



# contournement

## Programme de formation

### Parcours vibe coding

*Maîtrisez toutes les compétences pour créer des applications web avec l'IA - sans écrire une seule ligne de code*

*Mis à jour le 20/04/2026*

## PRÉSENTATION GÉNÉRALE

Le Vibe Coding, c'est la capacité à créer des applications en décrivant simplement ce qu'on veut à une IA. Mais derrière cette apparente simplicité, **créer une app qui fonctionne vraiment demande des bases solides** : comprendre l'IA avec laquelle on dialogue, structurer correctement ses données, maîtriser l'outil de création, et sécuriser son travail.

C'est exactement ce que ce parcours vous donne. Il suit le **cycle de vie complet d'un projet** : de la culture informatique indispensable pour se repérer dans le jargon du développement web, jusqu'au déploiement d'une application fonctionnelle en ligne, en passant par les IA génératives, l'architecture de données, les API et la gestion de versions.

Chaque module est autonome et peut être suivi indépendamment. Ensemble, ils forment un socle complet pour **créer, structurer, connecter, publier et sécuriser vos projets de Vibe Coding** en toute autonomie.

## PUBLICS-CIBLES

Tous types de publics, et en particulier :

- salarié-e-s en poste qui utilisent des outils informatiques, et cherchent à améliorer leur productivité ainsi que leurs compétences numériques
- demandeur-se-s d'emploi en recherche d'un enrichissement de leurs compétences numériques
- indépendant-e-s et porteur-se-s de projets entrepreneuriaux - traditionnels ou innovants -, qui utilisent les outils informatiques dans leur gestion et leur organisation

# ✓ PRÉREQUIS

Il n'y a pas de pré-requis de diplôme ou de qualification technique particuliers.

Les stagiaires doivent seulement maîtriser des compétences numériques de base, notamment :

- saisie au clavier
- navigation sur Internet
- utilisation basique de services professionnels et grand public sur le web (création de compte, publication de contenus, de posts, d'annonces, de commentaires, de messages, etc.)
- visioconférence (partage d'écran notamment)

## OBJECTIFS PÉDAGOGIQUES

Au terme de la formation, les stagiaires doivent être en mesure de mettre en application les compétences suivantes :

- Dialoguer efficacement avec les IA génératives pour produire des résultats utiles et fiables dans ses projets
- Concevoir et structurer une base de données solide et évolutive, adaptée aux besoins d'un projet réel
- Créer une application web complète avec Lovable, du cadrage initial au déploiement en ligne, sans écrire une seule ligne de code
- Sécuriser et versionner ses projets de développement assisté par IA avec Git et GitHub



# CONTENU

## Culture Informatique : pour un Vibe Coding éclairé

### Chapitre 0 : Introduction

- Tony vous présente le module et les objectifs

### Chapitre 1 : Histoire de l'informatique

- Une brève histoire d'Internet
- Une brève histoire de l'informatique personnelle

### Chapitre 2 : Découverte du Web

- Comment ça marche Internet ?
- Le fonctionnement du Web
- HTTP : comprendre l'essentiel en 4 minutes

### Chapitre 3 : Programmation et architecture web

- Quelques mots d'introduction
- Les familles de langages de programmation
- Quelles différences entre le frontend et le backend ?
- Introduction au HTML
- Découverte de la syntaxe HTML
- Découverte du langage CSS
- C'est quoi un algorithme ?
- Pour aller plus loin

### Chapitre 4 : Focus architecture et backend

- Architecture technique pour le Vibe Coding
- Le backend pour les débutants en partant de 0

- Comprendre SQL en 1 minute
- Pour aller plus loin sur le backend

## **Chapitre 5 : Git et contrôle de version**

- Quelques mots d'introduction
- Pour aller plus loin sur le contrôle de version

## **Chapitre 6 : Product management**

- Qu'est-ce que le product management ?
- PRD : Product Requirement Document
- Pour aller plus loin : Design UX/UI

## **Chapitre 7 : Sécurité**

- Sécuriser les données dans une app web
- Authentification : Le Guide Complet pour comprendre l'authentification Web
- Pour aller plus loin sur les bases de la sécurité

## **Chapitre 8 : Conclusion**

- Le mot de la fin
- Pour aller plus loin : le livre d'Alexis Kovalenko
- Quiz final

# **Prenez en main les IA génératives**

## **Embarquement**

- Allez, on embarque !

## **Chapitre 1 : Comprendre le phénomène des IA génératives**

- Démystifions un peu les IA génératives

- Un rapide historique
- Des modèles pour différents médias
- Quelques questions avant de commencer les mises en pratique

## **Chapitre 2 : Apprendre à communiquer avec les IA**

- Quelques mots d'introduction
- De la "completion" au "chat"
- Les bases du prompting
- Un élément clé : le contexte
- Pourquoi adopter une approche itérative ?
- Les "frameworks" pour écrire des prompts plus facilement
- La technique des shots
- La chaîne de pensées
- L'arbre de pensées
- Mise en pratique n°1
- Correction de la mise en pratique n°1
- Mise en pratique n°2
- Correction de la mise en pratique n°2

## **Chapitre 3 : Focus sur ChatGPT**

- Pourquoi ChatGPT ?
- Prendre en main l'interface
- Paramétrer et personnaliser ChatGPT
- Maximiser votre expérience de ChatGPT grâce aux projets personnalisés
- Travailler vos textes avec l'outil "canevas"
- Découvrir le puissant modèle GPT-4o, capable de raisonnements évolués et d'utilisation d'outils
- Créer vos propres assistants
- Donner de la connaissance à vos assistants
- Mise en pratique n°3
- Screencast de la mise en pratique n°3
- La plateforme d'OpenAI ("playground")

## **Chapitre 4 : Automatiser l'IA**

- Est-ce que l'IA, c'est de l'automatisation ?
- L'automatisation repose sur les API
- L'IA dans les outils no-code
- Automatiser l'IA avec les outils no-code
- Et les agents dans tout ça ?
- Mise en pratique n°4
- Screencast de la mise en pratique n°4

## **Chapitre 5 : Panorama des LLMs**

- Comment comparer des LLMs ?
- À la découverte de Claude (Anthropic)
- À la découverte de Mistral AI
- À la découverte de Gemini (Google)
- Les autres modèles

## **Conclusion**

- Des ressources pour aller plus loin
- Félicitations et merci — vous avez terminé la formation

# **Architecture de données**

## **Embarquement**

- Allez, on embarque !
- Xavier se présente

## **Chapitre 1 : Pourquoi l'architecture de données est essentielle**

- L'histoire d'une base chaotique
- Concevoir, ce n'est pas remplir
- Quiz de validation

## **Chapitre 2 : Penser et représenter sa structure — des concepts au modèle MCD**

- Le langage commun de la donnée : entités, attributs, relations
- Penser son modèle conceptuel pas à pas (MCD)
- Mise en pratique n°1
- Correction mise en pratique n°1

## **Chapitre 3 : Du texte au schéma ERD**

- Pourquoi passer du MCD à l'ERD ?
- Comprendre les clés primaires et étrangères
- Construire le schéma ERD de sa base de données
- Mise en pratique n°2
- Correction mise en pratique n°2
- Les conventions de nommage d'une base de données

## **Chapitre 4 : Du modèle logique à la base concrète**

- Du modèle à la base : traduire vos ERD dans vos outils préférés
- Implémenter son ERD dans Airtable
- Implémenter son ERD dans TimeTonic
- Implémenter son ERD dans Notion

## **Chapitre 5 : Les relations complexes et les tables de jonction**

- Les relations multiples : pourquoi elles posent problème ?
- Mise en pratique n°3
- Correction mise en pratique n°3

## **Chapitre 6 : L'art du rangement — nommage, structure et normalisation**

- La première forme normale (1NF)
- La deuxième forme normale (2NF)
- La troisième forme normale (3NF)
- Adapter la normalisation aux outils no-code

- Mise en pratique n°4
- Correction mise en pratique n°4
- Implémenter son ERD normalisé dans Airtable

## **Chapitre 7 : Mise en pratique finale**

- Mise en pratique n°5
- Correction mise en pratique n°5

## **Conclusion : des bases no-code aux backends avancés**

- L'architecture, un langage universel
- Ces principes d'architecture à l'épreuve de Xano
- Ces principes d'architecture à l'épreuve de Supabase
- Vous n'avez pas appris un outil, vous avez appris à penser la donnée

# **Les API**

## **Embarquement**

- C'est quoi cette formation aux APIs
- Allez, on embarque !

## **Chapitre 1 : Quelques illustrations pour commencer en images**

- Une API, c'est comme aller aux impôts
- Une API, c'est comme aller au resto
- En vrai, une API c'est une interface informatique

## **Chapitre 2 : Mais pourquoi les APIs au fait ?**

- Pourquoi a-t-on besoin d'une interface standardisée ?
- On pourrait pas juste faire pleins d'applications ?

## **Chapitre 3 : Les fondamentaux techniques (mais en douceur)**

- L'architecture client/serveur (mais sans le resto)
- Le protocole HTTP (c'est HYPER important)
- Le format de données JSON (comme dans Vendredi 13 mais sans le A)
- Plein d'exemples de JSON (pour devenir un.e [tueur.se](https://tueur.se) de ce format)
- L'authentification (car c'est pas open bar dans toutes les APIs)

#### **Chapitre 4 : Comprendre par la pratique (et grâce au no-code)**

- Un petit point de vocabulaire
- La pratique, ça commence par la documentation
- Appeler des APIs à la main, comme les pros
- Envoyer des données à APIs à la main, toujours comme des pros
- Appeler des APIs avec Zapier (parceque à la main ça va 5min)
- Faire son webhook avec Make (ex-integromat)
- Mise en pratique
- Screencast de la mise en pratique
- Comment ChatGPT peut vous aider à travailler avec des APIs

#### **Chapitre 5 : Découvrez le standard OAuth2**

- Une introduction à l'OAuth2
- Le "flow" de l'OAuth2 expliqué en détail
- Une illustration par l'exemple
- Des exemples d'outils no-code pour faire des requêtes en OAuth2
- Les agrégateurs d'APIs

## **Expressions régulières (Regex)**

### **Embarquement**

- Allez, on embarque !

### **Chapitre 1 : À la découverte des expressions régulières**

- C'est quoi une expression régulière ?
- A quoi ça peut servir ?
- Comment on s'en sert dans les outils no-code ?
- Attention, il y a plusieurs normes

## **Chapitre 2 : Les fondamentaux pour écrire des expressions régulières pertinentes**

- On écrit ensemble notre première regex
- Les classes de caractères
- Les quantificateurs
- Protéger les caractères spéciaux
- Les ancres
- La négation
- Les drapeaux
- Les groupes
- Le "pipe" | pour faire des choix dans les groupes
- Une antisèche pour récapituler les concepts principaux

## **Chapitre 3 : Mises en pratique dans les outils no-code**

- Mise en pratique n°1
- Screencast de la mise en pratique n°1
- Mise en pratique n°2
- Screencast de la mise en pratique n°2
- Extraire des données dans Zapier
- Manipuler des données dans Make
- Remplacer des caractères dans Notion

## **Chapitre 4 : Pour aller plus loin**

- Utiliser ChatGPT pour vous assister sur vos expressions régulières
- Découvrez RegexLearn pour continuer à pratiquer

## **Conclusion**

- Quelques ressources à explorer
- Félicitations, vous êtes arrivé.e au bout !

## **Lovable : De l'idée à l'app, grâce au Vibe Coding**

### **Embarquement**

- Nicolas se présente et introduit la formation
- Nicolas vous montre ce qu'il a créé avec Lovable
- Comprendre la tarification de Lovable (crédits, plans)
- Découvrir le tableau de bord de Lovable

### **Chapitre 1 : Cadrer et lancer son projet**

- Nicolas vous présente le projet fil rouge
- Bien cadrer son projet avant de prompter
- Lancer son projet avec le premier prompt
- Prendre en main l'interface Lovable

### **Chapitre 2 : Itérer et affiner son projet**

- Les bonnes pratiques pour prompter dans Lovable
- Toutes les façons de dialoguer avec Lovable (image, voix, sélection...)
- On itère en "live" pour ajouter de nouvelles fonctionnalités

### **Chapitre 3 : Connecter un back-end**

- Activer le back-end avec un prompt
- Comprendre Lovable Cloud (Supabase intégré)
- Gérer les utilisateurs et les droits d'accès

### **Chapitre 4 : Brancher des services externes**

- Intégrer de l'IA dans son app avec Lovable AI

- Connecter une API via les connecteurs natifs (ex : Slack)
- Brancher des API externes en s'aidant de la documentation

## **Chapitre 5 : Publier et mettre en ligne son app**

- Connecter son projet à GitHub (et pourquoi c'est utile)
- Déployer son app en ligne avec Lovable
- Associer un nom de domaine personnalisé
- Suivre les statistiques de son app

## **Chapitre 6 : Mise en pratique finale**

- Nicolas vous présente la mise en pratique finale
- Découvrez une version possible du projet

## **Chapitre 7 : Conclusion**

- Récap et perspectives : et maintenant ?
- Ressources pour aller plus loin
- Quiz final

# **Formation à Git**

## **Embarquement**

- Bienvenue
- Versionner ? Une petite explication rapide
- Un premier exemple
- Comprendre la ligne de commande
- Ressources sur la ligne de commande
- Votre première quête

## **Chapitre 1 : Les fondamentaux**

- Installer et configurer Git

- Un premier dépôt Git
- Le workflow de base
- Illustrations des 3 étapes du flux de base
- Récap des fondamentaux
- Votre deuxième quête

## Chapitre 2 : En ligne avec GitHub

- Son premier projet sur GitHub
- Cloner et synchroniser
- GitHub dans les éditeurs modernes
- L'outil GitHub Desktop

## Chapitre 3 : En mode collaboration

- Travailler à plusieurs
- Ne pas tout versionner
- Revenir en arrière
- Gérer les conflits
- Les branches
- Les Pull Requests
- L'IA comme assistant Git
- Votre quête finale

## Chapitre 4 : Pour terminer

- Conclusion
- Ressources pour aller plus loin



**28** heures soit 4 jour(s)

👉 dont 20 heures de théorie ou démonstration

👉 8 heures de mise en pratique

## MODALITÉS ET DÉLAIS D'ACCÈS

Un test de positionnement est réalisé en amont de la formation afin de s'assurer que les objectifs pédagogiques de la formation correspondent au besoin du stagiaire et que le niveau des prérequis est suffisant.

La formation est dispensée en **e-learning**, selon un principe **d'entrées et de sorties permanentes**, permettant aux participant-e-s de démarrer la formation à tout moment, après validation de l'inscription, dans le respect des délais de rétractation applicables (14 jours en cas de financement individuel).

## MOYENS TECHNIQUES ET D'ENCADREMENT

Les stagiaires disposent d'un accès à une **plateforme de formation en ligne** dédiée, leur permettant :

- de lire et naviguer entre les contenus pédagogiques textuels ;
- de consulter des vidéos pédagogiques et des présentations ;
- d'accéder à des **exercices de mise en pratique**, ainsi qu'à leurs corrections ;
- de compléter des **quiz d'évaluation** permettant de suivre l'avancée de l'acquisition des compétences ;
- d'accéder à des liens vers des ressources externes complémentaires.

La plateforme intègre plusieurs dispositifs d'accompagnement assurés par un **formateur dédié** :

- les stagiaires peuvent poser des **questions écrites directement sous chaque vidéo** de formation. Le formateur dédié y répond dans un délai maximum de **48 heures** ;
- les stagiaires peuvent également poser des questions via un outil de **questions-réponses asynchrones par vidéo (VideoAsk)**.
- Cet outil permet aux stagiaires de formuler leurs questions sous forme **textuelle, audio ou vidéo**, et de réaliser des **partages d'écran** afin de présenter leurs manipulations ou problématiques.
- Le formateur dédié peut répondre à son tour sous forme **écrite ou vidéo**, avec possibilité de **partage d'écran**, afin d'illustrer les réponses et démonstrations techniques.

En complément de l'accompagnement asynchrone, les stagiaires ont accès à des **sessions de mentorat collectif** organisées **chaque semaine**.

Les formateurs se tiennent à disposition dans une **salle de classe virtuelle ouverte**, permettant aux stagiaires de se connecter librement pour poser leurs questions relatives à la formation.

Un **courriel hebdomadaire** est adressé aux stagiaires afin de leur rappeler les créneaux horaires de ces sessions de mentorat collectif.

## MODALITÉS D'ÉVALUATION DE LA FORMATION

Le dispositif d'évaluation est composé d'un questionnaire, proposé en fin de session pour s'assurer de la bonne rétention des messages principaux de la formation. Il contiendra 10 questions, et les modalités de réussite en seront les suivantes : un score de 60% devra être atteint pour valider la formation.

Une attestation de réalisation est remise à chaque participant-e à l'issue de la formation.

Une attestation de réussite est remise aux participant-e-s satisfaisant les critères de réussite de la formation.



## MÉTHODES PÉDAGOGIQUES

Les modalités pédagogiques consistent en une alternance de phases de pédagogie :

- magistrale et par la démonstration, où les formateurs expliquent des éléments techniques et méthodologiques
- par la pratique, avec la réalisation d'exercices aboutissant à la production technique de livrables, produits avec les outils no-code
- active, avec des phases d'auto-apprentissage accompagné, suivis de correctifs, afin que les stagiaires soient acteurs de leur montée en compétence

# ACCESSIBILITÉ AUX PERSONNES EN SITUATION DE HANDICAP

Cette formation est ouverte au plus grand nombre, et notamment aux personnes en situation de handicap. Une attention particulière sera portée, dans le cadre du processus d'organisation des sessions, pour comprendre la nature de la situation de handicap et envisager les modalités d'adaptation ou de compensation les plus adéquates.

N'hésitez pas à nous indiquer les éventuelles situations de handicap de vos collaborateurs pour que nous en discutons, avec [notre référent handicap](#), en toute confidentialité.

## CONTACT

 **email : [formationpro@contournement.io](mailto:formationpro@contournement.io)**

Les Participants devront respecter les dispositions du règlement intérieur de Contournement, accessible sur notre site internet : <https://www.contournement.io/details/reglement-interieur>