

Livret d'Apprentissage - Développeur Web

Ce livret est conçu pour t'accompagner dans ta progression, étape par étape. À chaque compétence acquise, tu peux cocher les cases correspondantes pour suivre tes progrès.

1. Premiers pas dans le développement web

- ☐ Découvrir HTML5 pour structurer des pages web
 - ☐ Comprendre les balises de base (div, p, h1, etc.) et les attributs
 - ☐ Apprendre CSS3 pour la mise en forme et le design
 - ☐ Utiliser Flexbox pour aligner des éléments en CSS
 - ☐ Apprendre Grid pour structurer des mises en page complexes
 - ☐ S'initier à Sass et au SCSS
 - ☐ S'initier à JavaScript pour l'interactivité (ajouter des alertes, manipuler du texte)
 - ☐ Apprendre à utiliser Git pour le versionnement du code
 - ☐ Héberger des projets sur GitHub et apprendre à collaborer avec Git
-

2. Devenir un développeur front-end

- ☐ Découvrir les bases de l'UX/UI (prototypage avec Figma ou Adobe XD)
 - ☐ Apprendre à utiliser la console du navigateur pour déboguer le code
 - ☐ Tester et déboguer du JavaScript via la console du navigateur
 - ☐ Utiliser le framework Bootstrap pour créer des designs responsives rapidement
 - ☐ Découvrir d'autres frameworks CSS (Tailwind CSS, Foundation, Bulma...)
 - ☐ Comprendre et appliquer le Responsive Design avec media queries
 - ☐ Approfondir JavaScript avec `getElementBy...` et `querySelector...` pour manipuler le DOM
 - ☐ Utiliser `addEventListener` pour gérer les événements (clics, touches de clavier)
 - ☐ Maîtriser les requêtes asynchrones avec `fetch` et les promesses
 - ☐ Utiliser `async/await` pour simplifier le code asynchrone
 - ☐ S'initier à npm pour gérer les dépendances dans un projet JavaScript
 - ☐ Découvrir React.js ou Vue.js pour créer des interfaces utilisateur dynamiques
 - ☐ Utiliser React.js ou Vue.js pour développer des composants réutilisables
 - ☐ Apprendre à organiser son code JavaScript avec des modules
-

3. Maîtriser le développement back-end

- ☐ Développer des pages dynamiques avec PHP
 - ☐ Apprendre à traiter des formulaires et stocker des données en PHP
 - ☐ Découvrir les bases de données relationnelles avec MariaDB ou MySQL
 - ☐ Maîtriser les requêtes SQL de base (SELECT, INSERT, UPDATE, DELETE)
 - ☐ Créer une base de données et gérer ses tables
 - ☐ Découvrir les bases de données NoSQL avec MongoDB
 - ☐ Appliquer les bases de la Programmation Orientée Objet (POO) en PHP
 - ☐ Comprendre le concept de Model-View-Controller (MVC) en PHP
 - ☐ Mettre en place une application PHP avec une base de données (back-end complet)
 - ☐ Héberger une application web sur une plateforme (PlanetHoster, O2Switch)
 - ☐ Utiliser PHPUnit pour créer des tests unitaires et valider le code PHP
 - ☐ Découvrir Docker pour isoler des environnements de développement
-

4. Créer des applications web complètes avec Symfony

- ☐ Prendre en main Symfony pour créer une application web
 - ☐ Développer des APIs REST avec Symfony (exposer des données via HTTP)
 - ☐ Apprendre à protéger les APIs avec des tokens JWT ou OAuth2
 - ☐ Approfondir SQL (vues, triggers, procédures stockées) pour des bases de données complexes
 - ☐ Appliquer des bonnes pratiques de sécurité (protection contre les attaques XSS, injections SQL, CSRF)
 - ☐ Finaliser une application web complète avec Symfony (front-end dynamique et back-end sécurisé)
 - ☐ Héberger une application web Symfony sur un environnement professionnel (AWS, VPS)
 - ☐ Utiliser Docker pour déployer et gérer l'environnement de production
 - ☐ Créer des tests d'intégration et fonctionnels avec PHPUnit
 - ☐ Ajouter des tests automatisés pour garantir la stabilité de l'application
 - ☐ Comprendre le concept de CI/CD (Intégration Continue / Déploiement Continu)
-

5. Se perfectionner et préparer son avenir professionnel

- ☐ Créer un portfolio personnel en ligne pour présenter ses projets
- ☐ Finaliser un ou plusieurs projets de fin d'études (exemple : e-commerce, réseau social, etc.)

- ☐ Préparer des présentations pour expliquer ses projets (choix techniques, défis)
 - ☐ Se préparer aux entretiens techniques (questions sur HTML, CSS, JavaScript, PHP, Symfony)
 - ☐ Améliorer ses compétences en gestion de projet avec les méthodologies agiles (Scrum, Kanban)
 - ☐ Apprendre à travailler efficacement en équipe avec Git, GitHub, Trello, et Jira
 - ☐ Identifier l'importance de la veille technologique et la pratiquer !!!
-

Fonctionnement du livret :

1. **Progrès personnel** : Utilise ce livret pour suivre chaque compétence que tu acquiers. Coche chaque étape lorsque tu te sens à l'aise avec le concept.
2. **Ressources** : Chaque compétence peut être accompagnée de ressources externes (tutos, livres, documentation) que tu pourras explorer à ton rythme.
3. **Auto-évaluation** : À la fin de chaque module, prends du recul et évalue tes points forts et tes axes d'amélioration.
4. **Objectifs à long terme** : Ce livret est conçu pour t'aider à rester concentré sur tes objectifs de développement à la fois pendant ta formation et après, dans ta carrière.