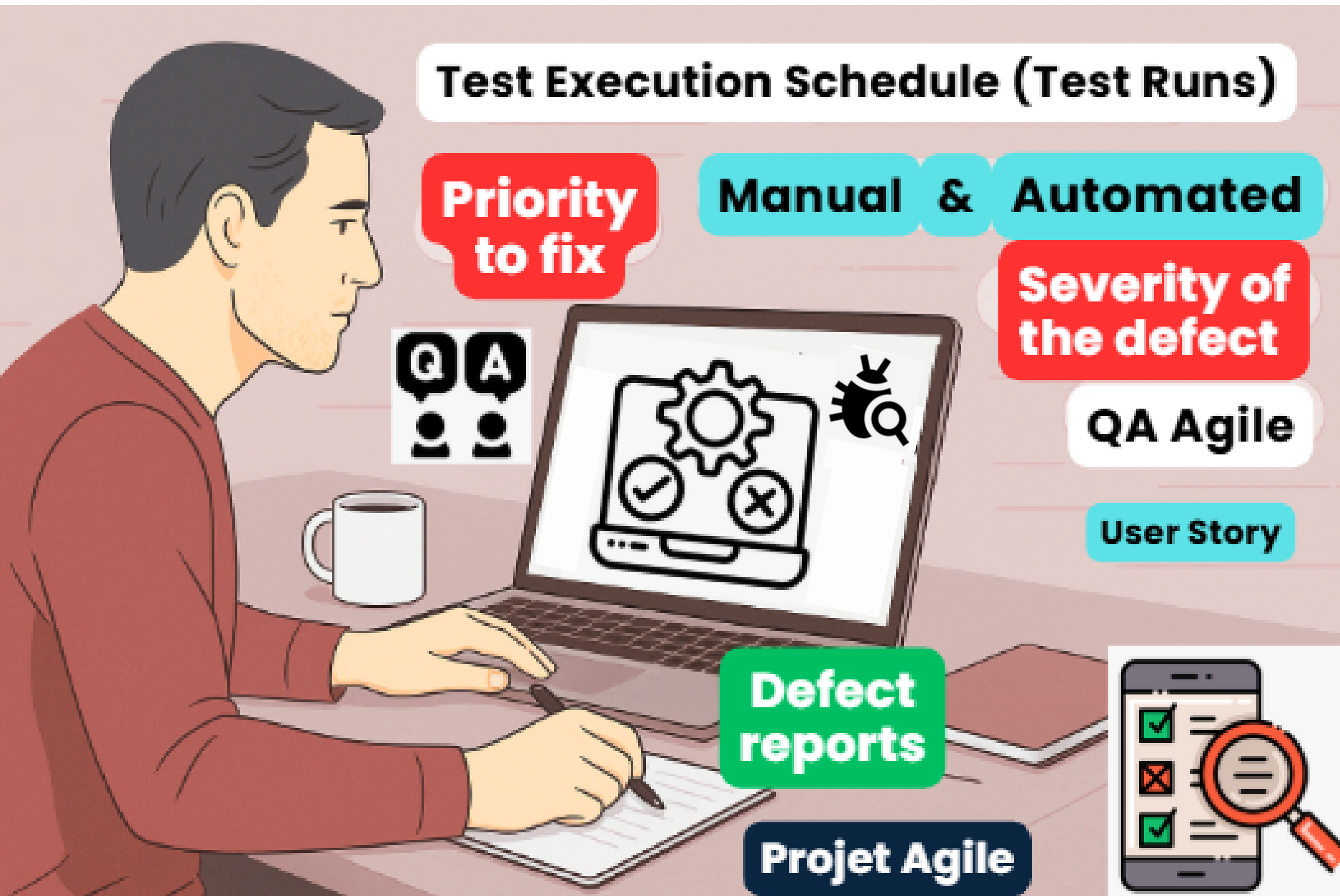


L'exécution des tests



Donner vie à la stratégie QA



 Hamid ZEMRANY
QA Engineer Strategist – ISTQB® • TMAP®

L'exécution des tests, le moment de vérité de la stratégie QA

Et si la **vraie** valeur du test
résidait dans l'**observation**
du **comportement réel**
du **système** ?

L'**exécution** des tests **consiste à**

- **lancer** les tests planifiés,
- **observer** les résultats,
- **agir** sur les anomalies détectées.
- **Déclencher** le processus de gestion des défauts

Objectif final : produire des informations fiables
sur la qualité du système.

Qu'est-ce que l'exécution des tests ?

L'exécution des tests comprend :

- Lancement des tests selon le **calendrier d'exécution**
- Comparaison des **résultats réels** aux **résultats attendus**
- **Enregistrement** des résultats
- **Analyse des anomalies** (gestion des défauts)

Les tests peuvent être :

- **manuels** ou **automatisés**
- réalisés en **binôme** ou en **test continu**

Étapes de l'exécution

- **Déclenchement** des procédures de test
- **Observation** et **documentation** du comportement réel
- **Vérification** de la conformité avec les attentes
- **Journalisation** complète des résultats

Objectif:

- détecter les écarts (défaillances)
- identifier d'éventuels défauts.

Analyse des résultats

Chaque test exécuté produit un **résultat réel**.
Ce résultat est comparé au **résultat attendu**.

En cas d'écart :

→ on parle de **défaillance**.

Les **causes racines** sont identifiées
par l'**analyse des causes racines**

Quand signaler une anomalie ?

Défaillance observée

→ Anomalie à signaler

Mais **ce n'est pas toujours un défaut réel.**

Il peut aussi s'agir d'un :

- Faux positif
- Problème de données
- Problème d'environnement
- Besoin de clarification fonctionnelle

→ Processus de gestion des défauts

Gestion des défauts

Traitement des anomalies signalées,
du signalement à la clôture

Une bonne gestion des défauts permet de :

- Faciliter les corrections
- Suivre la qualité du produit
- Proposer des améliorations

Ce workflow commun doit être suivi
par toutes les parties prenantes.

Cycle de vie d'un défaut

Un défaut suit généralement ce workflow :

- Signalement
- Analyse et reproduction
- Classification (sévérité, priorité)
- Décision : correction ou non
- Correction
- Validation
- Clôture

Statuts typiques :

Ouvert, En cours, Corrigé, Fermé,
Rejeté, Dupliqué, Ré-ouvert...

Rapport de défaut

Un rapport de défaut clair doit inclure :

- ID unique
- Titre + résumé
- Date + auteur + organisation
- Objet de test + environnement
- Contexte du test
- Description de la défaillance
- Logs, captures, données
- Résultat attendu vs réel
- Sévérité + priorité
- Statut + références

Les Outils de gestion peuvent remplir certains champs automatiquement (ID, date, statut)

Rôle stratégique des défauts

Les défauts ne sont pas que des erreurs à corriger.

Ils sont aussi :

- Des **indicateurs** de qualité produit
- Des **retours utiles** pour améliorer le processus
- Des **éléments-clés** pour ajuster la stratégie QA

Bien gérés, ils deviennent des leviers de performance.

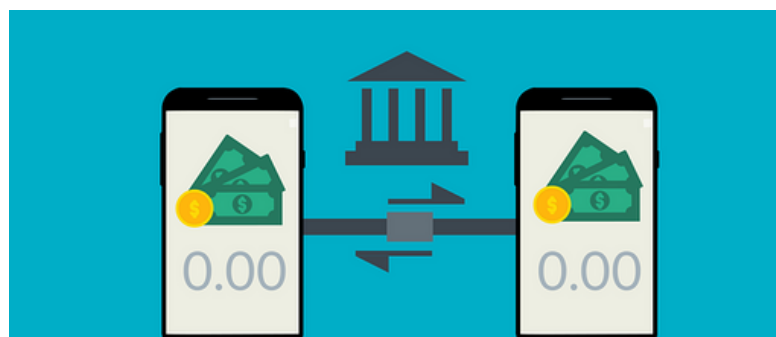
Specification by Example

Projet Agile Sprint 3

Nouvelle fonctionnalité :
Virement Instantané
Interbancaire via l'app mobile.

Contexte

Bancaire



Contexte du projet

Projet :

- Application bancaire mobile

Fonction critique :

- Virement instantané

Besoin utilisateur :

- Gain de temps, autonomie, sécurité

Objectif du Sprint 3

Déployer la fonctionnalité :
virement instantané interbancaire

 Une fonctionnalité
à forte valeur ajoutée
dans le parcours client.

Contexte Agile

Le QA travaille étroitement avec :

- PO
- Dév
- UX/UI designers
- ...

Avant d'exécuter :

Préparation minimale

Pour chaque User Story prête,
le QA s'assure que :

- Les critères d'acceptation sont clairs
- Les scénarios Gherkin sont définis
- L'environnement de test est disponible

User Story

En tant qu'utilisateur de l'app mobile,
Je veux transférer de l'argent instantanément
vers un compte externe,
afin de gérer mes urgences sans me déplacer.

Critères d'acceptation (Gherkin)

Scénario : Virement instantané

Étant donné que l'utilisateur est connecté

Et qu'un bénéficiaire est validé

Quand il saisit un montant

Et valide avec authentification forte

Alors le montant est débité instantanément

Et une confirmation s'affiche : "Virement effectué"

Objectif du QA Agile

Pendant l'exécution des tests, il vise à :

- **Vérifier** le comportement nominal
- **Tester** les limites fonctionnelles
- **Contrôler** les cas d'échec et anomalies

Exécution des tests


Souvent une combinaison de tests :

- Manuels fonctionnels
- Automatisés
- Techniques/API
- UI/UX

Test 1:

Cas Nominal

Virement Réussi

- **Préconditions** : Bénéficiaire validé, solde suffisant
- **Étapes** : Connexion → Montant → Validation → Authentification forte
- **Résultat attendu** : Confirmation instantanée
- **Résultat obtenu**  : Conforme
- **Statut** : Test OK

Tests effectués sur environnement pré-prod stable + authentification MFA simulée

Test 2:

Cas d'échec

Solde insuffisant

Préconditions : Solde < montant saisi

Étapes : Connexion → Montant → Validation

Résultat attendu : Message d'erreur clair : "Solde insuffisant"

Résultat obtenu ✗ : App crash

Statut : Test KO

Gestion des défauts



Si un écart est constaté
→ création d'un **ticket de bug**

Le bug doit être :

- Reproductible
- Documenté (étapes, contexte, logs éventuels)
- Priorisé (bloquant, majeur, mineur)

Outils : Jira (avec workflow QA → Dev → QA),
Screenshots, Logs, DevTools, ...

Réexécution après correction

Objectif :

- Vérifier que le bug est corrigé
- Contrôler la régression

Résultat obtenu  : Conforme

Statut : Test 2 validé après correction

Mise à jour du statut des tests dans les outils

Reporting

À la fin du sprint (ou du cycle de test) :

- Suivi des tests **passés / échoués / bloqués**
- Liste des **défauts ouverts** et **corrigés**
- **Feedbacks** partagés lors de la Sprint Review

Le quotidien QA Agile

En résumé, entre autres il :

- **Prépare** ses cas de test dès la Refinement
- **Exécute** tests manuels en début de sprint
- **Coordonne** avec l'automatisation
- **Suit** les anomalies
- **Réexécute** les tests post-correction
- **Rapporte** l'état d'avancement qualité en fin de sprint

Dans le respect et la bonne humeur

Le + en Agile :

Feedback rapide = valeur livrée

Grâce à l'exécution rapide des tests :

- Les défauts sont détectés tôt
- La livraison est plus fiable
- La valeur métier fonctionnelle est assurée à chaque sprint



**Tester, apprendre,
corriger, améliorer**



L'exécution des tests : là où la qualité rencontre la réalité.

*Tu veux des tests qui parlent métier,
qui détectent l'essentiel
et qui construisent la confiance
sprint après sprint ?*

*Alors pense collaboration,
traçabilité et réactivité.*

Et n'oublie jamais :

*☞ Un bug non remonté
est un risque non maîtrisé.*

Tu as aimé l'exécution des tests?



Like



Commente



Partage avec ton réseau



Enregistre pour ton prochain ...

HAMID ZEMRANY



QA Engineer
QA Strategist & Agile Enthusiast

ISTQB® • TMAP®

Testing & Automation
Web • Mobile • API



/hamid-zemrany-qa-strategist



hamid.zemrany@gmail.com