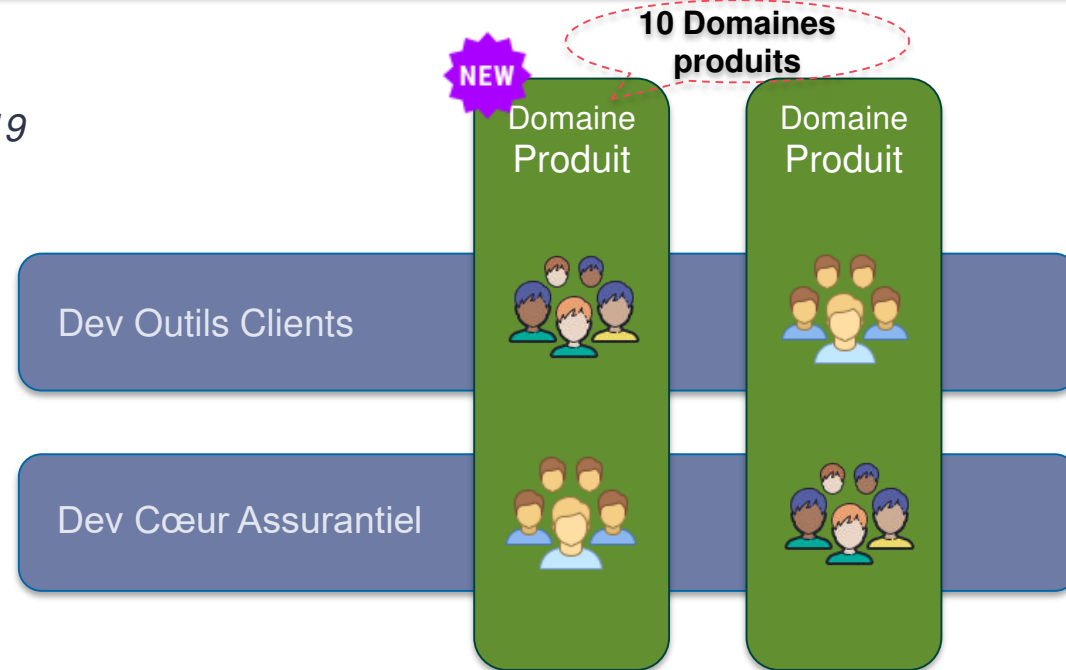
Contexte

Métiers



DSI

Dès 2019

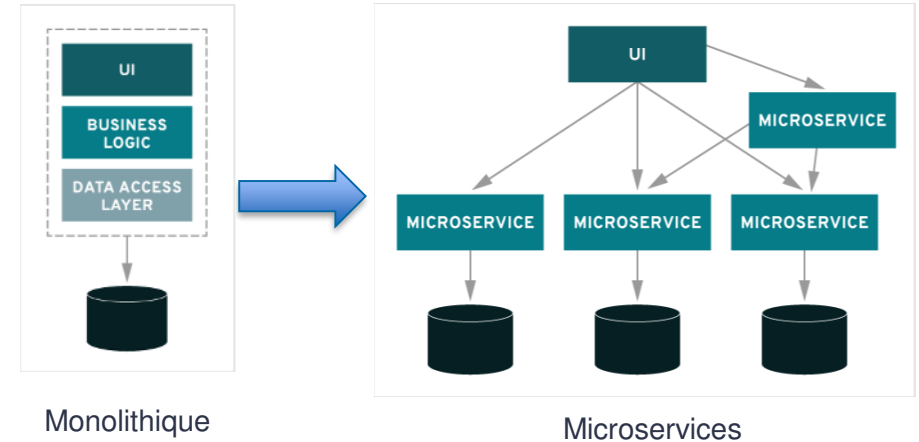


Exploitation  Ingénieur(e)s Ops

SSI  Ingénieur(e)s SSI

GPA  Equipe Test  UX Designer  Urbaniste SI  Qualité de Service IT 

Evolutions des architectures applicatives



Accélération du rythme des développements / livraisons



Passage progressif à des approches de développements itératives

Contexte

Problématiques du test

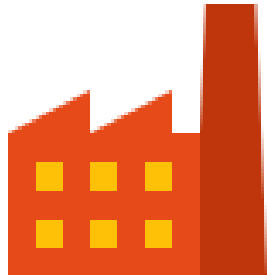
Rythme des développements



Test



Développement



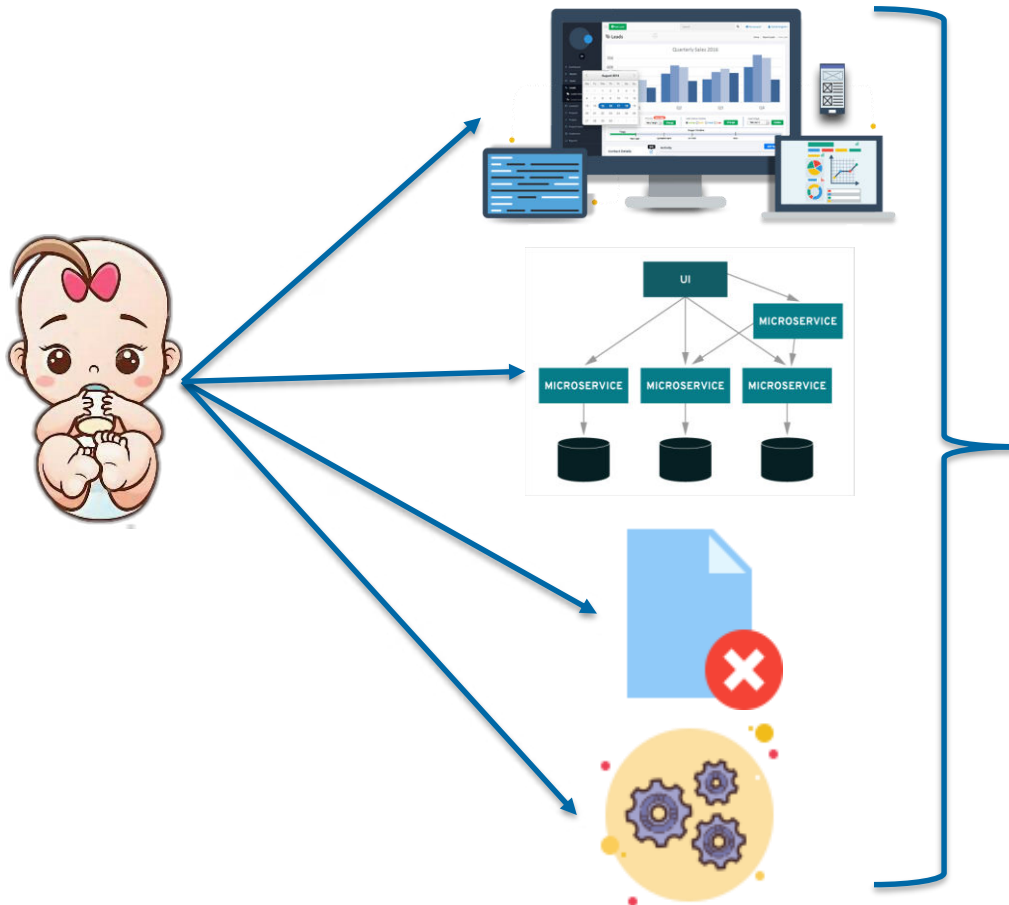
Production



Comment tester durant le sprint ?

Contexte

Problématiques du test



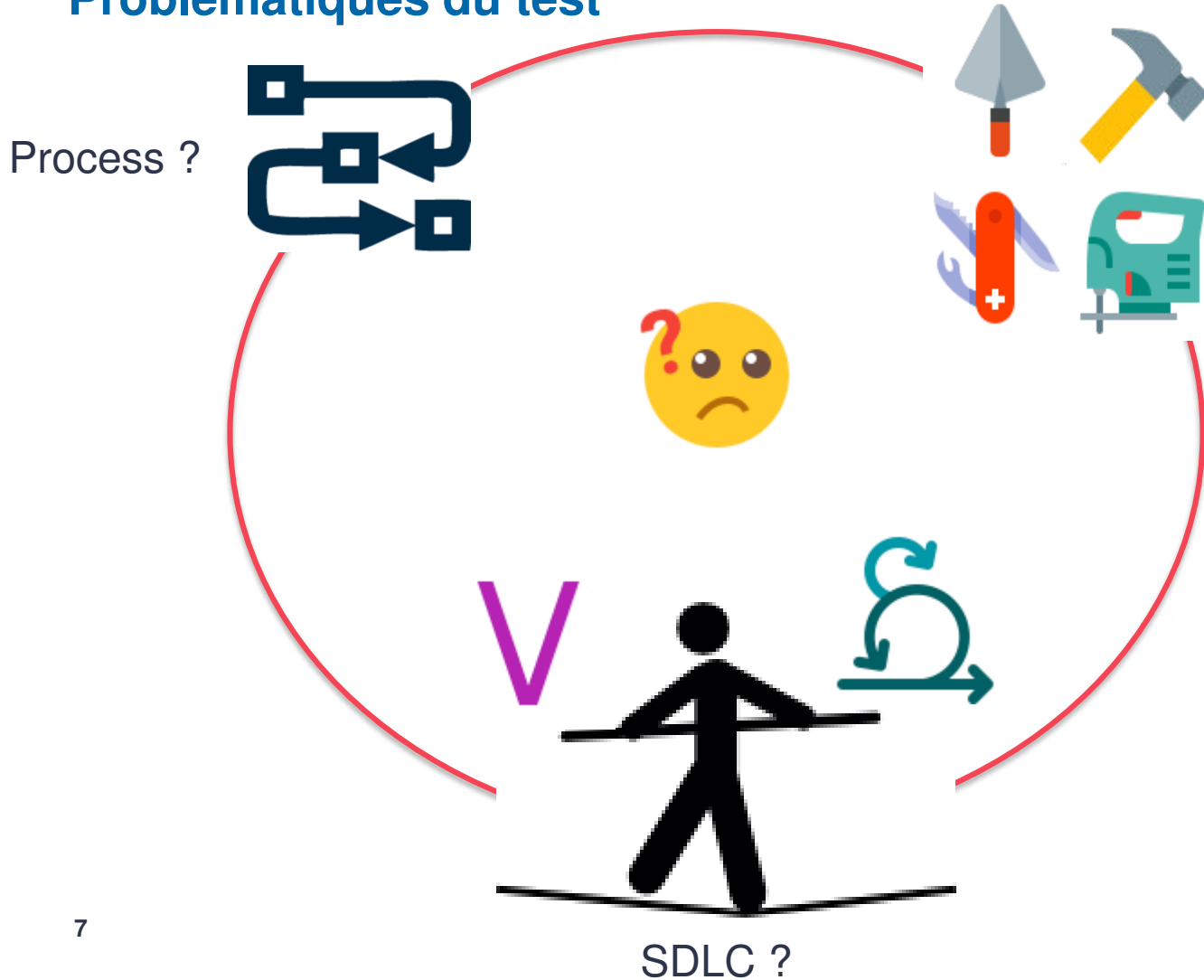
Equipe de test jeune, sans connaissance du métier, devant faire face:

- À un SI de 100+ systèmes applicatifs,
- À 1 domaine produit = plusieurs systèmes applicatifs,
- À l'évolution de l'architecture du SI (passage aux microservices),
- Sans ou avec peu de documentation existante,
- À Des spécifications / besoins pas toujours « lisibles »,
- À des pratiques de développement en pleine mutation, mais trainant encore de vieux « réflexes »,
- À l'absence de processus de test en place,
- Au peu d'outillages des tests



Contexte

Problématiques du test



Outils ?

Quels outils et quels process dans un contexte d'instabilité dans le cycle de vie de développement logiciel ?

(cohabitation de pratiques issues de l'agilité et des modèles traditionnels)

Réponse 1 – Organisationnelle

Métiers



Service
Métier



Service
Métier

DSI

Domaine
Produit

Domaine
Produit

Dev Outils Clients

Dev Cœur Assurantiel

Equipe Test

Exploitation



Ingénieur(e)s Ops

SSI



Ingénieur(e)s SSI

GPA



Equipe
Test



UX
Designer



Urbaniste
SI



Qualité de
Service IT

Intégration directe du testeur à un domaine produit



- Favoriser l'échange, l'interaction et la collaboration avec les développeurs
- Insuffler la démarche de test à l'équipe, responsabiliser l'équipe quant à la qualité



- Si les bonnes pratiques ne sont pas en place, le testeur se retrouve rapidement dépassé

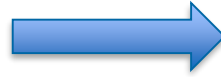
Réponse 1 – Organisationnelle

Constats



→ *Si les bonnes pratiques ne sont pas en place, le testeur se retrouve rapidement dépassé*

- *Silos entre développement et test encore très présents*
- *Le test / la qualité encore considéré comme « l'affaire » du testeur*



Aujourd'hui

Evolution du **Mindset** indispensable:

→ ***Rattachement opérationnel du testeur au responsable d'équipe du domaine produit:***

- ❖ *L'équipe devient garante à la fois des développements et des tests sur son domaine*
- ❖ *La qualité devient la responsabilité collective de l'équipe*

Réponse 2 – Méthodologie outillée



- *Comment tester pendant le sprint ? Accélérer nos activités de test ?*
- *Comment collaborer efficacement quand il y a encore des silos entre Métiers et DSI ?*
- *Comment partager et faire valider nos tests? Palier l'absence de documentation?*



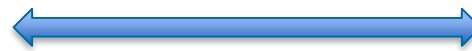
Tentatives avec la syntaxe Gherkin (Given ... When ... Then ...) → **Echec**

- *Pas de User Story*
- *Granularité des fonctionnalités pas assez fine*
- *Pas de collaboration en amont pour la définition des critères d'acceptation*
- *Pas d'automatisation des tests en place*

Déjà en place dans l'organisation



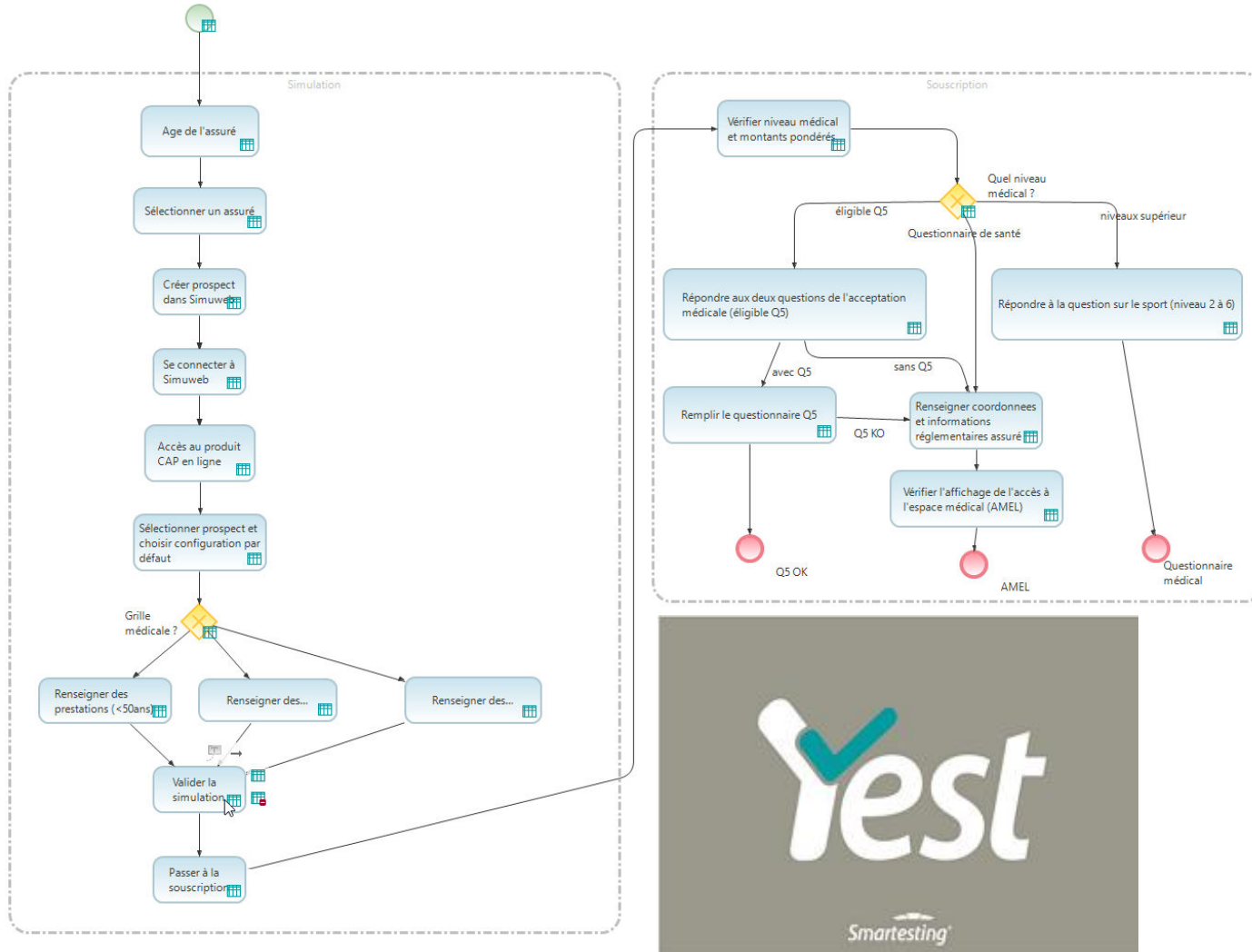
Gestion de projet et Suivi d'anomalies



Gestion des tests

Réponse 2 – Méthodologie outillée

Mise en place de Yest – Conception visuelle des tests

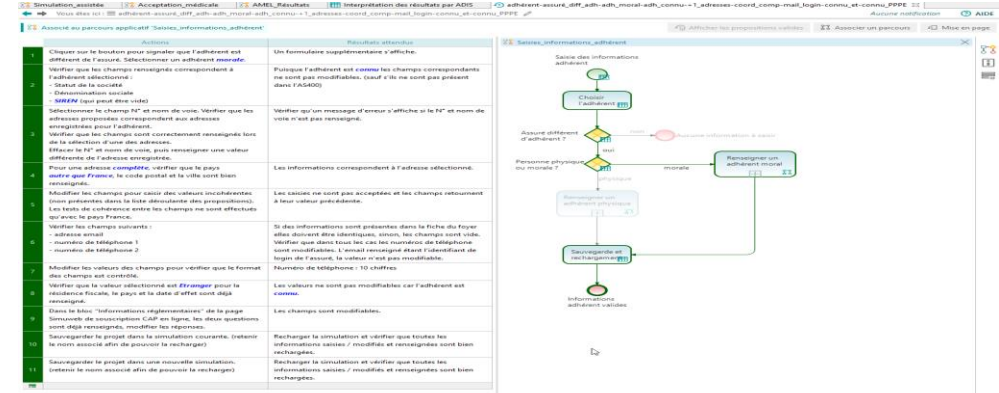
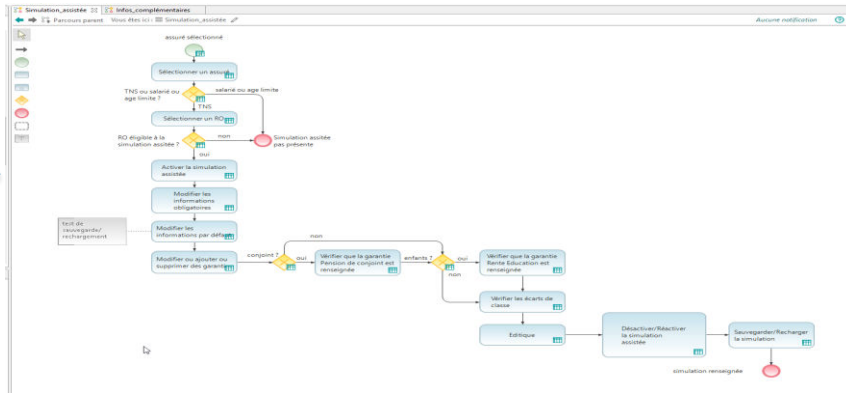
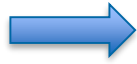


		Étapes de test		Direction suivante	Objectif
1	oui	1	Dans le projet AUT_label_projet, sélectionner AUT_sport pour la question : "Le client souhaite-t-il être assuré pour la pratique d'un sport non couvert dans la notice ?"	Renseigner coord...	avecSport_sansQ5
		2	Vérifier qu'il est impossible de sélectionner une réponse pour la deuxième question.		
2	non	1	Dans le projet AUT_label_projet, sélectionner AUT_sport pour la question : "Le client souhaite-t-il être assuré pour la pratique d'un sport non couvert dans la notice ?"		
		2	Dans le projet AUT_label_projet, sélectionner AUT_questionnaire_en_ligne pour la question : "Le prospect accepte-t-il de remplir son questionnaire de santé en ligne en votre présence ?"	Remplir le questio...	sansSport_avecQ5
3	non	3	Cliquer sur le bouton pour remplir le questionnaire.		
		1	Dans le projet AUT_label_projet, sélectionner AUT_sport pour la question : "Le client souhaite-t-il être assuré pour la pratique d'un sport non couvert dans la notice ?"	Renseigner coord...	sansSport_sansQ5
		2	Dans le projet AUT_label_projet, sélectionner AUT_questionnaire_en_ligne pour la question : "Le prospect accepte-t-il de remplir son questionnaire de santé en ligne en votre présence ?"		

MBT (Model-Based Testing)... mais pas que...

Réponse 2 – Méthodologie outillée

Mise en place de Yest – Conception visuelle des tests

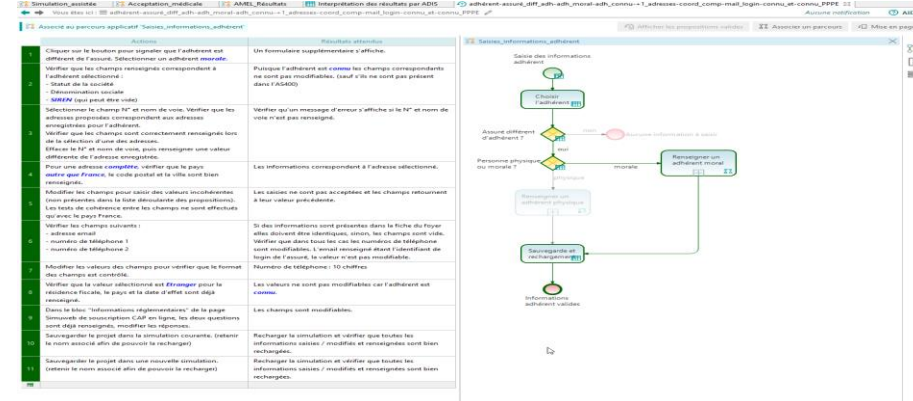
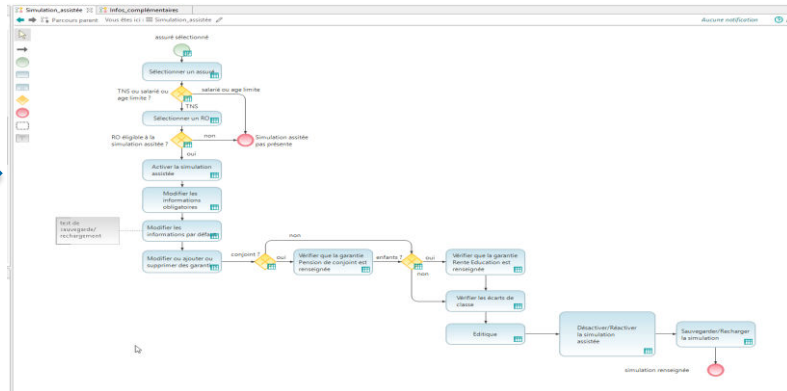


Accélération de nos phases de conception de test:

- ➔ *En conception initiale (ex: 6h pour 500 tests vs plus de 2,5j en conception classique) ➔ tests dans le sprint possibles*
- ➔ *En maintenance des tests (ex: 2h pour modifier 800+ tests critiques suite évolution de spécifications durant le sprint)*
- ➔ *Conception des tests itérative et incrémentale avec refactoring*
- ➔ *Conception à plusieurs (versioning Git)*
- ➔ *Pas besoin de modèles (BPMN, UML, etc.)*

Réponse 2 – Méthodologie outillée

Mise en place de Yest – Conception visuelle des tests



Maitrise de la couverture:

- ➔ *Meilleure visibilité sur ce que l'on veut tester*
- ➔ *Meilleure visibilité sur les fonctionnalités couvertes*
- ➔ *Conception des tests aussi simple sur une fonctionnalité isolée que sur un processus bout-en-bout*

Réponse 2 – Méthodologie outillée

Mise en place de Yest – Conception visuelle des tests



Ecosystème:

➔ *S'intègre à nos outils en place*

- *Publication Testrail: exécution manuelle et gestion des tests*
- *Publication Jira: traçabilité avec les besoins (exigences)*

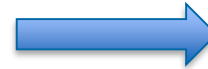
Réponse 2 – Méthodologie outillée

Mise en place de Yest – Conception visuelle des tests



Prise en main de l'outil peut être complexe:

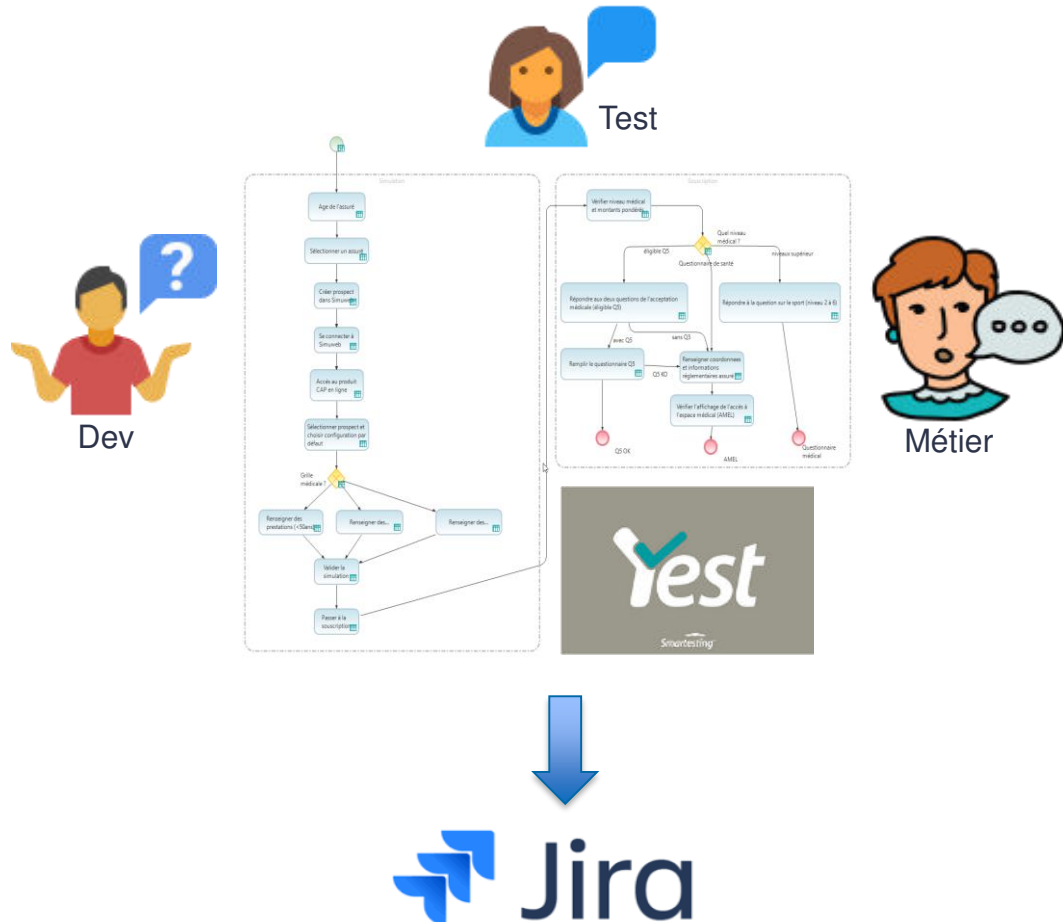
- *Nouvelles approches de conception de test*
 - *Conception itérative, incrémentale,*
 - *Conception visuelle, sous forme de workflow applicatif*
- *Nouveaux concepts*



Se faire accompagner au démarrage aide à aller plus vite

Réponse 3 – Collaboration

3 Amigos



Sous prétexte de validation des tests, mise en place d'ateliers 3 amigos



- *Echange et discussion entre Dev, Test et Métier*
- *Valider la compréhension du besoin par tous*
- *Echange facilité par le support visuel de Yest*
- *Parcours visuel compréhensible par tous*
- *Suivie de la couverture des fonctionnalités par des tests facilité grâce à l'intégration de Yest à Jira*

Réponse 3 – Collaboration

3 Amigos

Timing:

→ *L'atelier intervient souvent après le démarrage du développement*

Aujourd'hui

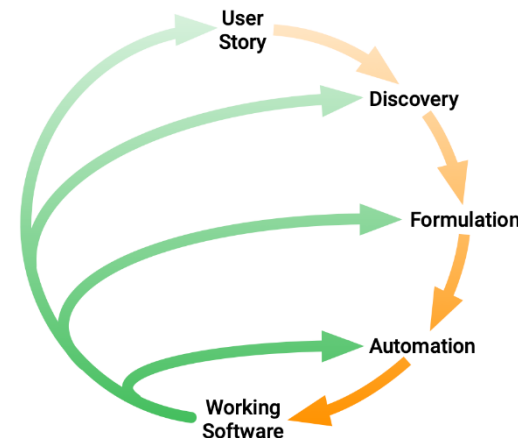
- *Prise de conscience de l'intérêt des ateliers*
- *Evolutions de nos pratiques en cours pour effectuer les ateliers avant le démarrage du sprint*
- *S'inscrire dans le processus de l'ATDD (Acceptance Test Driven Development)*

Découverte et Discussion autour du besoin

- *3 Amigos*
- *Clarification par des exemples → critères d'acceptation*

Formulation des tests

- *Au travers des parcours Yest*



Réponse 4 – Pair Testing

Mise en place du pair testing

The image shows two screenshots of a project management dashboard. The top screenshot displays a grid of task cards with various status icons (S, Y, I, A) and labels like 'Suivi de projet', '[Retour UX] Problème affichage des fonts sur IE9', and '[Acceptation médicale] Formalités médicales allégées'. A large blue arrow points from this dashboard to a second screenshot below it. In the second screenshot, the 'DEV EN COURS' tab is highlighted with a red dashed circle and labeled 'PAIR TEST', indicating the implementation of pair testing.

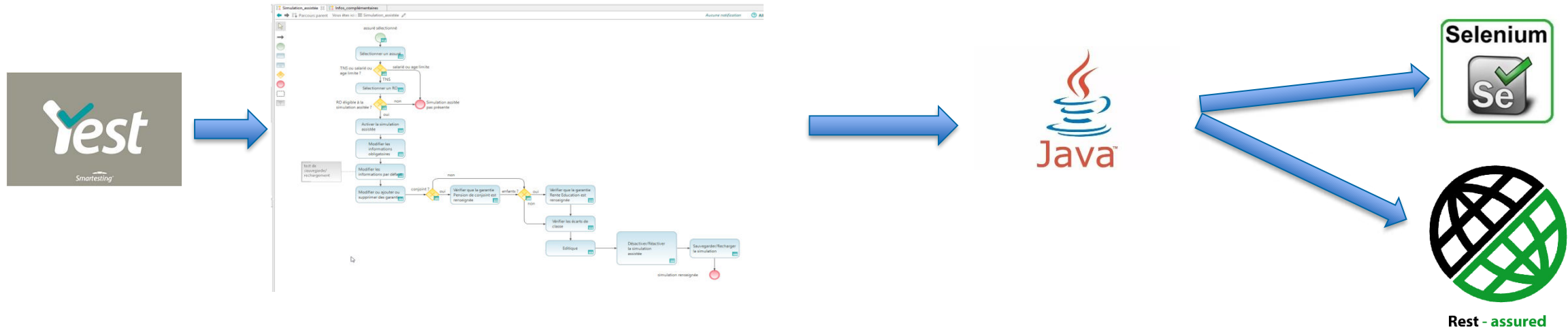


- Réduction du nombre d'anomalies détectés après la 1^{ère} livraison du développeur (50% à 70%)
- Rapide à déployer,
- Favorise l'échange entre le développeur et le testeur,
- Fait grandir mutuellement le développeur et le testeur sur le produit,
- Avec les parcours Yest, le pair test se fait parfois entre développeurs



- Réticence initiale de certains développeurs à l'égard de la pratique
 - Avoir les critères d'acceptation (parcours Yest) a permis entre autre de lever ces réticences

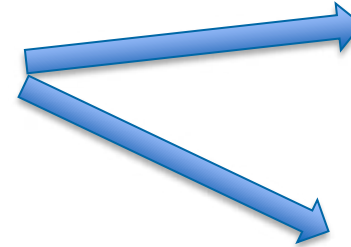
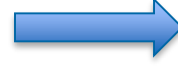
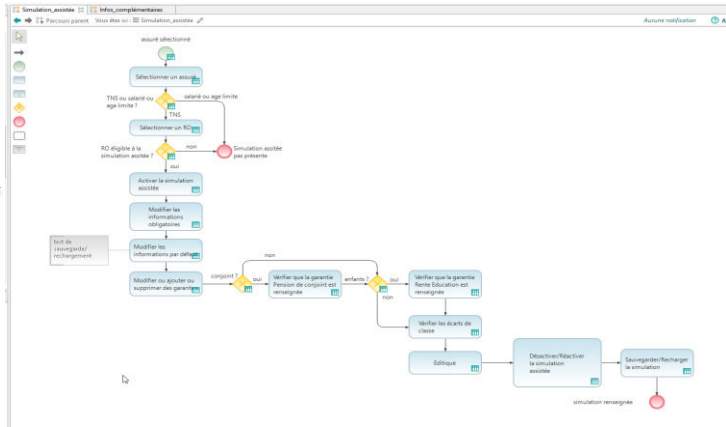
Réponse 5 – Automatisation



Mise en place de l'automatisation:

- Publication des scénarii de tests automatisés depuis Yest
- Publisher Java pour intégration avec Selenium et Rest-Assured
- Automatisation avec une approche par Mots-clés (Keyword Driven)

Réponse 5 – Automatisation



Rest - assured



→ *Travail de l'automaticien facilité*

- *Compréhension des fonctionnalités et des tests aisée grâce aux parcours visuels*
- *Collaboration entre le testeur et l'automaticien facilitée*

→ *Utilisation d'un même parcours aussi bien pour une exécution manuelle que pour une exécution automatisée*

→ *Les divers formats de publications disponibles (Java, Gherkin, RobotFramework, etc.) nous ont laissé le choix*

Réponse 5 – Automatisation

→ Le test automatisé n'est disponible que plusieurs sprints après l'introduction de la fonctionnalité → tests de régression impossibles à chaque sprint:

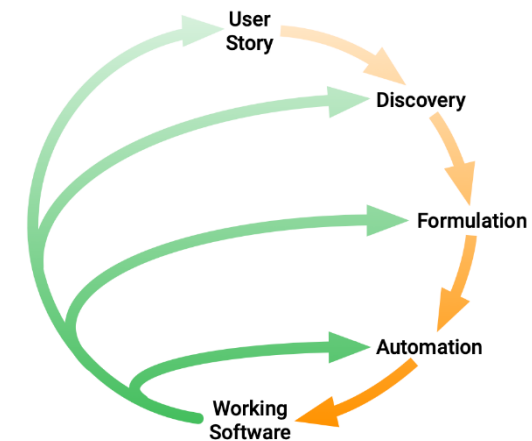
- Pas de définition des critères d'acceptation en amont du sprint
- Pas de compétences en automatisation au sein de l'équipe produit
- Tests automatisés non intégrés au pipeline CI/CD

Automatisation de l'exécution des tests

- S'appuyer sur les parcours Yest et les publications Java
- Insuffler la démarche TDD (Test Driven Development) auprès des développeurs

Aujourd'hui

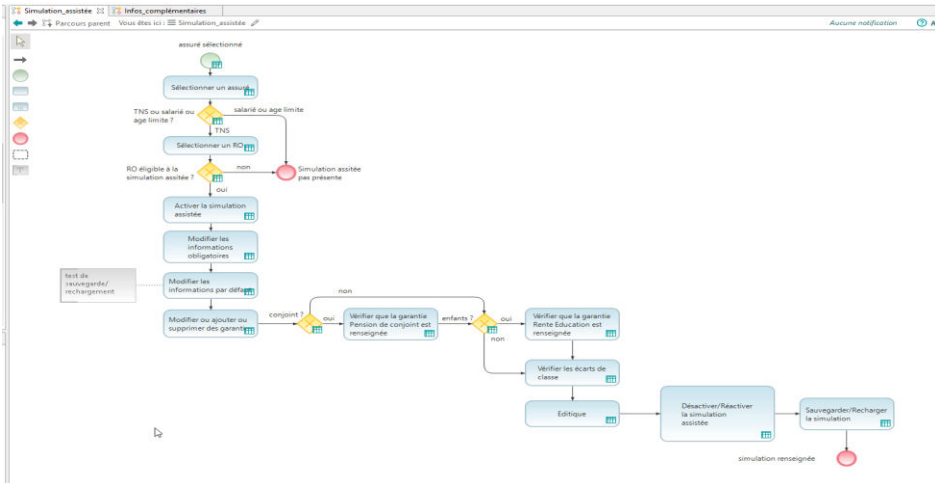
- Intégration des compétences en automatisation au sein des équipes produits (En cours)
- S'inscrire dans le processus de l'ATDD (Acceptance Test Driven Development)



Réponse 6 – Spécification vivante

1 constat:

Nos parcours visuels Yest sont toujours plus à jour que les spécifications !!!



VS



Vision **PRODUIT**

Table with 3 columns: Action, Etat de la souscription au moment de l'action, Résultat/règles. The table contains detailed information about the AMEL (Assurance Médicale Obligatoire) process, including conditions for eligibility and the generation of documents.

Confluence



Vision **PROJET**

Réponse 6 – Spécification vivante

Evolutions en cours / à venir:

- Adaptation de nos pratiques à l'approche ATDD: **Discussion – Formulation – Automatisation**
- Réorganisation de nos projets dans JIRA afin d'améliorer la gestion des backlogs produits par équipe → ***passer d'une gestion par projet à une gestion par produit***
- S'appuyer sur **Yest for Jira** pour la documentation de nos fonctionnalités dans Jira

Conclusion

Les chemins vers des pratiques de tests Agiles peuvent être divers et variés.

ADIS, par le pur des hasards, à trouver le sien dans la **Conception visuelle de test outillé**.

Des impacts au-delà du test:

- Prises de conscience qui s'opèrent auprès du **management** sur la nécessité d'évolution de la **philosophie de l'entreprise** autour de ses pratiques projets
- Adoption par certains développeurs des pratiques **TDD** en s'appuyant sur les critères d'acceptation illustrés par les parcours Yest



MERCI !