



GUIDE PRATIQUE

# 6 mois pour devenir développeur web junior : roadmap complète

De la toute première ligne de code à ton premier contrat. Une méthode structurée, des outils concrets, un calendrier réaliste.



**00 Introduction**

À qui s'adresse ce guide, comment l'utiliser, ce que "junior" veut dire

---

**01 Comprendre le métier avant de se lancer**

Frontend, backend, fullstack, salaires, réalité du marché

---

**02 Les fondations — Mois 1**

HTML, CSS, Git, CLI, VS Code

---

**03 La logique de programmation — Mois 2**

JavaScript vanilla, DOM, événements

---

**04 Choisir sa spécialité — Mois 3**

Frontend vs Backend, critères de choix selon ton profil

---

**05 Monter en compétence sur sa stack — Mois 4**

React/Vue ou PHP/Symfony ou Python selon ta voie

---

**06 L'IA dans ton workflow de développeur**

Copilot, Claude, ChatGPT : apprendre et coder avec l'IA en 2026

---

**07 Construire son portfolio — Mois 5**

Quels projets faire, GitHub, README pro, déploiement

---

**08 Décrocher son premier job — Mois 6**

CV, LinkedIn, où postuler, pitcher sa reconversion

---

**— Conclusion**

Les pièges à éviter, les signaux que tu es prêt

# Ce guide est fait pour toi si tu pars de zéro

Pas de prérequis. Pas de diplôme technique exigé. Juste une méthode claire et un calendrier qu'on peut tenir.

Tu n'as jamais écrit une ligne de code de ta vie. Ou presque. Tu regardes des tutoriels depuis des semaines sans savoir vraiment par où commencer. Tu entends parler de reconversion dans le dev web mais tu ne sais pas si c'est réaliste pour toi, dans ta situation.

Ce guide répond à une seule question : **comment passer de zéro à développeur web junior en 6 mois, concrètement ?**

Pas de promesse sur le salaire. Pas de "tu peux le faire en 3 mois si tu es motivé". Une roadmap réaliste, structurée mois par mois, avec les bons outils et les bonnes ressources.

## Comment utiliser ce guide

Ce n'est pas un cours. C'est une carte. Il te montre le chemin, les étapes, les bifurcations. Le travail, tu le fais toi.

Chaque chapitre correspond à une phase de ta progression. Lis-les dans l'ordre la première fois, puis reviens sur les parties dont tu as besoin. À chaque étape, tu trouveras les ressources gratuites pour progresser et les repères pour savoir si tu es prêt à passer à la suite.

### À RETENIR

6 mois, c'est une durée réaliste si tu consacres entre 2 et 4 heures par jour à l'apprentissage. Moins que ça, prévois 9 à 12 mois. Plus que ça, tu peux aller plus vite.

## Ce que "junior" veut vraiment dire

Le titre de développeur junior attire de nombreuses entreprises qui cherchent des profils à former. Ce qu'on attend vraiment d'un junior sur le marché français :

- Comprendre et écrire du code lisible dans au moins une technologie (HTML/CSS, JavaScript, PHP, Python)
- Utiliser [Git](#) au quotidien sans supervision constante
- Lire de la documentation en anglais et déboguer de manière autonome
- Communiquer clairement sur ce qu'on fait, ce qui bloque, et pourquoi
- Avoir un ou deux projets concrets à montrer (portfolio, [GitHub](#))

### CE GUIDE NE TE PROMET PAS UN JOB

Il te donne les bases, la structure et les outils pour te mettre en position de décrocher un entretien. Le reste dépend de ton travail, de ta constance et du marché au moment où tu postules.