El Hadji Osseynou Accrachi Le 12 avril 2022

Rebon Numérique

Parcours Développement

**C# Projet 1** : Logiciel pour vétérinaire

Description du projet :

Concevoir un logiciel pour vétérinaire permettant de gérer la liste d’animaux en pensions.

**PLAN** :

1. Lecture du dossier fonctionnel (objectif, exigence, recommandation, mots clés, résultats attendus, …).
2. Définir la structure pour contenir la liste des animaux.
3. Commencer par définir les différentes fonctions qui traitent les options du menu de mon logiciel
4. Ajouter un animal à la liste : l’animal doit avoir un ID (numéro d’identification), un type (chat, chien, oiseau, …), un nom (chaine de caractères), un âge (nombre entier), un poids (en entier), une couleur (rouge, bleu et violet uniquement) et un nom du propriétaire (chaîne de caractères)
5. Afficher la liste de tous les animaux en pension : parcourir la liste et afficher tous les éléments sous forme de tableau
6. Afficher la liste de tous les propriétaires : parcourir la liste et afficher toute la colonne des propriétaires sous forme de tableau
7. Calculer le nombre total des animaux en pension et l’afficher (le nombre total d’animaux est obtenu en incrémentant le nombre d’animaux à chaque ajout d’un animal)
8. Calculer le poids total des animaux en pension et l’afficher (le poids total des animaux est obtenu en incrémentant le poids total des animaux à chaque ajout d’un animal)
9. Afficher la liste des animaux selon la couleur (rouge, bleu ou violet) : parcourir la liste des animaux et afficher uniquement les animaux d’une couleur donnée
10. Retirer un animal de la liste à partit de son numéro d’identification (ID) : aller dans la liste à l’ID de l’animale à supprimer et décaler tous les animaux suivants à partir de la position de l’animal à retirer.
11. Définir une procédure pour quitter le logiciel
12. Tester individuellement les différents traitements (fonctions) de mon logiciel
13. Concevoir un menu qui affiche les différents traitements de mon logiciel
14. Mettre en place le contrôleur de mon application
15. Faire un test de l’ensemble du programme
16. Faire des tests unitaires pour corriger les bugs, affichages
17. Évaluer les différentes exigences du dossier fonctionnel avec le programme
18. Présenter le programme au formateur
19. Corrections et révision du programme

Types des variables utilisées :

1. Structure pour les éléments de la liste

* Liste d’animaux : tableau multidimensionnel (string **liste [10, 7],** avec 10 le nombre d’animaux en ligne et 7 le nombre de caractéristique de l’animal en colonne)
* ID animal : int (nombre entier)
* Nom animal : string (chaîne de caractère)
* Âge animal : int (nombre entier)
* Poids animal : int (nombre entier)
* Couleur animal (rouge, bleu et violet) : string (chaîne de caractère)
* Nom propriétaire : string (chaîne de caractère)

1. Types des variables additionnelles

* Nombre total d’animaux : int
* Poids total des animaux : int

Exigence à respecter

1. Choix invalide, afficher le message : « Le choix n'est pas valide... »
2. Le nombre maximum d’animaux est de 10
3. L’ajout d’un animal ne doit pas écraser les animaux déjà dans la liste
4. Un ID reliant chaque animal à un identifiant
5. Le système doit saisir ces informations et les conserver en mémoire tant et aussi longtemps que le logiciel est en exécution. Lorsque le logiciel se termine alors les données ne sont pas conservées.
6. Forcer l’utilisateur à choisir une couleur parmi le rouge, bleu ou violet