

# CLUB UTILISATEURS SQUASH

---

Jeudi 19 novembre 2020

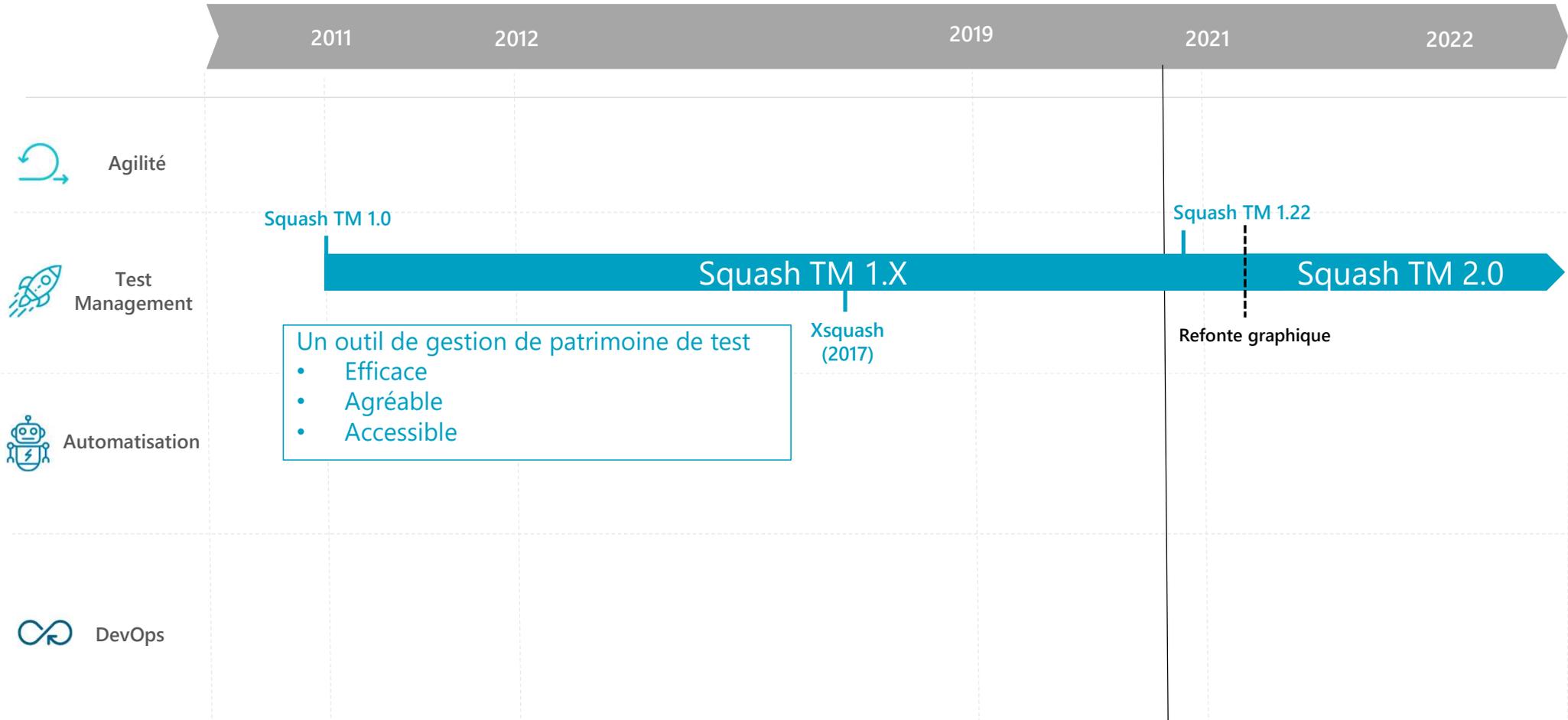


## SOMMAIRE / ORDRE DU JOUR

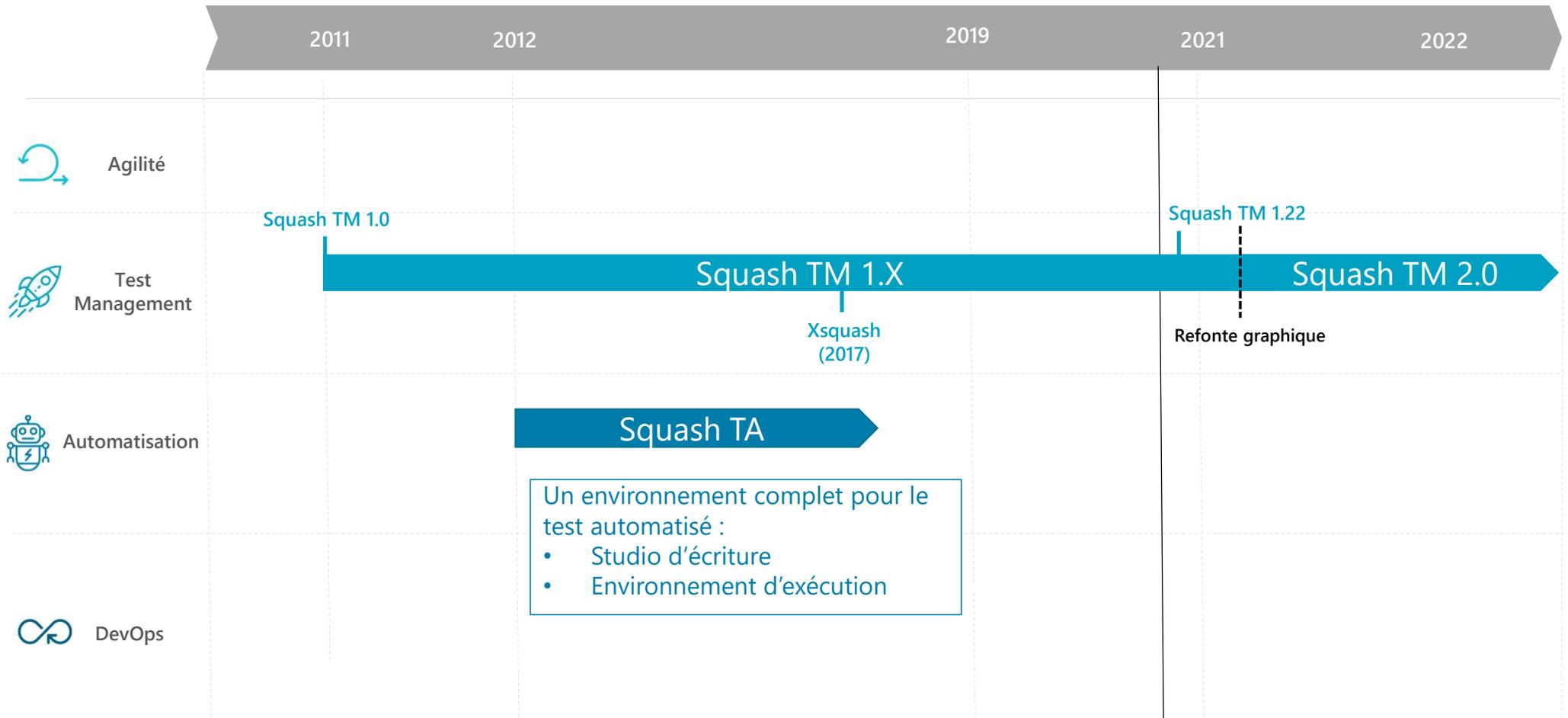
1. PRÉSENTATION DE LA NOUVELLE OFFRE SQUASH
2. LE BDD AVEC SQUASH
3. L'INTÉGRATION DES TESTS AUTOMATISÉS À LA CHAÎNE CI/CD AVEC SQUASH
4. REFONTE DE L'INTERFACE GRAPHIQUE DE SQUASH TM

# 1 PRÉSENTATION DE LA NOUVELLE OFFRE SQUASH

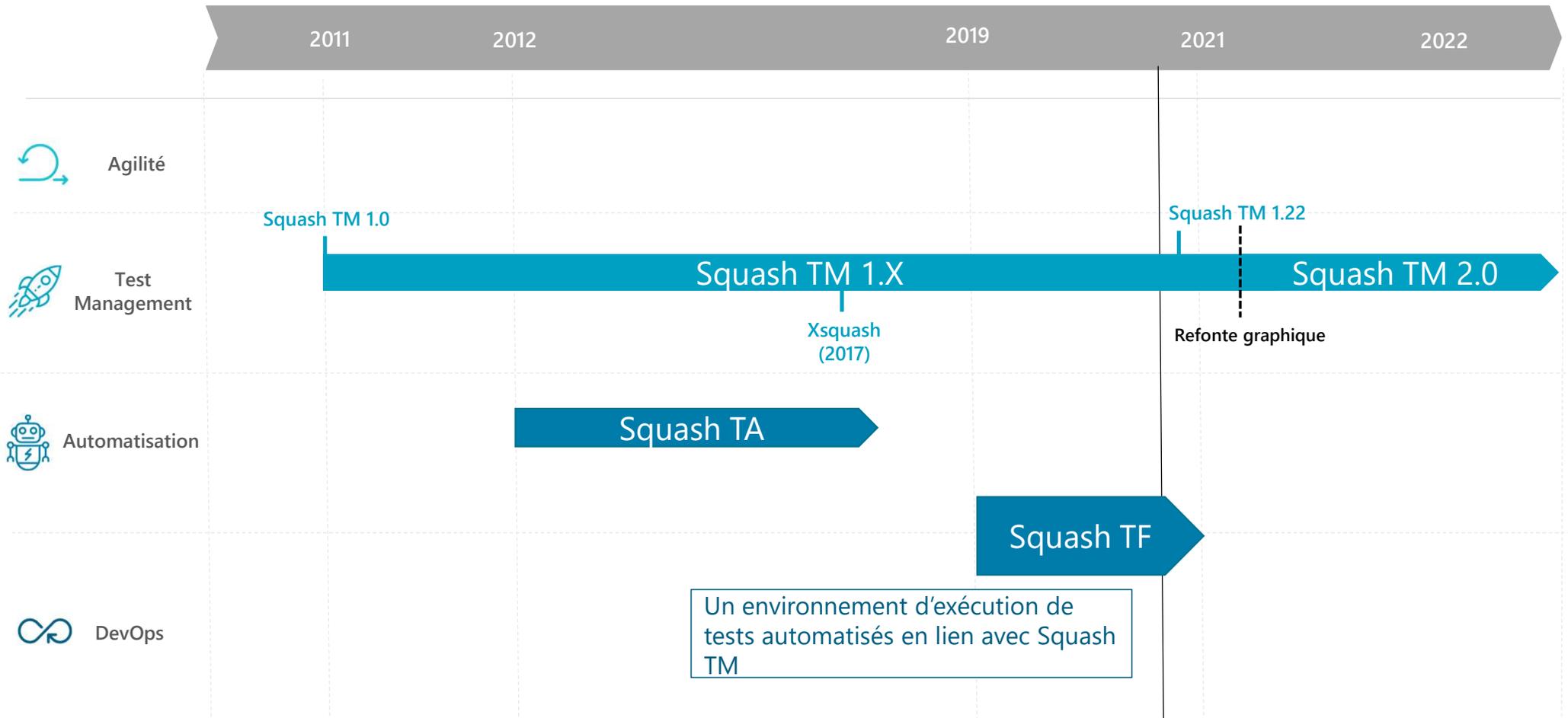
# RETOUR SUR LES ÉVOLUTIONS PRODUITS DE SQUASH



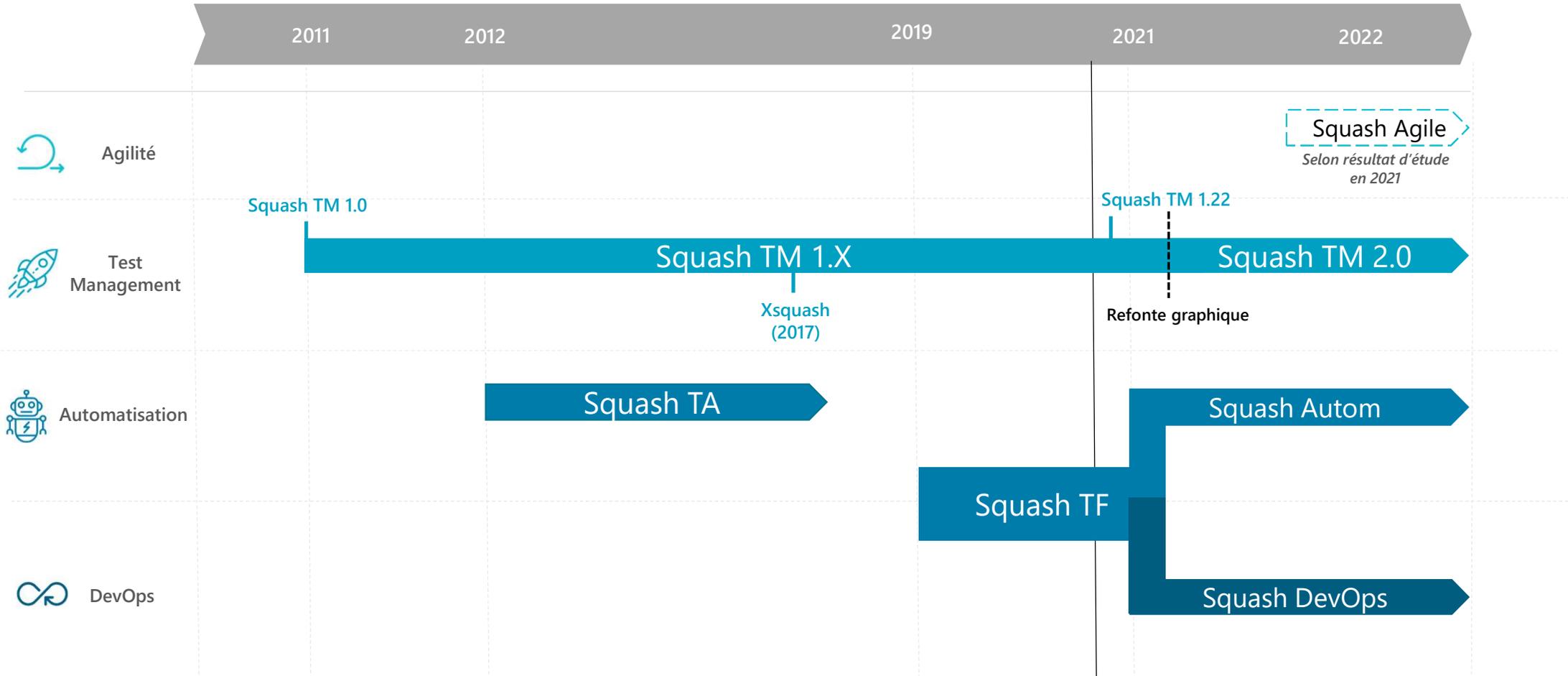
# RETOUR SUR LES ÉVOLUTIONS PRODUITS DE SQUASH



# RETOUR SUR LES ÉVOLUTIONS PRODUITS DE SQUASH



# RETOUR SUR LES ÉVOLUTIONS PRODUITS DE SQUASH



# PÉRIMÈTRES DES DIFFÉRENTS MODULES SQUASH



## Squash Test Management

		COMMUNITY	PREMIUM
Gestion du patrimoine du tests	Gérer vos référentiels d'exigences, cas de test et jeux de données	○	●
	Construire un patrimoine de test lié aux objets agiles issus de Jira	○	●
	Visualiser les tests et les résultats des exécutions dans Jira	○ Jira Server	● Jira Cloud
Gestion des campagnes de tests	Gérer la planification des tests et leur exécution	○	●
	Concevoir des plans d'exécution optimisés par le biais d'assistants		●
	Déclarer des anomalies préremplies avec les étapes de tests vers les bugtrackers	○ Mantis, Bugzilla	● Jira, Redmine, RTC, Tuleap
Pilotage de la qualité globale	Piloter l'avancement des projets de test via des indicateurs visuels et des tableaux de bord personnalisés	○	●
	Générer des bilans de test préremplis par Squash à partir d'un périmètre d'exécution		●
Intégration	Gérer l'authentification et sa délégation (LDAP, Active Directory, SAML 2.0)		
	Industrialiser l'administration fonctionnelle avec une API REST		●



## Squash Autom



## Squash DevOps

# PÉRIMÈTRES DES DIFFÉRENTS MODULES SQUASH



Squash Test Management

## Squash Autom

		COMMUNITY	PREMIUM
Conception avec Squash TM	Ecrire les tests au <b>format BDD</b>	<input type="radio"/>	<input checked="" type="radio"/> Avec auto-complétion
	Constituer et gérer une <b>bibliothèque d'actions</b> pour le BDD		<input checked="" type="radio"/>
	Structurer le process d'automatisation à l'aide de tâches et d'un <b>workflow d'automatisation</b> , directement dans Jira		<input checked="" type="radio"/>
Implémentation	<b>Associer les scripts</b> de test aux cas de test manuels, et les intégrer dans le plan d'exécution du référentiel de test	<input type="radio"/> Squash TM	<input checked="" type="radio"/> Squash TM
	Créer des <b>plans d'exécution « as code » (PEAC)</b> pour orchestrer précisément l'exécution des tests automatisés en dehors du référentiel de test	<input type="radio"/>	<input checked="" type="radio"/>
Exécution des tests automatisés	Piloter l'exécution des tests automatisée <b>depuis le référentiel de test</b>	<input type="radio"/> Squash TM	<input checked="" type="radio"/> Squash TM
	Piloter l'exécution des tests automatisés <b>via les PEAC</b>	<input type="radio"/>	<input checked="" type="radio"/>
	<b>Transmettre des paramètres</b> de Squash TM aux tests automatisés	<input type="radio"/> Cas de test	<input checked="" type="radio"/> + Campagne, Itération, Suite
	Consulter <b>l'historique des exécutions</b> automatisées dans Squash TM	<input type="radio"/> Suites automatisées	<input checked="" type="radio"/> + Tests automatisés
	Récupérer des scripts automatisés depuis le gestionnaire de code <b>Git</b>	<input type="radio"/>	<input checked="" type="radio"/>
	Déclencher, enchaîner et piloter des <b>tests automatisés multi-technologies</b>	<input type="radio"/> Cucumber, Cypress, Robot Framework, Junit, SoapUI	<input checked="" type="radio"/> + UFT, Ranorex, Agilitest, Katalon (2021)
	Visualiser les résultats de campagnes de tests automatisés via des <b>rapports HTML user friendly</b> (format Allure)	<input type="radio"/>	<input checked="" type="radio"/>

# PÉRIMÈTRES DES DIFFÉRENTS MODULES SQUASH



Squash Test Management



Squash Autom



Squash DevOps

		COMMUNITY	PREMIUM
Orchestration DevOps	Déclencher vos PEAC depuis une <b>étape de pipeline dédiée dans Jenkins</b>	○	●
	<b>Intégrer</b> vos PEAC dans <b>GitLab CI, Azure DevOps, AWS CodeBuild, Cloudbees CI, Xebia Labs XL</b>	○ (mi-2021)	● (mi-2021)
Exécution des tests automatisés	Récupérer des <b>plans d'exécution prédéfinis depuis Squash TM</b> , avec passage de paramètres	○ Cas de test	● + Campagne, Itération, Suite
	<b>Exploiter les résultats</b> des exécutions automatisées dans <b>Allure EE</b>		● 2021

# GRILLE TARIFAIRE DE L'OFFRE PREMIUM

OFFRE SERVEUR		Tarifs annuels en €HT				
SQUASH TM	Nombre d'utilisateurs déclarés dans Squash TM	50 utilisateurs	100 utilisateurs	250 utilisateurs	500 utilisateurs	500+ utilisateurs
		TARIF DE BASE ANNUEL	6 000 €	10 000 €	20 000 €	35 000 €
SQUASH AUTOM	TARIF ANNUEL	75 % du tarif de base annuel (*)				
SQUASH DEVOPS	TARIF ANNUEL	75 % du tarif de base annuel (*)				

(\*) pour utilisation de Squash AUTOM et/ou Squash DEVOPS avec un autre référentiel de test que Squash TM, tarif de base indexé sur le nombre d'utilisateurs de ce référentiel

OFFRE SERVICE		Tarifs annuels en €HT							
SQUASH TM	Nombre d'utilisateurs déclarés dans Squash TM	5 utilisateurs	10 utilisateurs	25 utilisateurs	50 utilisateurs	100 utilisateurs	250 utilisateurs	500 utilisateurs	500+ utilisateurs
		SAAS		1 080 €	2 160 €	4 320 €	8 640 €	15 000 €	30 000 €
	CDS EXTERNALISÉ OUTILLAGE DE TEST en sus de l'offre SERVEUR ou SaaS (modulaire avec catalogue de service en marque blanche pour vos projets)		à partir de (**) 10 000 €		à partir de (**) 15 000 €		à partir de (**) 20 000 €	à partir de (**) 30 000 €	Sur devis
SQUASH AUTOM SQUASH DEVOPS	MCO DE TEST FACTORY (sur site client)				Sur devis				

(\*\*) selon services choisis, sur devis

Exemple de calcul pour une offre Serveur pour 100 utilisateurs avec les modules Squash TM, Squash AUTOM et Squash DEVOPS : 10 000€ + 0,75 x 10 000€ + 0,75 x 10 000€ = 25 000 €HT/an

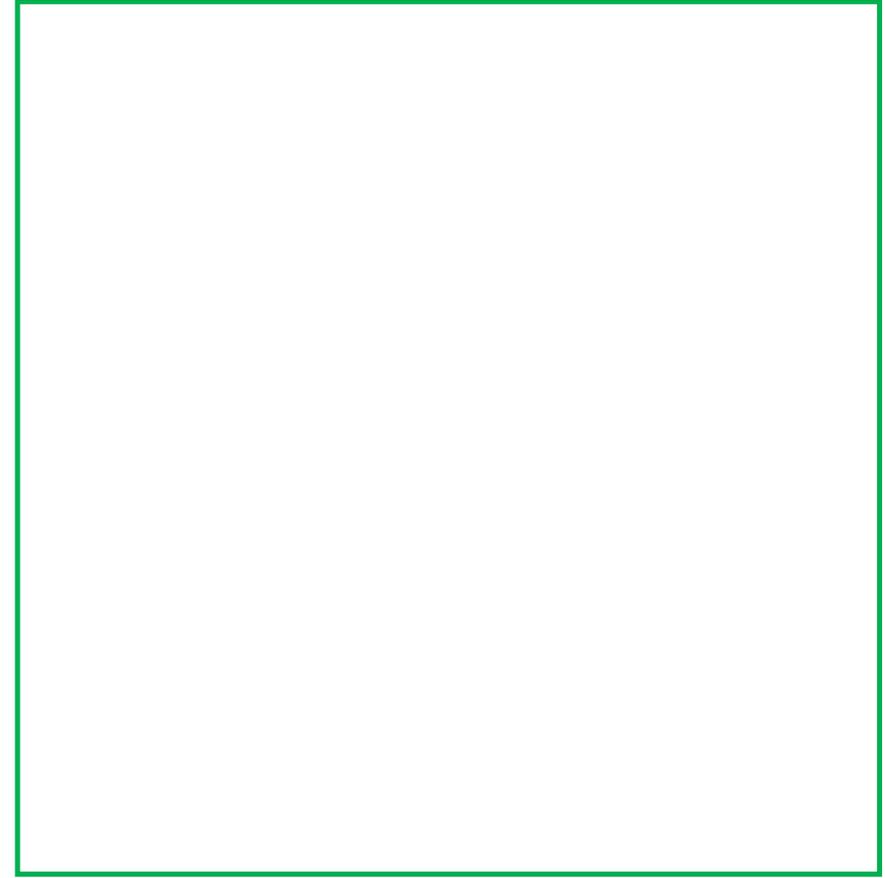
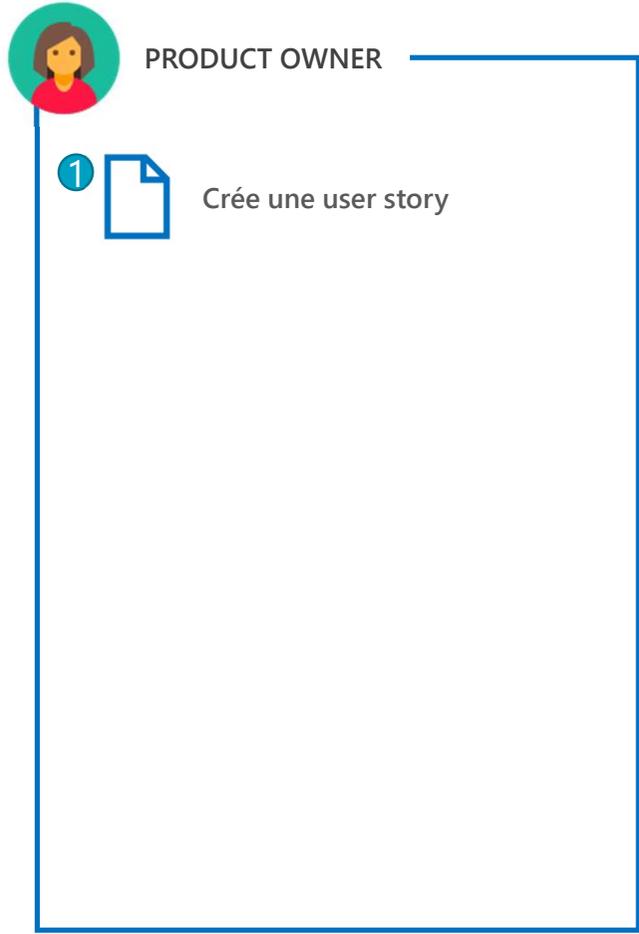


## 2 LE BDD AVEC SQUASH (SQUASH TM 1.22)

# LE BDD AVEC SQUASH



◆ Jira Software

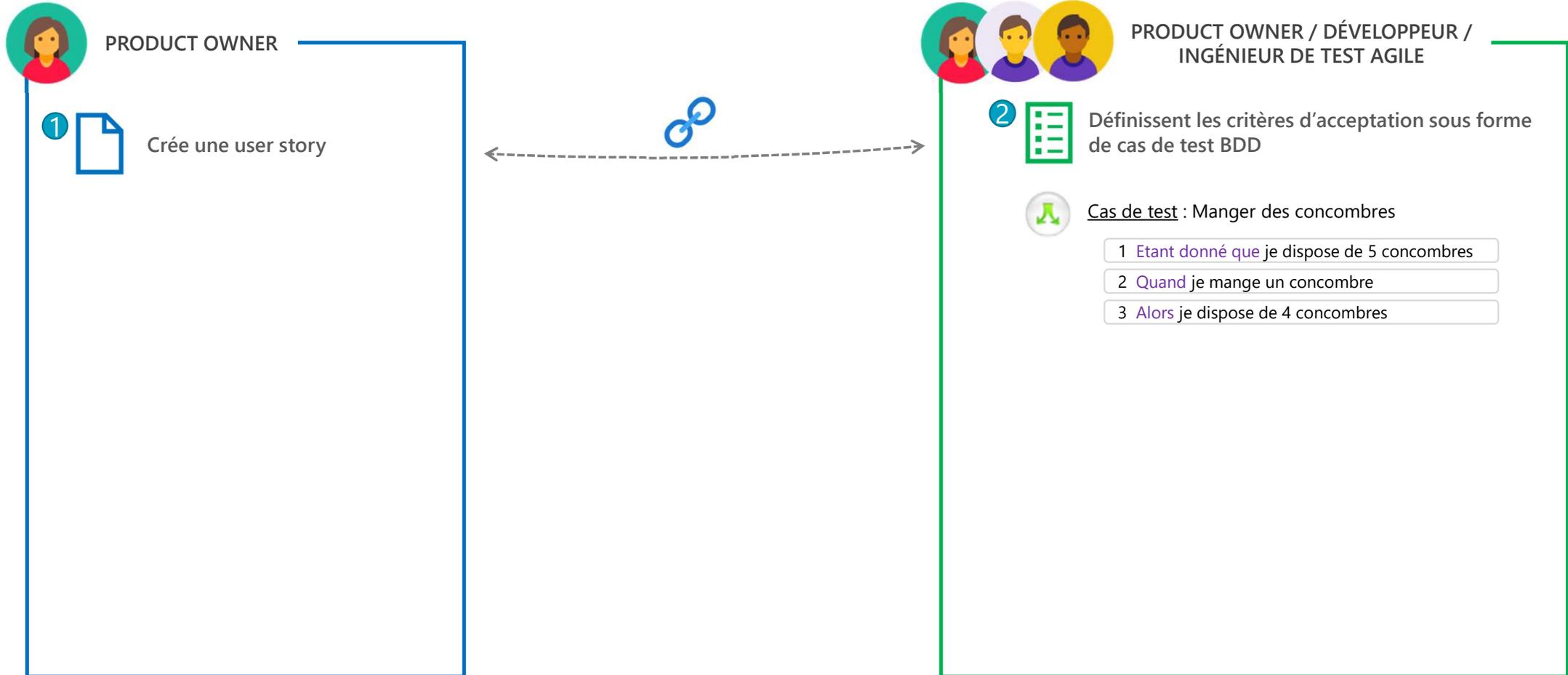


# LE BDD AVEC SQUASH



◆ Jira Software

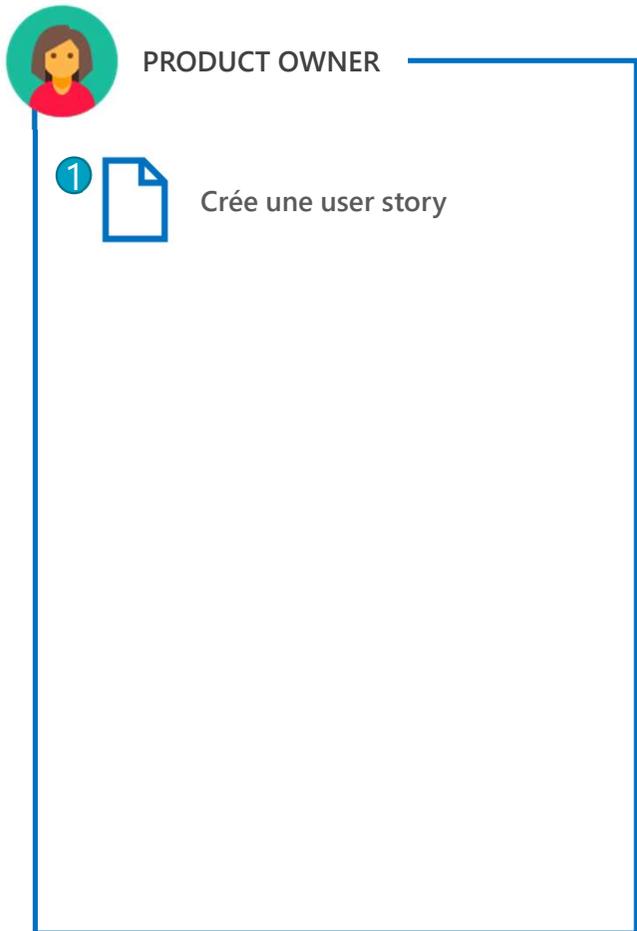
squash



# LE BDD AVEC SQUASH



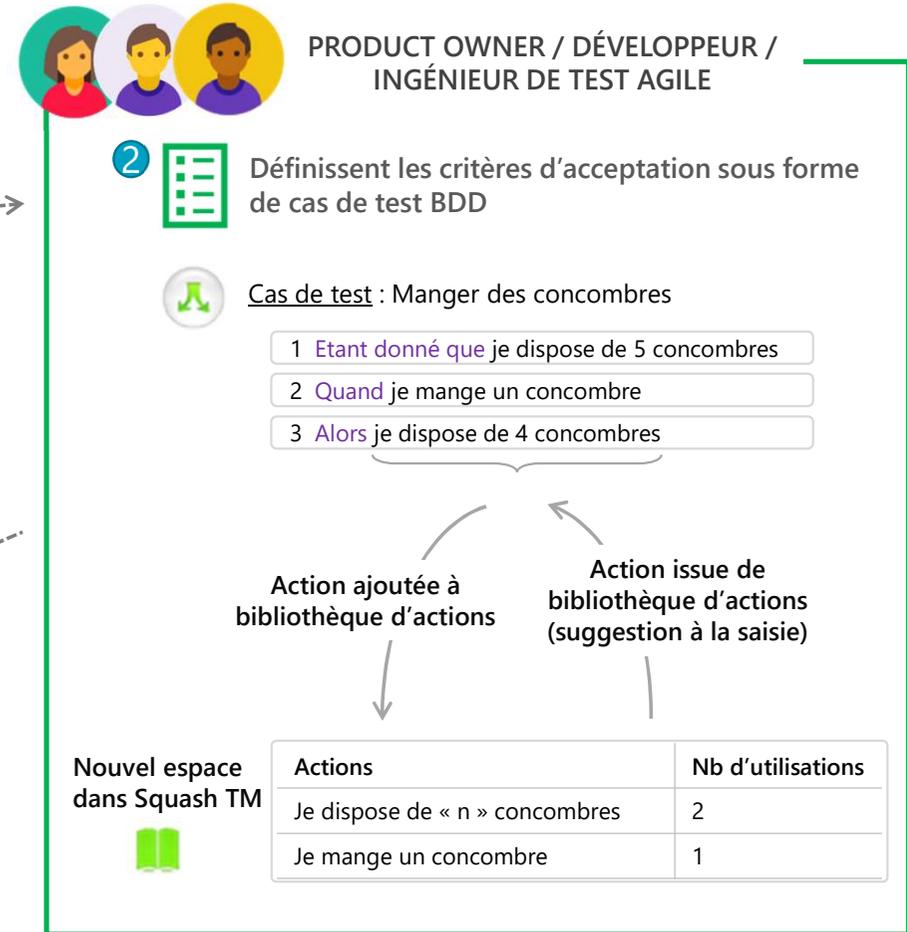
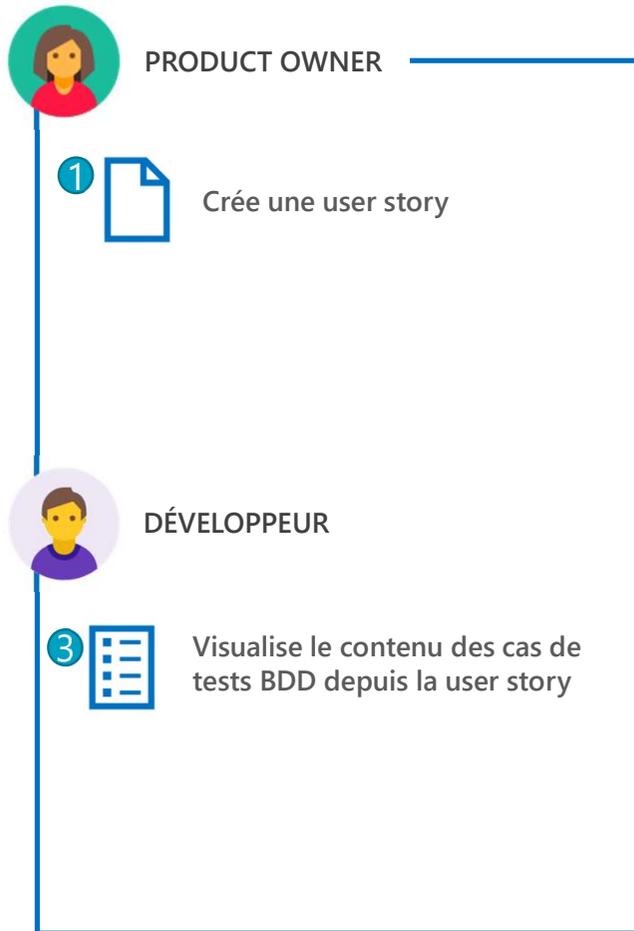
## Jira Software



# LE BDD AVEC SQUASH



## Jira Software



XsQUASH

# LE BDD AVEC SQUASH



INGÉNIEUR DE TEST AGILE

5



Partage le scénario BDD

Demande l'automatisation des actions



Conversion du scénario selon la technologie d'implémentation paramétrée



AUTOMATICIEN



Récupère le scénario de test

# LE BDD AVEC SQUASH



INGÉNIEUR DE TEST AGILE

5



Partage le scénario BDD

Demande l'automatisation des actions



Conversion du scénario selon la technologie d'implémentation paramétrée



AUTOMATICIEN



6

Récupère le scénario de test



7

Implémente les actions manquantes

```
@When("je mange un concombre")
public void eatingCucumber(int cucumberQuantity) {
    final CucumberConsumer joe = CucumberConsumerFactory.getConsumer();
    final Cucumber cucumber = CUCUMBER_JAR.getCucumber();
    joe.eat(cucumber);
}
```



8

Mets à disposition les actions



# LE BDD AVEC SQUASH



INGÉNIEUR DE TEST AGILE

5



Partage le scénario BDD

Demande l'automatisation des actions



Conversion du scénario selon la technologie d'implémentation paramétrée

10



Exécute le test automatisé



Actions marquées comme étant implémentées



AUTOMATICIEN



6

Récupère le scénario de test



7

Implémente les actions manquantes

```
@When("je mange un concombre")
public void eatingCucumber(int cucumberQuantity) {
    final CucumberConsumer joe = CucumberConsumerFactory.getConsumer();
    final Cucumber cucumber = CUCUMBER_JAR.getCucumber();
    joe.eat(cucumber);
}
```



8

Mets à disposition les actions



9

A terminé l'automatisation



## AVANTAGES PAR RAPPORT AUX CAS DE TEST GHERKIN



### ● Ecriture plus intuitive pour l'utilisateur



- Sélection des mots-clés dans une liste déroulante
- Suggestion d'actions existantes à la saisie
- Utilisation des paramètres et jeux de données comme pour les cas de test classiques

### ● Constitution d'une bibliothèque d'actions



- Réutilisation d'actions d'un cas de test à un autre (y compris en inter-projet)
- Maintenance et rédaction des scénarios facilitées
- Informations sur l'implémentation de chaque action

### ● Export des scénarios pour une implémentation dans différentes technologies



- Conversion des scénarios selon la technologie cible
- Pas d'impact pour l'utilisateur lors de la rédaction des cas de test
- L'utilisateur n'a pas à gérer toutes les subtilités liées à chaque langage

# CONVERSION DES SCÉNARIOS SELON LA TECHNOLOGIE CIBLE



## Manger des concombres

1	Given	je dispose de 5 concombres
2	When	je mange un concombre
3	Then	je dispose de 4 concombres

.feature



```
# language: en
Feature: Manger des concombres

Scenario: Manger des concombres
  Given je dispose de 5 concombres
  When je mange un concombre
  Then je dispose de 4 concombres
```



Fichier transmis à  
l'automaticien



```
Integer nombre;

@Given("je dispose de {int} concombres")
public void je_dispose_de_concombres(Integer nombre) {
    this.nombre = nombre;
}

@when("je mange un concombre")
public void je_mange_un_concombre() {
    nombre = nombre - 1;
}

@Then("je dispose de {int} concombres")
public void je_dispose_de_N_concombres(Integer expected) {
    Assert.assertEquals(expected, nombre);
}
```

Actions  
implémentées

# CONVERSION DES SCÉNARIOS SELON LA TECHNOLOGIE CIBLE



## Manger des concombres

1	Given	je dispose de 5 concombres
2	When	je mange un concombre
3	Then	je dispose de 4 concombres

.feature



```
# language: en
Feature: Manger des concombres

Scenario: Manger des concombres
  Given je dispose de 5 concombres
  When je mange un concombre
  Then je dispose de 4 concombres
```



```
Integer nombre;

@Given("je dispose de {int} concombres")
public void je_dispose_de_concombres(Integer nombre) {
    this.nombre = nombre;
}

@When("je mange un concombre")
public void je_mange_un_concombre() {
    nombre = nombre - 1;
}

@Then("je dispose de {int} concombres")
public void je_dispose_de_N_concombres(Integer expected) {
    Assert.assertEquals(expected, nombre);
}
```

Fichier transmis à  
l'automaticien

Actions  
implémentées



.robot

```
*** Settings ***
Resource  squash_resources.resource

*** Test Cases ***
Manger des concombres
  Given je dispose de "5" concombres
  When je mange un concombre
  Then je dispose de "4" concombres
```



```
*** Settings ***
Documentation  Keywords for Squash test

*** Variables ***
${concombres}

*** Keywords ***

Given je dispose de ${arg1} concombres
  ${concombres} = ${arg1}

When je mange un concombre
  ${concombres} = ${concombres} - 1

Then je dispose de ${expected} concombres
  Should Be Equal  ${concombres}  ${expected}
```

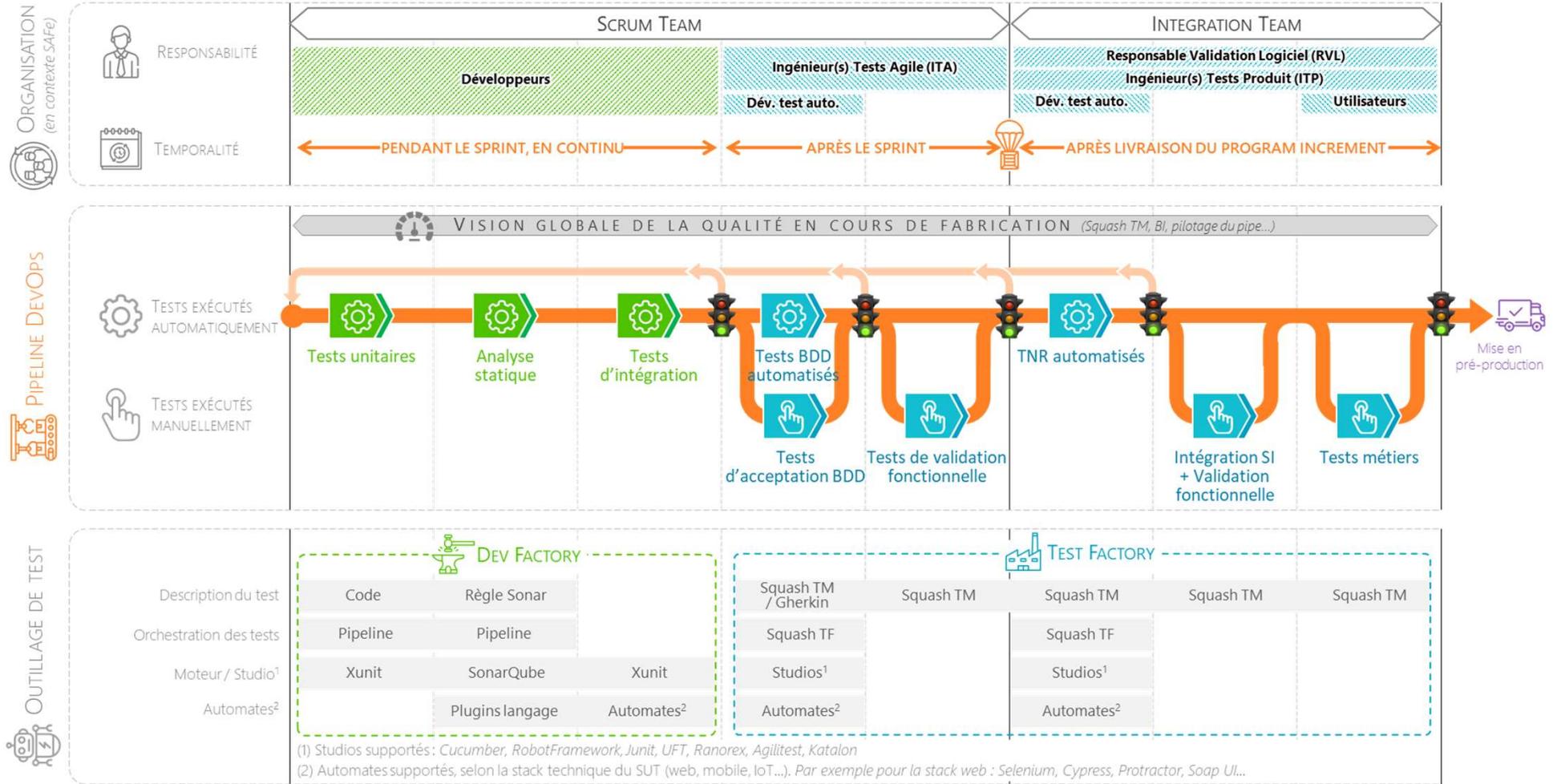


# 3 L'INTÉGRATION DES TESTS AUTOMATISÉS À LA CHAÎNE CI/CD AVEC SQUASH

- 3.1 – Les tests dans le CI/CD
- 3.2 – L'orchestrateur Squash de tests automatisés
- 3.3 – Détail des services de l'orchestrateur



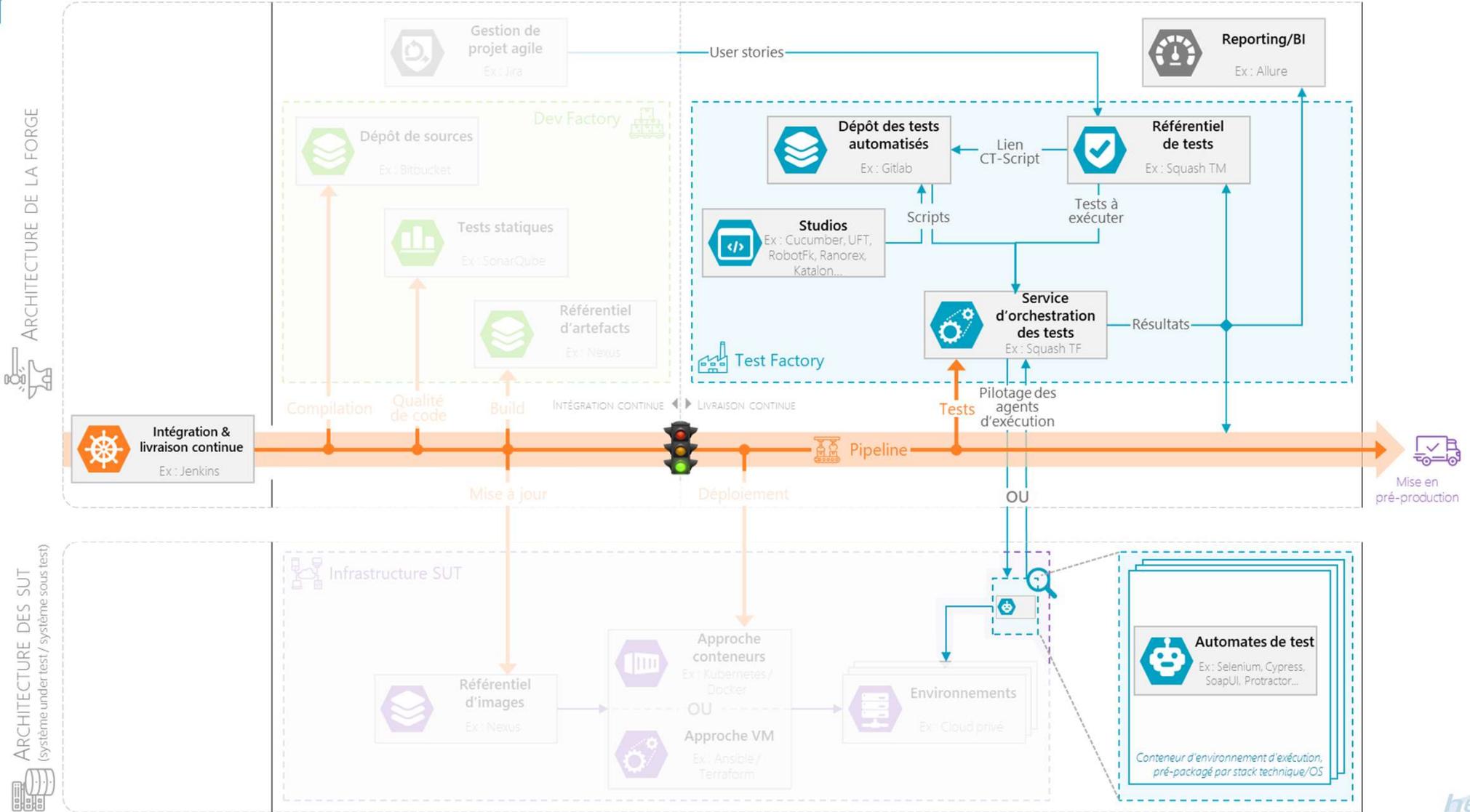
# INTÉGRATION DES TESTS DANS LE CI/CD





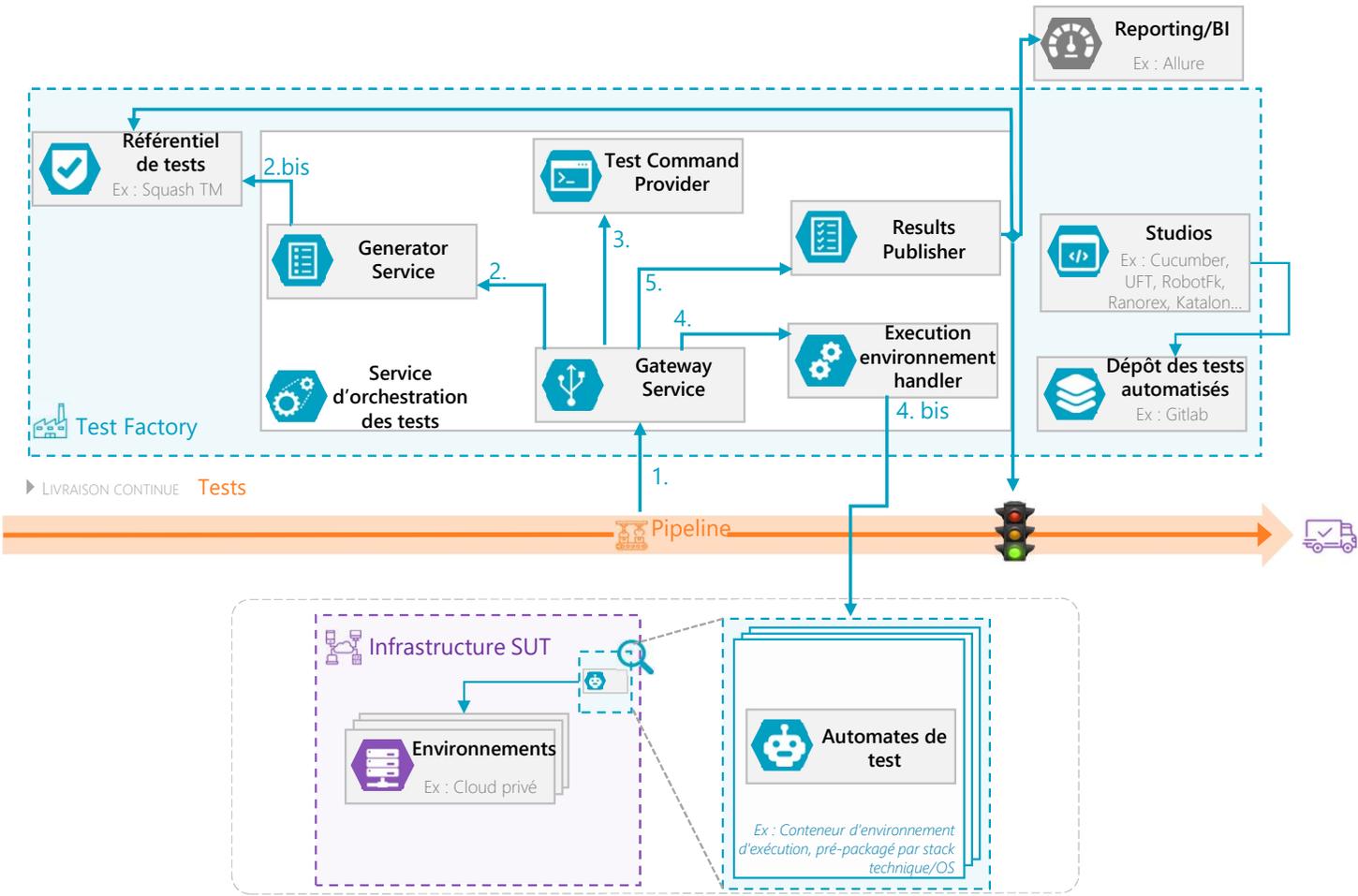
3.1

# TEST FACTORY: INTÉGRATION À LA CHAÎNE DEVOPS





# 3.2 L'ORCHESTRATEUR SQUASH

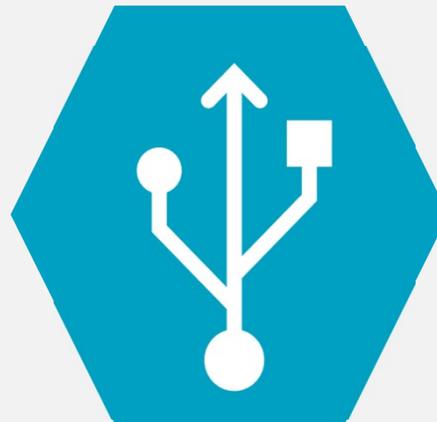
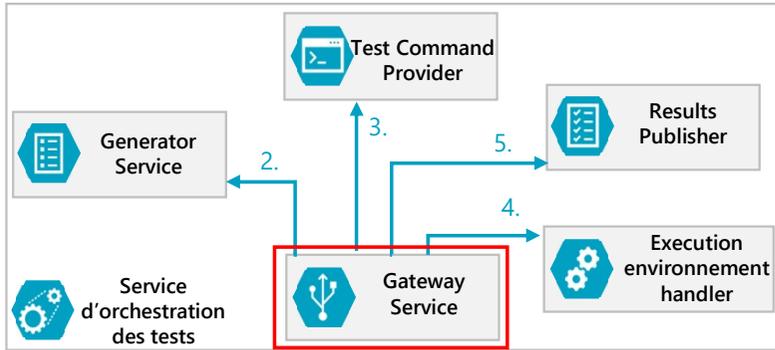


- 1 Transmission du plan d'exécution « as code »
- 2 Récupération d'une liste de test définis dans un référentiel de test
- 3 Interprétation des étapes du descriptif en ligne de commande
- 4 Exécution des commandes sur l'environnement d'exécution
- 5 Transmission des résultats



3.3

## DÉTAILS DES SERVICES



### Gateway Service

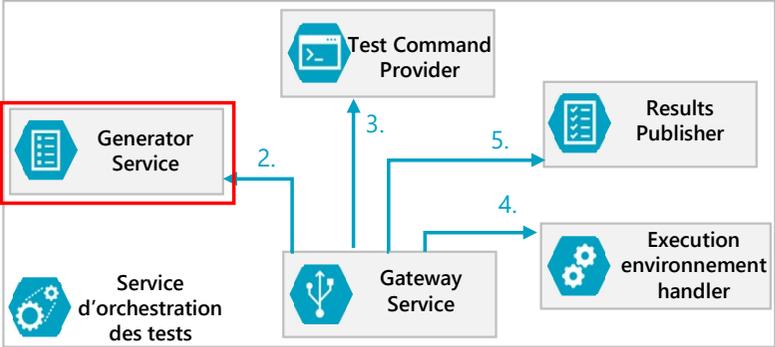
- Service d'interfaçage entre l'orchestrateur et son appelant
- Réceptionne les plans d'exécution « as code »

```
def workflow_id = runSquashTFWorkflow(
  workflowPathName: 'testWorkflow.json',
  workflowTimeout: '20S',
  serverName: 'local insecure'
)
```



3.3

# DÉTAILS DES SERVICES



```
"generator": "tm.squashtest.org/tm.generator@v1",
"with": {
  "testPlanUuid": "feeca3a1-31c3-4bb9-8eb6-8229b1149650",
  "testPlanType": "Iteration",
  "squashTMUrl": "http://127.0.0.1:8082/squash/",
  "tagLabel": "criticality",
  "tagValue": "high"
}
```



Community/Premium



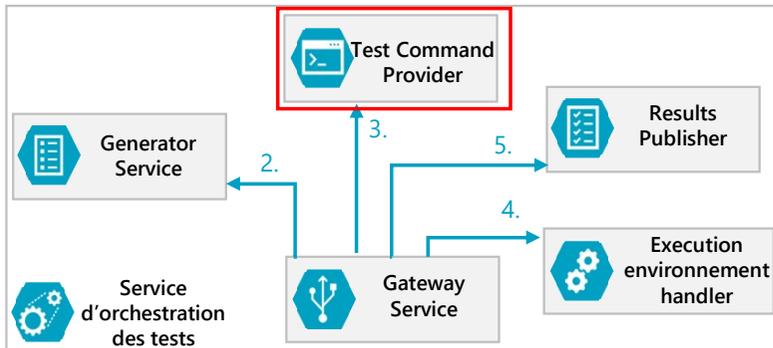
## Generator Service

- Récupère une liste de test à exécuter depuis un référentiel de test

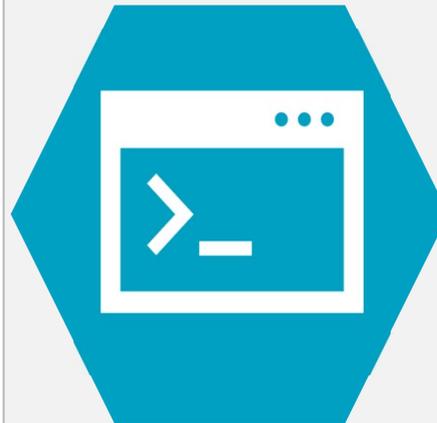


3.3

## DÉTAILS DES SERVICES



```
- uses: action/checkout@v2
  with:
    repository: https://github.com/robotframework/RobotDemo.git
- uses: robotframework/robot@v1
  with:
    datasource: RobotDemo/data_driven.robot
```



### Test Command Provider

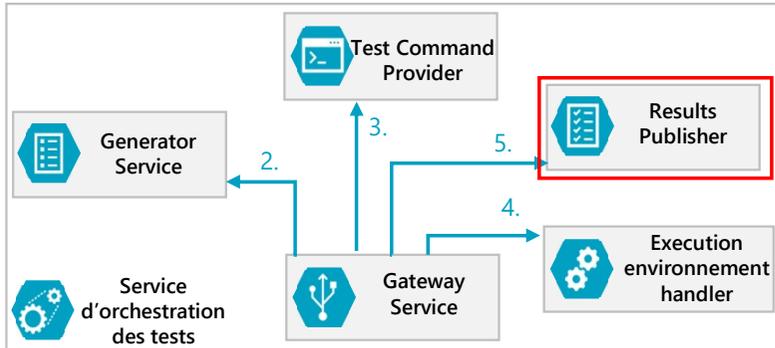
- Fournit des actions appelables au sein du plan d'exécution « as code » Squash pour en faciliter l'écriture
- Exploité par les services « générateurs » pour créer les étapes du plan d'exécution « as code » Squash





3.3

## DÉTAILS DES SERVICES



squash

Community/Premium



Premium



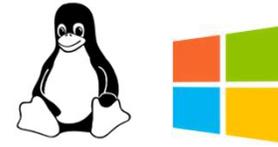
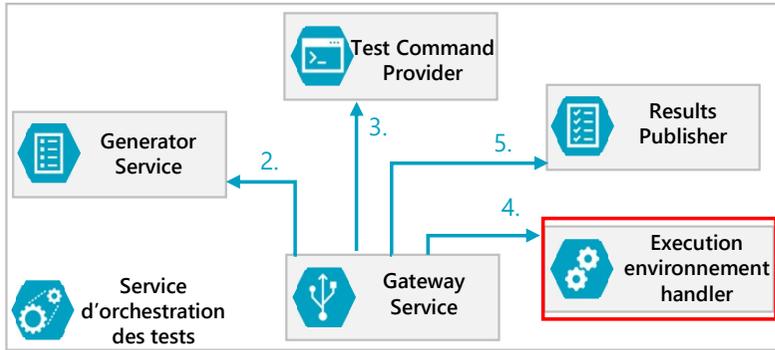
### Results Publisher

- Publie des résultats en fin d'exécution
- Transfère les rapports d'exécutions



3.3

# DÉTAILS DES SERVICES



## Execution Environment Handler

- Passe les commandes du workflow Squash à l'environnement d'exécution

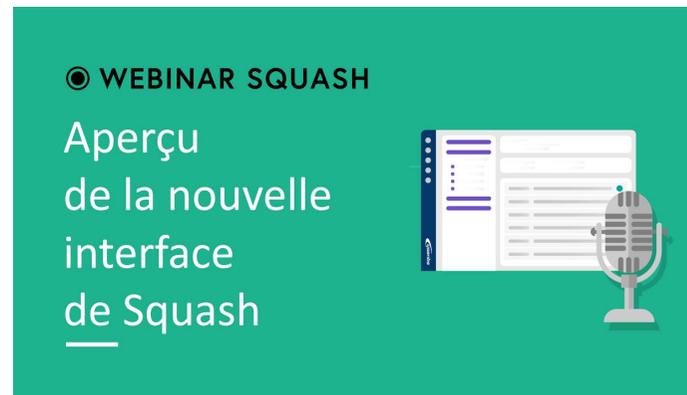
# 4 REFONTE DE L'INTERFACE GRAPHIQUE DE SQUASH TM

# REFONTE DE L'INTERFACE GRAPHIQUE DE SQUASH TM

- Instance de démo disponible en ligne

<https://demo-refonte.squashtest.org/login>

- Webinar dédié disponible sur notre chaîne YouTube



# ROADMAP SQUASH 2021 PRÉVISIONNELLE

# ROADMAP SQUASH 2021 PRÉVISIONNELLE



# ANNEXES

# SQUASH TM 1.22

## CAS DE TEST AU FORMAT BDD

- Nouveau format BDD avec caractéristiques classiques :
  - Attributs : Importance, Nature, Type, champs et listes personnalisées
  - Association aux exigences
  - Ajout de pièces jointes
- Pas de test basés sur la syntaxe Gherkin et composés :
  - D'un mot-clé : Given, When, Then, And, But
  - D'une action : phrase qui suit le mot-clé

The screenshot shows a web interface for managing test cases. At the top, there is a green header with a back arrow and the title "Vérifier que la machine est disponible". Below the header, the creation and modification dates are listed: "Créé le : 17/07/2020 16:53 (admin2)" and "Modifié le : 17/07/2020 17:19 (admin2)". To the right of these dates are buttons for "Renommer" and "Imprimer". Below this is a navigation bar with tabs: "Informations", "Pas de test" (which is active), "Paramètres", "Jalons", "Pièces jointes", and "Exécutions". Under the "Pas de test" tab, there is a form to add a new step: "Ajouter un pas de test : [Given] Saisir une action" with an "Ajouter" button and a "Supprimer" button. Below the form is a table with three columns: "#", "Mot-clé", and "Action". The table contains three rows of test steps, each with a delete icon.

#	Mot-clé	Action
1	Given	la machine est branchée
2	When	je passe mon badge
3	Then	je constate que mon solde s'affiche

## VARIABILISATION DES CAS DE TEST BDD

- Une action est composée de parties fixes et de **paramètres**
- Au sein des pas de test, les paramètres des actions peuvent être valorisés de différentes manières :

Par des valeurs libres  
(saisies entre " ")

#	Mot-clé	Action
1	Given	la machine est branchée
2	And	mon solde est au moins de 0.80
3	When	je sélectionne le Cappuccino
4	Then	la machine me sert un Cappuccino et mon compte est débité de 0.80

Par des paramètres du cas de test  
(saisis entre < >)

=> Correspond au Scenario Outline Gherkin

#	Mot-clé	Action
1	Given	la machine est branchée
2	And	mon solde est au moins de <prix>
3	When	je sélectionne le <produit>
4	Then	la machine me sert un <produit> et mon compte est débité de <prix>

#	Jeu de données	prix	produit
1	jdd1	0.40	Espresso
2	jdd2	0.50	Lungo
3	jdd3	0.80	Cappuccino

## 3.1 BIBLIOTHÈQUE D' ACTIONS

- Nouvel espace dans Squash
- Une bibliothèque d'actions par projet
- Depuis cette espace :
  - Ajout / Suppression d'actions
  - Gestion des paramètres des actions : nom et valeur par défaut
  - Consultation d'informations propres aux actions : liste des cas de test utilisant l'action, informations sur son implémentation

The screenshot displays the 'Bibliothèque d'actions' (Actions Library) interface. On the left, a sidebar shows a list of actions under 'Demo BDD (cucumber)' and 'Demo BDD (robot)'. The main area shows a detailed view for the action 'je sélectionne le "produit"'. The interface includes a description field, a parameters table, a table of test cases using the action, and an implementation table.

**Parameters**

#	Nom	Valeur par défaut
1	produit	Cappuccino

**Cas de test utilisant cette action**

#	Référence	Nom	Importance	Statut	Statut d'automatisation
1	CT02	Vérifier la livraison d'un Cappuccino	Faible	En cours de rédaction	Automatisé
2	CT03	Vérifier la livraison des produits	Faible	En cours de rédaction	-

**Implémentation**

#	Technologie	Date de dernière implémentation
1	Robot Framework	28/09/2020 11:16

## SUGGESTION D' ACTIONS À LA RÉDACTION DE PAS DE TEST

- A la rédaction de pas de test, l'utilisateur se voit suggérer les actions déjà présentes dans la bibliothèque d'actions du projet et correspondant à sa saisie :
  - Une action peut être partagée entre plusieurs cas de test
  - Si il n'y a pas de correspondance avec les actions déjà présentes, une nouvelle action est créé dans la bibliothèque du projet



## TRANSMISSION A L'AUTOMATICIEN ET EXECUTION

- Transmission automatique dans Git (via le workflow d'automatisation) des scripts des cas de test BDD
- Conversion du script dans un format qui dépend de la technologie d'implémentation cible :
  - .feature pour une implémentation en Cucumber
  - .robot pour une implémentation en Robot Framework
- Paramétrage de la technologie d'implémentation au niveau du projet Squash :
  - Une technologie par projet
  - Pas d'impact sur la rédaction des cas de test BDD
- Exécution manuelle ou automatisée

# XSQUASH (PLUGIN JIRA)

- Prise en compte des cas de test au format BDD
- Ajout de nouvelles informations dans les onglets Cas de test et Exécutions

Toutes Commentaires Journal de travail Historique Activité **Cas de test Squash TM** Exécutions Squash TM

Seuls les cas de test associés directement à la demande sont affichés.

ID	Libellé	Référence	Importance	Statut d'écriture	Statut d'automatisation	Dernière exécution
1840	Is it Friday yet (param CT)		Haute	Approuvé	Fini	<span style="color: red;">●</span>

**Description**  
Aucune

**Jeux de données**

Nom	
1	jdd Nope

Paramètres	
Nom	Valeur
1	answer

2	jdd Yep
---	---------

**Pas de test**

Action	
1	GIVEN today is Sunday
2	WHEN I ask whether it's Friday yet
3	THEN I should be told <answer>

1892	Is it Friday yet (Yep)		Moyenne	En cours de rédaction	Fini	<span style="color: green;">●</span>
1838	Is it Friday yet (Nope)		Faible	En cours de rédaction	/	<span style="color: gray;">●</span>



## NETTOYAGE DES SUITES AUTOMATISÉES

- Il est possible de supprimer les anciennes suites automatisées et exécutions automatisées :
  - Paramétrage d'une durée de conservation des suites et exécutions automatisées
    - ⇒ Au niveau de chaque projet
    - ⇒ Action réalisée par le chef de projet
  - Suppression des suites et exécutions automatisées antérieures à la durée de conservation paramétrée pour chaque projet
    - ⇒ Au niveau de l'instance
    - ⇒ Action réalisée par l'administrateur

# SQUASH TM 2.0

# PAGE DE CONSULTATION D'UN CAS DE TEST

The screenshot displays the Squash test case management interface. On the left is a sidebar with navigation options: Exigences, Cas de test (highlighted), Campagnes, Pilotage, and Automatisation. Below these are sections for Administration, Bugtrackers, Aide, Jalons, and Filtre projets. The main area is titled 'Accès au formulaire de création' and shows details for test case 'ECOL\_01'. At the top, there are status buttons: 'À approuver' (selected), 'Très haute', and 'Succès'. The 'Informations' section lists: Statut (À approuver), Id (6), Importance (Très haute), Nature (Sécurité), Type (Non-régression), Jalons (MEP v.2), and TNR. The description states: 'Le test a pour but de vérifier qu'il est possible d'accéder au formulaire de création des écoles.' The 'Exigences vérifiées par ce cas de test' section contains a table with one entry:

#	PROJET	RÉFÉRENCE	EXIGENCE	JALONS	CR.	ST.	VÉRIFIÉE PAR
1	Demo Squash	CLAS_02	Accès au formulaire de création d'une classe	28/01/2019	▲	▲	Ce cas de test

The 'Paramètres et jeux de données' section includes a table for test data:

NOM	URL	env_name	navigateur
Firefox	http://www.openecole.fr	Qual	Firefox 50+
IE	http://www.openecole.fr	Dev	IE 10+

# TABLEAU DE BORD DES CAS DE TEST

**OPEN ECOLE**

- 01 - Connexion à OpenEcole
- 02 - Gestion des écoles
  - ECOL\_01 - Accès au formulaire de cré...
  - ECOL\_02 - Ajout d'une école
  - ECOL\_03 - Modification d'une école
  - ECOL\_04 - Suppression d'une école
  - ECOL\_05 - Editer les données d'une é...
  - ECOL\_06 - Recherche simple d'une éc...
  - ECOL\_07 - Faire une recherche à parti...
  - ECOL\_08 - Recherche multi-critères d'...
- 03 - Inscription des élèves
- 04 - Gestion des classes
- 05 - Gestion de la facturation des études
- 06 - Gestion de la facturation de la cantine
- 07 - Gestion du ramassage scolaire
- 08 - Gestion des comptes clients

**Open Ecole**

Tableau de bord Généré le 12 nov. 2020 à 12:43:20

**Associations aux exigences**

Catégorie	Pourcentage	Nombre
CT associés à aucune exigence	34.8%	8
CT associés à une exigence	47.8%	11
CT associés à plus d'une exigence	17.4%	4

**Statut**

Catégorie	Pourcentage	Nombre
En cours de rédaction	13%	3
À approuver	30.4%	7
Approuvé	43.5%	10
Obsolète	8.7%	2
À mettre à jour	4.35%	1

**Importance**

Catégorie	Pourcentage	Nombre
Faible	26.1%	6
Moyenne	17.4%	4
Haute	17.4%	4
Très haute	39.1%	9

**Pas de test**

Catégorie	Pourcentage	Nombre
CT avec aucun pas de test	87%	20
CT avec 11 à 20 étapes	8.7%	2
CT avec 21 étapes ou plus	4.35%	1

Total des cas de test : **23**

# PAS DE TEST

The screenshot displays the Squash test management interface. On the left is a sidebar with navigation options like 'Exigences', 'Cas de test', 'Campagnes', 'Pilotage', 'Automatisation', 'Administration', 'Bugtrackers', 'Aide', 'Jalons', and 'Filtre projets'. The main area shows a test case 'Ajout d'une école' (ECOL\_02) with a status of 'Approuvé'. Below the title, there are status indicators for 'Approuvé', 'Haute', and 'Échec'. The 'Prérequis et Pas de test' section is expanded to show prerequisites and test steps.

**Ajout d'une école**  
 ECOL\_02  
 ● Approuvé ● Haute ● Échec

**Prérequis et Pas de test**

▼ PRÉREQUIS  
 L'application est installée.

1 ▶ Appel de "Connexion BDD - avec paramètres" avec le jeu de données "Chrome"

▼ Appel de "Connexion application - profil Consultation" avec le jeu de données "Choisir un jeu de données"

	ACTION	RÉSULTAT ATTENDU
2	1 Saisir l'adresse URL : http://nom_du_serveur/openmairie/openecole	La page d'authentification s'affiche.
	2 Saisir le nom d'utilisateur : <b>consultant</b> Saisir le mdp : <b>consultant</b>	L'utilisateur est authentifié et accède à la page d'accueil de l'application.

▼ ACTION RÉSULTAT ATTENDU

3 Dans le menu vertical situé à gauche de l'écran, aller dans Gestion > Ecoles. La page de gestion des écoles s'affiche.

▼ ACTION RÉSULTAT ATTENDU

Cliquer sur le bouton [+]. La page de création d'une école s'affiche.

4 PIÈCES JOINTES  
 Cahier de test-060820201617.docx 297,24 Ko Ajouté le 29/09/2020 10:36

EXIGENCES ASSOCIÉES  
 Demo Squash ECOL\_01 Accès à l'écran de gestion des écoles

▼ ACTION RÉSULTAT ATTENDU

# ASSOCIATION CAS DE TEST - EXIGENCES

The screenshot displays the Squash test management interface. The main window is titled "Accès au formulaire de création" and shows a test case "ECOL\_01" with a status of "À approuver". The interface is divided into several sections:

- Exigences vérifiées par ce cas de test:** A table showing the association between the test case and requirements.
- Paramètres et jeux de données:** A table listing test data sets.
- Cas de test appelé par:** A table listing other test cases that call this one.

**Table 1: Exigences vérifiées par ce cas de test**

#	PROJET	RÉFÉRENCE	EXIGENCE
1	Demo Squash	CLAS_02	Accès au formulaire de création d'une classe

**Table 2: Paramètres et jeux de données**

NOM	URL	env_name
Firefox	http://www.openecole.fr	Qual
IE	http://www.openecole.fr	Dev

**Table 3: Cas de test appelé par**

#	PROJET	RÉFÉRENCE	CAS DE TEST
Aucun élément à afficher			

The interface also features a sidebar with a navigation menu and a right-hand pane showing a "Référentiel des exigences" (Requirements Reference) tree. The tree lists various requirements categories such as "01 - Connexion à OpenEcole", "02 - Gestion des écoles", "03 - Inscription des élèves", etc., with specific test cases like "CONN\_01 - Login ou mot de passe incorrect" and "ECOL\_01 - Accès à l'écran de gestion des écoles" highlighted.

# RECHERCHE DE CAS DE TEST

SQUASH TM

- Exigences
- Cas de test**
- Campagnes
- Pilotage
- Automatisation
- Administra...
- Bugtrackers
- Aide
- Jalons
- Filtre projets

SO

### Recherche de cas de test

Rechercher Nouvelle recherche

+ Ajouter un critère

Périmètre : Demo Squash

Importance : Moyenne, Très haute

### « Résultats de recherche

#	PROJET ↑	ID	RÉFÉRENCE ↑	NOM ↑	ST.	IMP.	NATURE	TYPE	AUTOM.	CRÉÉ PAR	MODIFIÉ PAR
1	Demo Squash	27	CANT_01	Affecter des ...	●	⚡	Non définie	Non défini	-	admin	jjjoly
2	Demo Squash	28	CANT_02	Facturer des ...	●	⚡	Non définie	Non défini	-	admin	jjjoly
3	Demo Squash	20	CLAS_01	Créer un ens...	●	⚡	Non définie	Non défini	-	admin	jjjoly
4	Demo Squash	19	CLAS_02	Affecter des ...	●	⚡	Non définie	Non défini	-	admin	
5	Demo Squash	2	CONN_01	Connexion B...	●	⚡	Sécurité	Non défini	-	admin	admin
6	Demo Squash	3	CONN_03	Connexion a...	●	⚡	Non définie	Évolution	-	admin	admin
7	Demo Squash	6	ECOL_01	Accès au for...	●	⚡	Sécurité	Non-régression	-	admin	admin
8	Demo Squash	7	ECOL_02	Ajout d'une ...	●	⚡	Fonctionnel	Non-régression	-	admin	admin
9	Demo Squash	13	ECOL_06	Recherche si...	●	⚡	Fonctionnel	Non-régression	-	admin	admin
10	Demo Squash	17	ELEV_01	Inscrire un él...	●	⚡	Non définie	Non défini	-	admin	jjjoly
11	Demo Squash	22	FACT_02	Affecter des ...	●	⚡	Non définie	Non défini	-	admin	
12	Demo Squash	25	FACT_03	Facturer des ...	●	⚡	Non définie	Non défini	-	admin	

1 - 12 / 12

50

# ADMINISTRATION

The screenshot displays the Squash administration interface. On the left is a dark sidebar with navigation options: Exigences, Cas de test, Campagnes, Pilotage, Automatisation, Administration, Bugtrackers, and Aide. The main area is titled 'Gestion des projets' and shows a list of projects. The selected project, 'MAD\_Demo Xsquash', is detailed in the right-hand pane.

**Gestion des projets**

#	NOM
26	MAD_Demo Xsquash
27	MAD_Poubelle
28	MBE_Demo Squash
29	MBE_Demo Xsquash
30	MBE_Poubelle
31	MODELE Demo Squash
32	MODELE Demo Xsquash
33	Open Ecole
34	SLA_Demo Squash
35	SLA_Demo Xsquash
36	SLA_Test
37	SOU_Bac à sable
38	SOU_Demo Squash
39	SOU_Demo Xsquash
40	Squashteam
41	TBR_Demo Squash
42	TBR_Demo Xsquash
43	TBR_Demo_Squash_Cucu
44	TBR_Redmine
45	YBO_Demo Squash
46	YBO_Demo Xsquash

**MAD\_Demo Xsquash**

**Informations**

Modèle associé	MODELE Demo Xsquash	Création	30/01/2019 11:24 (mdelobeau)
Étiquette	(Cliquer pour éditer...)	Modification	23/01/2020 18:27 (admin)
Description	Cliquez pour renseigner une description.		

**Bugtracker**

Bugtracker: Jira BT

Nom du projet: MAD\_Demo Xsquash x mdelobeau x + Ajouter projet

**Habilitations**

#	UTILISATEUR/ÉQUIPE	PROFIL	TYPE
1	Marion Delobeau (mdelobeau)	Chef de projet	Utilisateur
2	testeur1 (testeur1)	Testeur référent	Utilisateur
3	testeur2 (testeur2)	Testeur avancé	Utilisateur
4	testeur3 (testeur3)	Designer de tests	Utilisateur

**Exécution**

Modification des cas de test pendant l'exécution:

Statuts optionnels: Non testable  Arbitré

**Automatisation**