

# La Checklist pour se lancer dans l'automatisation avec n8n - Par Tizy

 Objectif : vérifier que vous (ou votre équipe) êtes prêts à construire, tester et maintenir des automatisations robustes avec n8n, sans mettre vos données ou vos opérations en danger.

## Compétences logiques & conception de workflows

- **Compréhension de la logique algorithmique**  
Êtes-vous à l'aise avec les notions de séquence, conditions (if/else), boucles, branchements, erreurs, sorties multiples ?
- **Capacité à décomposer un process métier en étapes claires**  
Pouvez-vous traduire un flux métier en enchaînement logique (sans "trous" ou ambiguïtés) ?
- **Gestion des cas d'erreurs / cas limites**  
Que se passe-t-il si une API ne répond pas ? Si une donnée est vide ? Si un fichier est mal nommé ?
- **Notions de test & débogage progressif**  
Savez-vous tester un nœud isolé, utiliser les logs de n8n, et valider une sortie avant d'aller plus loin ?
- **Compréhension du fonctionnement d'un "trigger"**  
Savez-vous d'où part votre automatisation, comment elle est déclenchée (manuellement, à l'heure, via webhook...) ?

## Maîtrise des formats de données & structures JSON

- **Lecture et compréhension du format JSON**  
Êtes-vous capable d'identifier un objet, une clé, une valeur, une liste imbriquée ?
- **Manipulation de données dans n8n (via expressions)**  
Savez-vous récupérer dynamiquement une donnée avec `{{ $json["nom"] }}` ou des variantes plus complexes ?
- **Identification d'erreurs de structure**  
Êtes-vous capable de repérer un champ manquant, mal typé ou mal formaté dans une réponse API ou un output de nœud ?
- **Savoir nettoyer, reformater ou filtrer les données**  
Par exemple, extraire le prénom à partir d'un champ nom complet, convertir une date, supprimer les doublons, etc.



## Compétences API & intégrations externes

- **Compréhension des principes d'API REST**  
Connaissez-vous les notions de GET, POST, PUT, DELETE, endpoints, headers, body, tokens ?
- **Utilisation d'outils pour tester les APIs**  
(Postman, Insomnia, test dans n8n) pour vérifier les réponses d'un appel API.
- **Gestion de l'authentification (token, OAuth2, clé API)**  
Êtes-vous capable de configurer l'authentification dans un nœud HTTP ou via les credentials de n8n ?
- **Interprétation des réponses API (erreurs, statuts, messages)**  
Savez-vous lire une réponse HTTP 200, 401, 500, et en déduire ce qu'il faut ajuster ?
- **Création de requêtes personnalisées avec le nœud HTTP Request**  
En cas de connecteur absent dans n8n, pouvez-vous construire manuellement une requête fonctionnelle ?



## Bases en Javascript (ou volonté de monter en compétence)

- **Lecture et adaptation d'un script court en Javascript**  
Savez-vous lire un forEach, manipuler des chaînes, gérer des dates ou des objets simples ?
- **Compréhension des erreurs courantes JS**  
Savez-vous identifier une variable mal déclarée, une propriété undefined, une erreur de syntaxe ?



## Utilisation de n8n (interface, logique, limites)

- **Connaissance des différents types de nœuds (Trigger, Function, Merge, IF, Set, etc.)**  
Savez-vous les combiner efficacement pour construire un scénario cohérent ?
- **Capacité à versionner ou dupliquer un workflow**  
Travaillez-vous sur des copies avant de modifier un scénario en production ?
- **Compréhension des limites de n8n auto-hébergé**  
Système de gestion des ressources, stockage temporaire, erreurs silencieuses si non monitorées...

- **Connaissance des coûts potentiels (exécution en boucle, surcharge CPU)**

Savez-vous qu'un mauvais scénario peut consommer des milliers d'opérations et alourdir la facture ou planter le serveur ?



## **Organisation, documentation et pilotage de projet**

- **Documentation claire des workflows**

Chaque scénario est-il nommé, commenté, explicité ? (Objectif, déclencheur, logique, données manipulées)

- **Plan de tests / sandbox avant production**

Avez-vous un environnement ou des jeux de données de test ?

- **Stratégie de monitoring ou d'alerte en cas d'erreur**

Des alertes Slack / email sont-elles prévues en cas d'échec de scénario ?

- **Définition de la responsabilité en cas de bug**

Qui est en charge du suivi, de la correction, et de l'itération des workflows ?

- **Capacité à maintenir le scénario sur le long terme**

Avez-vous évité les dépendances à une seule personne non documentée ?

Attention : Cette checklist est un outil de discernement, pas un test. Si certaines des cases restent vides, ce n'est pas un problème mais un bon signal pour demander un accompagnement ou planifier une montée en compétence. Automatiser, c'est aussi prendre soin de la stabilité de votre SI.