Raphael TOUZET & Sébastien PEREIRA

***Projet design UX.I***

ZenCamp

***1. Introduction***

***1.1 Contexte***

ZenCamp site de réservation séjours bien-être (retraites de yoga, méditation, digital detox). La plateforme propose des séjours sous formats divers : individuels, groupes, ou événements spéciaux, en assurant une expérience utilisateur sécurisée, intuitive, et personnalisée.

***1.2 Objectifs***

Diversité d'offres : Proposer des retraites variées par thèmes (yoga, méditation), durées, et localisations.

Accessibilité simplifiée : Permettre une réservation simple et rapide via une interface claire.

Paiement sécurisé : Garantir un processus d'achat sécurisé et transparent.

***2. Fonctionnalités principales***

***2.1 Page d'accueil***

Présentation des retraites populaires, promotions et nouveautés.

Accès rapide aux catégories (yoga, méditation, digital detox).

Recommandations personnalisées basées sur les préférences utilisateur.

***2.2 Catalogue des retraites***

Filtres disponibles :

Localisation : Montagne, mer, campagne.

Thématique : Yoga, méditation, jeûne, digital detox.

Durée : Week-end, semaine, séjours longs.

Budget : Prix, promotions.

Tri possible par popularité, date, prix et avis utilisateurs.

***2.3 Fiche séjour***

Chaque retraite dispose :

Description détaillée avec planning, photos, vidéos.

Prix avec mention claire des éventuelles réductions.

Avis des utilisateurs avec notation de 1 à 5 étoiles.

Bouton clair et visible pour « Réserver ».

***2.4 Panier de réservation***

Gestion intuitive permettant d’ajouter, modifier ou supprimer des séjours.

Calcul automatique du prix total avec détails (prix séjour, taxes éventuelles).

Vérification automatique de disponibilité.

Passage rapide à l'étape de paiement.

***2.5 Processus de réservation***

Récapitulatif détaillé : Séjours sélectionnés, dates, coûts.

Formulaire utilisateur : Coordonnées, préférences particulières (repas spécifiques, accessibilité).

Paiement sécurisé : Options telles que carte bancaire, PayPal.

Confirmation immédiate : Email avec détails de la réservation et contact organisateur.

***2.6 Compte utilisateur***

Création simple via email et mot de passe sécurisé.

Récupération de mot de passe sécurisée.

Gestion des informations personnelles et préférences.

Historique complet des réservations.

Liste de souhaits pour planifier des séjours ultérieurs.

***2.7 Barre de recherche***

Présente sur toutes les pages, rapide et efficace.

Recherche avancée par mot-clé, thématique, localisation, dates et prix.

***2.8 Avis utilisateurs***

Seuls les utilisateurs ayant effectué un séjour peuvent laisser un avis.

Modération avant publication pour assurer qualité et pertinence.

***3. Contraintes techniques***

***3.1 Technologies***

Frontend : React.js ou Vue.js pour garantir fluidité et rapidité.

Backend : Node.js ou Symfony pour robustesse et flexibilité.

Base de données : MySQL ou PostgreSQL pour fiabilité et performance.

Sécurité : HTTPS obligatoire, chiffrement sécurisé des données utilisateurs.

***3.2 Hébergement***

Infrastructure cloud (AWS, Google Cloud) garantissant évolutivité et performances optimisées.

***3.3 Performance***

Temps de chargement inférieur à 2 secondes.

Support de 1000 utilisateurs simultanés sans dégradation de performance.

***4. Sécurité et conformité***

Paiements sécurisés via protocole 3D Secure.

Conformité RGPD stricte dans la gestion des données personnelles.

***5. Livrables***

Maquette

Une image contenant plein air, nature, arbre, paysage

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran, Police

Le contenu généré par l’IA peut être incorrect.

Code sources

Une image contenant texte, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Une image contenant capture d’écran, texte, Logiciel multimédia, logiciel

Le contenu généré par l’IA peut être incorrect.

Une image contenant texte, capture d’écran

Le contenu généré par l’IA peut être incorrect.

Tests Unitaires

Les tests unitaires ciblent les différentes unités du code (fonctions, méthodes, composants individuels).

Technologies utilisées :

Jest (Frontend - React/Vue)

PHPUnit ou Jest (Backend - Symfony/Node.js)

Cas de tests :

Validation des formulaires d’inscription/connexion.

Validation des entrées utilisateurs (filtres, recherches).

Calcul précis des prix (prix séjour, promotions, taxes).

Disponibilité correcte des séjours lors d’ajout au panier.

Gestion sécurisée des identifiants et mots de passe.

Exemple de test :

Vérifier que le panier ne permet pas l’ajout d’un séjour non disponible.