

Guide de Conception : Système de Gestion « La Cerisaie »

Ce guide explique comment nous avons transformé les besoins d' un camping en une solution informatique robuste en langage C.

1. Comprendre le Problème (L' Analyse)

Le camping gère des **emplacements** (tentes, bungalows, etc.) et des **activités sportives**. Le défi est de passer d' une gestion papier à une gestion numérique capable de :

- Retrouver instantanément les tarifs.
 - Savoir quel emplacement est libre ou occupé.
 - Calculer une facture précise sans erreur humaine.
-

2. La Structure des Données (La Fondations)

Pour que l' ordinateur comprenne le camping, nous avons créé des « moules » appelés **Structures**.

- **Le Type d' Emplacement** : C' est le catalogue des prix. Il définit qu' un « Bungalow » coûte 17,50€ par personne et par jour.
 - **L' Emplacement** : C' est le terrain physique. Il a un numéro, une surface et sait s' il est libre ou occupé.
-

3. Les Choix Techniques (La Stratégie)

A. La Modularisation

Plutôt que d' écrire un seul fichier géant et illisible, nous avons séparé le travail :

1. **Le fichier .h** : Il liste les différentes fonctions.
2. **Le fichier .c** : Il contient les détails des différentes fonctions.
3. **Le main.c** : Il fait appelle aux fonctions.

B. Mémoire RAM vs Disque Dur

- **Le Tableau (RAM)** : Les tarifs sont chargés dans un tableau au démarrage. C' est ultra-rapide pour lire un prix pendant que le programme tourne.
- **Le fichier binaire (disque)** : Avant de quitter, le programme enregistre tout dans types.dat. Ainsi, rien n' est perdu quand on éteint l' ordinateur.

4. Pourquoi ce Code est « Propre » ?

Nous avons appliqué des règles de programmation modernes :

- **Pas de sauts brusques** : Nous n' utilisons pas de break ou continue au milieu des boucles. Pourquoi ? Pour que le code soit comme une histoire : on commence au début, on finit à la fin, sans téléportation imprévue.
- **Commentaires de fin de bloc** : Chaque fois qu' une boucle ou une condition se ferme, un commentaire indique ce qui se termine (ex :). C' est une bouée de sauvetage pour ne pas se perdre dans les accolades `// fin while{ }`

5. Comment fonctionne le flux de données ?

1. **Démarrage** : Le programme cherche le fichier . S' il existe, il remplit le tableau .types.dattabTypes
2. **Utilisation** : L' utilisateur consulte les tarifs. Le programme lit directement dans la mémoire vive (très rapide).
3. **Fermeture** : L' utilisateur choisit « Quitter ». Le programme prend le tableau et l' écrase proprement sur le disque dur pour la prochaine fois.