

# **Projet ChenYl-Tech – Gestion d’un chenil en langage C**

## **Présentation générale**

Le projet **ChenYl-Tech** consiste à développer une application en langage C permettant de gérer un refuge pour animaux. Réalisé dans le cadre du module informatique de la filière pré-ING1 (année 2024–2025), ce projet propose une interface utilisateur simple et robuste, exclusivement textuelle, avec des fonctionnalités essentielles pour la gestion quotidienne d’un chenil.

## **Objectifs pédagogiques**

- Appliquer les connaissances en C à travers un projet concret.
- Structurer proprement un projet modulaire en plusieurs fichiers.
- Assurer la robustesse du code face aux erreurs utilisateur.
- Gérer des tableaux de structures.
- Implémenter des variantes algorithmiques en lien avec des règles métiers concrètes.

## **Fonctionnalités implémentées**

L'application permet de :

- Ajouter un animal** confié au refuge, avec saisie sécurisée (nom, espèce, poids, année de naissance, commentaire facultatif).
- Rechercher un ou plusieurs animaux** par nom, espèce ou tranche d’âge (jeune < 2 ans, sénior > 10 ans).
- Adopter un animal** (suppression de la base) via son identifiant unique.
- Afficher l'inventaire par quartiles d'âge** (INV\_AGE\_ASC) afin de visualiser la distribution d’âge des animaux.
- Estimer la charge de nettoyage hebdomadaire** (DAY\_CLEAN) selon l’espèce de chaque pensionnaire.

Toutes les entrées sont protégées pour éviter les plantages du programme, même en cas de saisies invalides (exemples : lettres à la place de chiffres, caractères spéciaux, entrées vides, etc.).

## Structure du projet

Le projet est découpé en plusieurs fichiers C, chacun correspondant à une tâche spécifique, avec un seul fichier d'en-tête (`includes.h`) centralisant toutes les définitions, bibliothèques et prototypes.

### Arborescence :

- `main.c` : Menu principal et boucle de l'application.
- `ajouter.c` : Fonction pour ajouter un nouvel animal.
- `rechercher.c` : Rechercher un animal selon différents critères.
- `adopter.c` : Supprimer un animal par son ID (adoption).
- `inv_age_asc.c` : Variante pour répartir les animaux par quartiles d'âge.
- `day_clean.c` : Variante pour calculer la charge hebdomadaire de nettoyage.
- `utils.c` : Fonctions sécurisées pour la lecture des entrées utilisateur.
- `get_current_year.c` : Fonction système pour obtenir l'année en cours.
- `includes.h` : Fichier unique d'en-tête.
- `Makefile` : Fichier pour automatiser la compilation.

## Robustesse et sécurité

Le code intègre des **mécanismes de sécurisation avancés** :

- Lecture protégée des entrées via `fgets` et `strtol/strtof` avec détection des erreurs.
- Évitement des débordements mémoire.
- Boucles de relance claire pour l'utilisateur en cas de mauvaise saisie.
- Aucun `scanf` direct sans protection.
- Gestion stricte des limites (pas plus de 50 animaux).
- Messages d'erreurs explicites.

# Compilation et exécution

## Prérequis :

Système Unix/Linux (ou terminal compatible comme WSL sur Windows).  
Compilateur C (GCC ou compatible).

## Commandes :

`make` : Compile automatiquement tous les fichiers.  
`./chenil` : Lance l'application.  
`make clean` : Supprime les fichiers objets et l'exécutable.

## Contraintes respectées

Application modulaire (chaque fonction = 1 fichier).  
Entrées utilisateur 100% sécurisées.  
Application stable (aucun crash possible).  
Utilisation exclusive de la mémoire (pas de sauvegarde fichier, par choix).  
Espèces gérées : **chien, chat, hamster, autruche** (possibilité d'en ajouter facilement).

## Encadrement

Projet encadré par :

**Eva Ansermin** – [eva.ansermin@cyu.fr](mailto:eva.ansermin@cyu.fr)  
**Romuald Grignon** – [romuald.grignon@cyu.fr](mailto:romuald.grignon@cyu.fr)

## Conclusion

Ce projet a permis d'appliquer de manière concrète les bonnes pratiques de développement logiciel en langage C : modularité, robustesse, gestion mémoire et clarté du code. L'approche a mis l'accent sur la stabilité et l'expérience utilisateur, avec un haut niveau de sécurisation contre les erreurs de saisie. Les variantes

proposées ont enrichi l'application avec des fonctionnalités pratiques liées à la gestion quotidienne du refuge.