

Plan d'accompagnement web front et mobile

17 **Semaine 0 : Logique & Algorithmes**

 **Objectifs:** Poser les bases de la logique de programmation

Théorie (approche visuelle & pratique)

- C'est quoi un algorithme ? (Exemples du quotidien, ex : recette de cuisine)
- Les types de données : chaînes, nombres, booléens
- Les variables et l'affectation
- Les conditions : **si, sinon**
- Les boucles : **tant que, pour**
- Les fonctions : comment décomposer un problème
- Logique booléenne : **ET, OU, NON**

Pratique

- Coding game

17 **Semaine 1 : Mise à niveau — Fondations du web**

Objectifs : Comprendre les bases HTML/CSS/JS, structurer une page, écrire du code propre

Théorie

- Introduction au web : navigateur, serveur, rendu, DOM
- Sémantique HTML (formulaires, balises, accessibilité)
- CSS : sélecteurs, box model, positionnement, flexbox
- JS : variables, fonctions, conditions, boucles, événements

Pratique

- Reproduire une maquette simple (carte profil ou landing)
- Intégrer une section responsive en Flexbox
- JS : créer une TODO liste basique (add/remove, DOM)

17 **Semaine 2 : Structuration — Bonnes pratiques**

Objectifs : Gagner en rigueur, structurer du code propre et réutilisable

Théorie

- Organisation de projet (dossiers, fichiers)
- Bonnes pratiques CSS (naming, DRY, utilitaires)

- Introduction à Git + GitHub (commit, push, branch)
- JS : manipulation DOM, événements, debug

Pratique

- Cloner et améliorer un mini-projet open source (formulaire, galerie...)
- Créer un site 3 pages (responsive, navigable)
- Mise en ligne sur GitHub Pages



Semaine 3 : Front-end moderne & composants

Objectifs : Découvrir un framework (Vue), comprendre l'état, les props

Théorie

- Introduction à Vue : composant, props, état, JSX
- Organisation logique (lift state up, logique de composants)
- Tailwind CSS ou un système de design

Pratique

- Cloner une interface simple (ex : tableau de tâches, profil utilisateur)
- Créer des composants réutilisables (bouton, carte, modal...)



Semaine 4 : Mobile & autonomie

Objectifs : Toucher à React Native, autonomie sur des petites tâches du projet

Théorie

- Différences web vs mobile (layout, navigation)
- Expo + React Native : base d'une app mobile
- Lecture de code et pair programming

Pratique

- Intégrer un écran mobile dans une app (écran login ou liste)
- Reproduire un écran mobile d'un design figma simple
- Participer à une tâche réelle du backlog (avec accompagnement)



Suivi & accompagnement

- **Daily check-ins** (15 min) : objectifs, blocages
- **1 pair programming par semaine**
- **1 démo en fin de semaine** : montrer ce qu'il a appris
- **Feedback en continu**

Plan d'accompagnement Backend



Semaine 0 – Logique & Bases PHP



Objectifs :

- Comprendre la logique de programmation orientée serveur
- Apprendre à manipuler les bases du langage PHP
- Préparer le terrain pour Laravel



Théorie :

- Rôle du back-end vs front-end
- Notion de serveur HTTP, requêtes et réponses
- Syntaxe PHP : variables, types, opérateurs, conditions, boucles
- Fonctions & portée des variables
- Structures de données : tableaux indexés et associatifs



Pratique :

- Script PHP simple sans base de données : calculs, gestion de formulaires (GET, POST)
Exo : Page de contact en PHP (affichage, validation simple, enregistrement en fichier)



Semaine 1 – PHP Orienté Objet + Initiation aux bases de données



Objectifs :

- Appréhender la POO en PHP
- Introduction à MySQL et interactions de base



Théorie :

- Classes, objets, attributs, méthodes
- Constructeurs, visibilité (public, private)
- Notions de base de données relationnelle
- Introduction à PDO (connexion, requêtes préparées)



Pratique :

- Création d'une classe `Contact` avec enregistrement en BDD
- Exo : mini gestionnaire de contacts (CRUD de base)



Semaine 2 – Introduction à Laravel



Objectifs :

- Comprendre la structure d'un projet Laravel
- Créer ses premières routes et vues
- Utiliser Eloquent pour interagir avec la base

Théorie :

- Présentation de Laravel (MVC, Artisan, Blade, Eloquent)
- Structure d'un projet
- Routing, contrôleurs, vues Blade
- Migration, Seeder, Eloquent Model

Pratique :

- Créer une nouvelle app Laravel
- CRUD basique d'un modèle `Task`
- Utilisation de `php artisan make:*`
- Mise en place de la DB avec migration/seeder

Semaine 3 – Approfondissement Laravel

Objectifs :

- Savoir organiser son code
- Gérer les relations en base
- Utiliser les formulaires et la validation

Théorie :

- Relations `hasOne`, `hasMany`, `belongsTo`
- Middleware, validation, messages d'erreurs
- Authentification Laravel Breeze (si souhaité)

Pratique :

- Ajout d'une relation `Task belongs to User`
- Formulaire avec validation (`Request` personnalisé)
- Exo : mini système de to-do list multi-utilisateur

Semaine 4 – Projet de consolidation

Objectifs :

- Mettre en œuvre toutes les notions apprises dans un projet complet

Projet final :

- Définir un besoin simple : blog, gestion de tâches, suivi de commandes...
Réaliser les maquettes (vite fait), base de données, CRUD complet

- Versionner le code avec Git
- Documentation rapide de l'app (routes, fonctionnalités, structure)