**Cahier des Charges : Gestion de Camping**

**1. Contexte et Objectifs**

Le projet consiste à créer un **site web** pour la gestion d'un camping, qui comprend plusieurs types d'hébergements :

* **Emplacements nus**
* **Tentes meublées**
* **Mobil-homes de différentes tailles**

Le site devra permettre aux utilisateurs de réserver ces hébergements en ligne et de gérer les aspects administratifs liés aux réservations (ex. : tarifs saisonniers, gestion des disponibilités, etc.). L'application en **JavaScript Objet** sera utilisée pour certaines fonctionnalités côté client.

**2. Description des Fonctionnalités**

**2.1. Gestion des Hébergements**

* **Types d'hébergements** :
  + Emplacements nus
  + Tentes meublées (avec nombre de personnes, équipements inclus)
  + Mobil-homes de différentes tailles (définir des modèles standards de taille, capacité et équipements)
* **Gestion des capacités** :
  + Chaque type d'hébergement doit pouvoir être associé à une capacité spécifique (ex. : nombre de personnes, superficie).
* **Affichage sur le site** :
  + Le client doit pouvoir voir les types d'hébergement disponibles à la réservation.

**2.2. Gestion des Tarifs**

* **Tarification dynamique** :
  + Les prix doivent dépendre de la **saison** (basse, haute, etc.). Une même catégorie d'hébergement peut avoir différents prix selon la période.
  + Une fermeture hivernale est prévue (du 01/10 au 31/03)
  + Le système doit permettre de configurer les périodes et les tarifs associés pour chaque type d'hébergement.
* **Gestion des promotions** **: (Bonus)**
  + Possibilité d’ajouter des remises pour certaines périodes ou offres spéciales.

**2.3. Gestion des Réservations**

* **Réservation en ligne** :
  + Les utilisateurs doivent pouvoir sélectionner un type d'hébergement, une date d’arrivée et une date de départ.
  + Le système doit afficher la disponibilité en fonction des dates choisies.
  + Les réservations doivent avoir le nombre de personnes en détails (ex : 2 adultes + 2 enfants)
* **Vérification de disponibilité** :
  + Le système doit **empêcher les chevauchements de réservations** (ex. : éviter qu'un même bien soit réservé pour deux clients à la même période).
  + **Contrôle des dates de réservation** : Les dates de réservation doivent être vérifiées pour éviter que des réservations ne soient faites dans le passé ou en dehors des périodes d'ouverture du camping.
* **Gestion des annulations** **: (bonus)**
  + L'utilisateur peut annuler une réservation dans un certain délai (ex. : 48 heures avant l’arrivée).

**2.4. Gestion des Utilisateurs**

* **Inscription et authentification** :
  + Un système d'authentification des utilisateurs avec des rôles (ex. : administrateur, utilisateur classique) doit être mis en place.
  + Les utilisateurs peuvent créer un compte, se connecter, et consulter l'historique de leurs réservations.
* **Gestion des profils** :
  + Les utilisateurs peuvent modifier leurs informations personnelles, consulter leurs réservations passées et à venir, etc.

**2.5. Interface d'Administration**

**2.5.1. Gestion des Hébergements (CRUD)**

* **Création, Modification, Suppression des biens** :
  + L’administrateur doit pouvoir **ajouter, modifier ou supprimer** les types d'hébergements (emplacements nus, tentes meublées, mobil-homes).
  + Lors de la création ou modification d'un bien, l'administrateur devra spécifier :
    - Type d'hébergement
    - Capacité (nombre de personnes)
    - Description (équipements, superficie, etc.)
    - Tarification
    - Disponibilité
* **Affichage des biens** :
  + L'admin doit pouvoir consulter la liste des biens existants avec un tableau de bord qui inclut les informations essentielles : type, capacité, statut de réservation, prix, etc.

**2.5.2. Gestion des Réservations (Planning)**

* **Vue du planning des réservations** :
  + L'administrateur doit pouvoir consulter un **planning global** des réservations, avec la possibilité de filtrer par type de bien, dates, et état des réservations (confirmée, en attente, annulée).
  + Ce planning doit être visuel, avec les dates de début et de fin des réservations clairement affichées pour chaque bien.
* **Actions sur les réservations** :
  + L'administrateur pourra **modifier ou annuler** des réservations si nécessaire.
  + Il pourra également **ajouter manuellement des réservations** en cas de besoin (par exemple, pour des clients appelant directement).

**2.6. Application JavaScript Objet**

**2.6.1. Gestion des Entrées et Sorties**

* L'application JavaScript doit permettre de **visualiser les entrées et sorties journalières** des clients pour chaque bien (emplacement nu, tente meublée, mobil-home).
* Chaque **entrée** et **sortie** doit être enregistrée avec les informations suivantes :
  + **Date d'arrivée** et **date de départ**
  + **Nom du client**
  + **Type d'hébergement**
  + **Statut de nettoyage** :
    - Une option **check** permettra à l'administrateur ou au personnel de signaler qu'un bien a été **nettoyé** après le départ d'un client.
    - Lorsqu'un bien est nettoyé, l'option de **check** doit être activée et le bien doit être marqué comme prêt pour une nouvelle réservation.

**2.6.2. Interaction avec le Backend**

* L'application JavaScript devra récupérer les **données des réservations** depuis le backend, notamment les entrées et sorties journalières.
* La fonctionnalité de **check de nettoyage** doit être envoyée au backend pour mettre à jour le statut du bien.

**2.6.3. Interface utilisateur**

* L'interface JavaScript devra afficher un calendrier ou un tableau des **entrées et sorties** pour chaque type d'hébergement avec des couleurs ou des icônes indiquant si un bien est disponible ou déjà réservé.
* Le personnel pourra marquer un bien comme **nettoyé** via un bouton de **check** pour chaque entrée/sortie.

**3. Technologies**

**3.1. Site général - Symfony 7**

* **Framework Symfony 7** pour le développement backend.
* **Doctrine ORM** pour la gestion des bases de données (hébergements, réservations, utilisateurs, etc.).
* **Twig** pour la gestion des templates et de l'affichage côté serveur.
* **Sécurité** : Mise en place d'un système d'authentification avec **Symfony Security** (rôles utilisateurs, sécurité des données).

**3.2. Appli Frontend**

* **HTML5, CSS3 et JavaScript** pour le frontend.
* **JavaScript Objet** pour gérer la logique dynamique (réservation, affichage des disponibilités sur le calendrier, gestion des entrées/sorties, nettoyage).

**3.3. Base de données**

* **Base de données relationnelle** (MySQL ou MariaDB) pour stocker les informations sur les utilisateurs, les hébergements, les réservations, et les tarifs.

**4. Contraintes Techniques**

* **Réactivité** : Le site doit être **responsive**, adapté aux mobiles et tablettes.
* **Performance** : Le système doit être performant, surtout lors de la consultation des disponibilités en temps réel.
* **Sécurité** : Protection des données des utilisateurs et des informations sensibles (réservations, paiements).

**5. Planning de Développement**

1. **Phase 1 : Analyse et conception**
   * Définition des exigences fonctionnelles.
   * Conception de la base de données et des modèles d’entités.
   * Rédaction des spécifications techniques.
2. **Phase 2 : Développement Site**
   * Création des entités et des relations de la base de données.
   * Développement des fonctionnalités principales de gestion des hébergements et des réservations.
3. **Phase 3 : Développement Appli**
   * Création de l'interface utilisateur, en particulier le calendrier interactif et la gestion des entrées/sorties et nettoyage.
   * Intégration de la logique JavaScript pour les interactions côté client.