

Cahier des Charges - Portfolio / Projet de Fin d'Année

1. Introduction

1.1 Présentation du Projet

Ce projet de portfolio constitue une vitrine numérique présentant les compétences et réalisations acquises tout au long de la formation. Il sera évalué par un jury dans le cadre de la validation du diplôme de la Web@cadémie.

Le portfolio doit être :

- **Hybride** : il servira à la fois d'outil d'apprentissage et de démonstration professionnelle.
- **Évolutif** : il devra être mis à jour régulièrement pour refléter les nouvelles réalisations.
- **Accessible en ligne** : sous forme d'une application web "one page".
- **Personnalisé** : il doit refléter l'identité et les compétences du candidat

2. Objectifs

Objectifs Généraux

- Mettre en avant les compétences acquises en développement web.
- Prouver la capacité à structurer, développer et déployer une application web.
- Présenter une solution ergonomique, performante et responsive.
- Assurer la qualité et la maintenabilité du code.

Objectifs Techniques

- Développer une application React avec TailwindCSS.
- Suivre une approche modulaire et réutilisable dans le développement des composants.
- Implémenter un système de gestion des données dynamique.
- Assurer une compatibilité multi-plateforme et une bonne performance.
- Mettre en place des tests et un suivi de la qualité.

3. Cadrage du Projet et Conceptualisation

Choix Techniques

Framework Front-end : React

- Composants réutilisables : React permet une structure modulaire facilitant la maintenance et l'évolution du projet.
- Gestion de l'état efficace : via React Context API ou Redux si nécessaire.
- Optimisation des performances : grâce au Virtual DOM et au rendu conditionnel.

CSS : TailwindCSS

- Rapidité de développement : système de classes utilitaires permettant une intégration rapide.
- Design responsive intégré : flexibilité pour s'adapter aux différentes résolutions.
- Personnalisation avancée : possibilité de surcharger les styles avec un fichier de configuration.

Architecture du Projet

Le projet suivra une structure modulaire :

- Pages : organisation en sections indépendantes (Accueil, Projets, Contact).
- Composants : réutilisables et optimisés.
- Données : chargement dynamique à partir d'un fichier JSON ou d'une API.

Maquettage et UI/UX

- Utilisation de Figma pour concevoir les maquettes et assurer une cohérence visuelle.
- Respect des principes de Flat Design et d'accessibilité.
- Expérience utilisateur optimisée pour mobile-first.

Déploiement de l'Environnement de Travail

- Versionnage du code : utilisation de GitHub pour suivre l'évolution du projet.
- Outils de communication : Github Project pour la gestion des tâches.
- Collaboration : pair-programming et revue de code via GitHub PR.

4. Développement de la Solution Web

Implémentation des Fonctionnalités

- Présentation interactive des projets avec animations et transitions.
- Système de filtrage et tri pour naviguer parmi les réalisations.
- Formulaire de contact sécurisé avec validation des données.
- Optimisation du SEO pour améliorer la visibilité.

Gestion des Données

- Stockage des projets sous forme de JSON statique ou via une API headless (ex: Firebase).
- Utilisation de localStorage pour sauvegarder les préférences de l'utilisateur.

Déploiement

- Hébergement sur Vercel ou Netlify pour assurer un accès rapide et fiable.
- Configuration d'un nom de domaine personnalisé.

5. Assurance Qualité et Maintenance

Tests et Optimisation

- Tests unitaires avec Jest et tests d'intégration avec React Testing Library.
- Vérification de la performance avec Google Lighthouse.
- Analyse de la compatibilité navigateur et correction des bugs.

Documentation

- Rédaction d'un guide technique expliquant la structure et les choix du projet.
- Création d'une documentation utilisateur pour faciliter la prise en main.

Améliorations et Suivi

- Mise en place d'un système de feedback utilisateur.
- Analyse des statistiques d'usage via Google Analytics.
- Mise à jour continue en fonction des évolutions technologiques et des besoins.

6. Conclusion

Ce projet de portfolio représente une vitrine numérique démontrant les compétences en développement web, UI/UX et gestion de projet. Grâce à React et TailwindCSS, il sera rapide, performant et évolutif, tout en offrant une expérience utilisateur fluide et engageante.