### 🚀 Évaluation finale - Développeur Web & Web Mobile

📅 Durée du projet : 2 semaines

## 🎯 Projet : Portfolio Dynamique avec Dashboard

Vous allez concevoir une application MERN permettant aux utilisateurs de gérer et afficher leurs compétences via un portfolio dynamique. L'application devra être sécurisée, responsive (mobile-first) et conforme aux exigences RGPD avec un système de gestion des cookies (Tarteaucitron.js) et un Captcha (Google reCAPTCHA ou Tarteaucitron) pour sécuriser l’authentification.

## 🏗 Compétences validées et Astuces

### 1️⃣ Maquetter des interfaces utilisateur

📌 Outils recommandés : Figma, Adobe XD  
📌 Bonnes pratiques UX/UI :  
✅ Design mobile-first : commencez la conception sur mobile, puis adaptez au desktop  
✅ Navigation intuitive entre le portfolio et le dashboard  
✅ Affichage des compétences sous forme de cartes interactives

💡 Astuce : Utilisez un système de grille CSS (Flexbox, Grid) pour assurer une mise en page fluide.

### 2️⃣ Mettre en place une base de données relationnelle

📌 Technologies : MongoDB + Mongoose  
📌 Collections MongoDB :

* users : nom, email, mot de passe (haché avec bcrypt), rôle (admin/user)
* skills : titre, catégorie, niveau (débutant, intermédiaire, expert), image (Cloudinary)
* settings : préférences utilisateur, choix des cookies

💡 Astuce : Pensez à indexer les champs utilisés fréquemment pour optimiser les requêtes.

### 3️⃣ Développer des composants d’accès aux données (SQL ou NoSQL)

📌 Backend : Express + MongoDB  
✅ API REST avec routes /api/skills (CRUD)  
✅ Upload des images sur Cloudinary via Multer  
✅ Authentification sécurisée avec JWT & bcrypt

💡 Astuce : Stockez le token JWT dans un httpOnly cookie pour éviter les attaques XSS.

### 4️⃣ Développer des composants métier côté serveur

📌 Sécurité :  
✅ helmet pour protéger les en-têtes HTTP  
✅ cors pour gérer les accès cross-origin  
✅ Tarteaucitron.js pour la gestion des cookies (RGPD)  
✅ Google reCAPTCHA ou Tarteaucitron CAPTCHA pour bloquer les bots  
✅ Middleware de rôles (admin, utilisateur) pour sécuriser le dashboard

💡 Astuce : Ajoutez un logger (Winston, Morgan) pour suivre les actions des utilisateurs.

### 5️⃣ Documenter le déploiement de l’application

📌 README.md avec :  
✅ Installation et exécution du projet  
✅ Déploiement backend (Render, Railway, Heroku)  
✅ Déploiement frontend (Vercel, Netlify)  
✅ Collection Postman pour tester l’API

💡 Astuce : Faites un schéma d’architecture pour clarifier l'organisation du projet.

### 6️⃣ Réaliser des interfaces utilisateur statiques et dynamiques

📌 Frontend avec React (Mobile-First & Responsive) :  
✅ Portfolio dynamique affichant les compétences  
✅ Dashboard sécurisé pour ajouter/modifier/supprimer des compétences  
✅ Formulaire de connexion avec reCAPTCHA  
✅ Gestion des cookies avec Tarteaucitron.js  
✅ Utilisation de React Router et Axios  
✅ Mise en page responsive (Mobile-First) avec Tailwind CSS ou Bootstrap

💡 Astuce :

* Testez votre site sur différents écrans (Chrome DevTools) pour un rendu optimal.
* Utilisez React Query ou Redux Toolkit pour optimiser la gestion des requêtes API.

## 📂 Livrables attendus

✅ Code source GitHub avec un README clair  
✅ Rapport de projet expliquant les choix techniques  
✅ Diaporama de présentation  
✅ Collection Postman pour les tests API

⏳ Durée : 2 semaines

🔥 Montrez votre expertise ! 🔥