

Faire parler tes outils entre eux

Prototyper une automatisation avec contrôle humain et preuve de fonctionnement

Module BR4 - FAIRE - 7 min - version 1.1

À vérifier à l'écran avant de suivre ce module

Ces capacités dépendent du produit et de l'offre et restent ouvertes jusqu'à observation réelle.

- Outils d'automatisation : [À VÉRIFIER : quels services d'automatisation sont utilisables avec l'assistant visé, et à quel coût]
- Déclencheurs disponibles : [À VÉRIFIER : quels événements peuvent déclencher un traitement - réception d'un message, dépôt d'un fichier, heure fixe]
- Coût à l'usage : [À VÉRIFIER : le traitement est-il facturé à la tâche, et selon quel compteur ?]

Volatilité V3 - relevé réel requis avant publication.

Prix, limites et noms de menus changent sans prévenir. Si ce que tu vois à l'écran ne correspond pas, c'est l'écran qui a raison.

Une tâche répétitive peut devenir un bon candidat à l'automatisation. Mais ni sa fréquence ni sa durée ne suffisent à décider : il faut aussi regarder la conséquence d'une erreur, la possibilité de contrôle et le coût d'entretien.

Ce que tu sauras faire en sortant

- Décrire une automatisation sans choisir d'abord un produit.
- Séparer déclencheur, traitement et sortie.
- Tester une version où le résultat revient vers toi avant toute action externe.
- Mesurer tes propres corrections et décider à partir de ces observations, sans seuil universel inventé.

L'essentiel, en trente secondes

Trois éléments :

- Déclencheur - ce qui lance le processus.
- Traitement - ce qui transforme la matière.
- Sortie - où arrive le résultat et qui peut être affecté.

Pour le premier prototype Solutio, la sortie revient vers toi. Cette règle de prudence réduit l'impact d'une erreur pendant que tu observes le comportement.

Il n'existe pas dans ce module de règle du type « un mois », « quinze minutes par semaine » ou « une erreur sur dix ». Tu fixes une période d'observation suffisante pour rencontrer plusieurs cas représentatifs de ta tâche et tu notes ce qui s'est réellement passé.

Décrire avant de construire

``text Quand [DÉCLENCHÉUR], je veux que [TRAITEMENT]. Le résultat doit arriver [SORTIE]. Si le traitement est incorrect, [CONSÉQUENCE]. Je le détecte grâce à [CONTRÔLE / SIGNE DE VIE]. Aujourd'hui, la tâche est réalisée [FRÉQUENCE OBSERVÉE] et prend [TEMPS MESURÉ CHEZ MOI]. ``

Les deux derniers champs sont tes mesures, pas des chiffres fournis par le module.

Micro-défi

Quatre sorties fictives :

- brouillon stocké dans un dossier de test que tu relis ;
- message envoyé automatiquement à un client ;
- publication automatique sur un site public ;
- modification automatique d'un montant dans un outil de facturation.

Attendu pour le premier prototype : la 1. Les autres ont un impact externe ou engageant et passent d'abord par BR3 et un cadre validé.

La phase d'observation

Pendant le prototype, note au minimum :

- nombre de sorties produites ;
- sorties acceptées sans modification ;
- sorties corrigées ;
- sorties rejetées ;
- type de correction ;
- incident ou absence de sortie attendue.

Le but n'est pas de fabriquer un « taux acceptable » universel. La même fréquence d'erreur n'a pas la même conséquence dans un brouillon interne et dans une action vers un tiers.

Décision à la fin :

- garder validation humaine ;
- corriger le traitement et retester ;
- réduire le périmètre ;
- renoncer ;
- ou envisager une automatisation supplémentaire si le risque et le contrôle ont été explicitement acceptés dans le cadre compétent.

Prévoir un signe de vie

Une automatisation peut ne pas produire le résultat attendu sans que tu t'en aperçoives immédiatement. Le module recommande donc un signal observable : compteur, journal, statut, notification ou autre trace réellement disponible dans le produit.

Cette recommandation ne signifie pas que les automatisations « échouent toujours en silence ». Elle signifie : ne dépends pas du silence comme preuve que tout va bien.

Exemple fictif : un format d'entrée change et ton traitement ne retrouve plus le champ attendu. Si ton processus produit un compteur ou un journal, l'écart peut devenir visible.

Le cas où ça se passe mal

Cas reconstitué.

Le traitement continue à s'exécuter mais ne récupère plus l'information attendue. Aucun contrôle n'a été prévu. Le problème n'est découvert qu'à la relecture tardive.

Correction : définir avant le test quel signal montre qu'une exécution a eu lieu et quel signal révèle une sortie vide ou inhabituelle.

Ce que ce module ne règle pas

- Il ne choisit aucun outil ni ne donne son prix.
- Il ne garantit pas qu'un traitement automatisé soit plus fiable qu'un traitement manuel.
- Il ne fournit aucun seuil universel de durée, fréquence, taux d'erreur ou période d'essai.
- Il ne remplace pas BR3 pour les actions vers des tiers, l'argent ou les données sensibles.
- Il ne prouve pas qu'un compteur ou journal est disponible : c'est une capacité produit à observer.

Quand tu t'arrêtes

- La sortie agit sur un tiers, de l'argent ou un contenu public sans contrôle approprié.
- Tu ne peux pas dire ce qui serait affecté en cas d'erreur.
- Aucun mécanisme observable ne permet de savoir si le processus a fonctionné.
- Les données d'entrée n'ont pas passé le tri F3.
- Tu n'as pas encore suffisamment d'observations pour comprendre les corrections typiques.

À toi

Choisis une tâche réelle mais utilise une sortie de test interne.

- Mesure toi-même le temps et la fréquence actuels.
- Décris déclencheur / traitement / sortie / conséquence / contrôle.
- Exécute plusieurs cas représentatifs avec validation humaine.
- Note chaque correction et rejet.
- Décide : garder, corriger, réduire, renoncer ou retester.

C'est réussi si ta décision finale cite tes propres observations et non un seuil fourni par le module.

Vérifie ton geste

- Tu n'as qu'un seul exemple réussi. Insuffisant pour décider d'une automatisation externe ; continue l'observation.
- Une sortie part vers un client. Le premier prototype revient vers toi ; BR3 avant toute action externe.
- La tâche prend peu de temps chez toi. Cela peut réduire l'intérêt, mais aucune durée du module ne tranche à ta place.
- Tu corriges plusieurs sorties. Décris les corrections et améliore le traitement avant de retirer la validation.
- Aucun signal ne montre qu'une exécution a eu lieu. Ajoute un contrôle observable ou garde le processus manuel.

Seuil de réussite : 4 sur 5, questions 2 et 5 obligatoires.

Selon ce que tu obtiens

- Beaucoup de corrections garde la validation et travaille le traitement.
- Peu de corrections mais forte conséquence possible le risque peut justifier de garder la validation malgré tout.
- Aucun signal de fonctionnement ne passe pas en automatique.
- Le montage prend plus d'effort que le problème ne le mérite renonce ; c'est une décision valide.

Fiche mémo

- Déclencheur · traitement · sortie.
- Premier prototype sortie vers toi.
- Temps/fréquence = tes mesures, pas celles du module.
- Observe acceptations, corrections, rejets, incidents.
- Pas de seuil universel : la conséquence décide.
- Signe de vie observable avant automatisation supplémentaire.

Ce que tu viens de pratiquer

- Tu as supprimé les faux seuils de durée, fréquence et taux d'erreur.
- Tu as construit une décision à partir d'observations réelles de ton processus.
- Tu as prévu une preuve de fonctionnement au lieu de supposer que l'absence d'alerte signifie succès.

Sources et statut : docs/bloc-brancher-PROVENANCE.md`. BR4 reste S2 et brouillon ; les trois marqueurs produit restent ouverts.