

Systeme de Gestion d'une Clinique Vétérinaire

Un outil CLI pour moderniser le suivi des patients

- **Équipe PySynergy :**
 - ADDI Safaa
 - CHRAA Zakaria
 - JAOUHARI Mounir
 - LAMOURI Mohamed Amine
 - **Encadrant :** Oussama
 - **Année :** 2025/2026
-

Introduction et Contexte

Contexte et Problématique

- **Situation initiale :** De nombreuses petites cliniques vétérinaires s'appuient sur une gestion papier des dossiers des patients, ce qui entraîne des inefficacités et des risques. 1
 - **Problèmes identifiés :**
 - Risque élevé de perte d'informations et d'erreurs humaines. 2
 - Recherche d'historiques médicaux lente et fastidieuse. 3
 - Difficulté à générer des rapports et des statistiques (revenus, motifs de consultation fréquents). 4
 - **Notre Solution :** Une application en ligne de commande (CLI) développée en Python pour automatiser et sécuriser la gestion de la clinique. 5
-

Objectifs du Projet

Objectifs Fonctionnels et Techniques

- **Fonctionnalités Clés :**
 - Gestion complète des propriétaires et de leurs animaux. 6
 - Enregistrement et consultation de l'historique médical. 7
 - Recherche par mots-clés dans les diagnostics. 8
 - Sauvegarde et chargement des données au format JSON. 9
 - Génération de rapports d'activité avec graphiques (Pandas & Matplotlib). 10
 - **Objectifs Techniques :**
 - Mettre en œuvre les principes de la **Programmation Orientée Objet** . 11
 - Concevoir une **architecture modulaire** et testable. 12
 - Assurer la **persistance des données** de manière fiable. 13
-

Architecture de l'Application

Architecture Modulaire

Notre application est conçue en plusieurs modules distincts pour garantir la séparation des préoccupations, la maintenabilité et l'évolutivité. 14

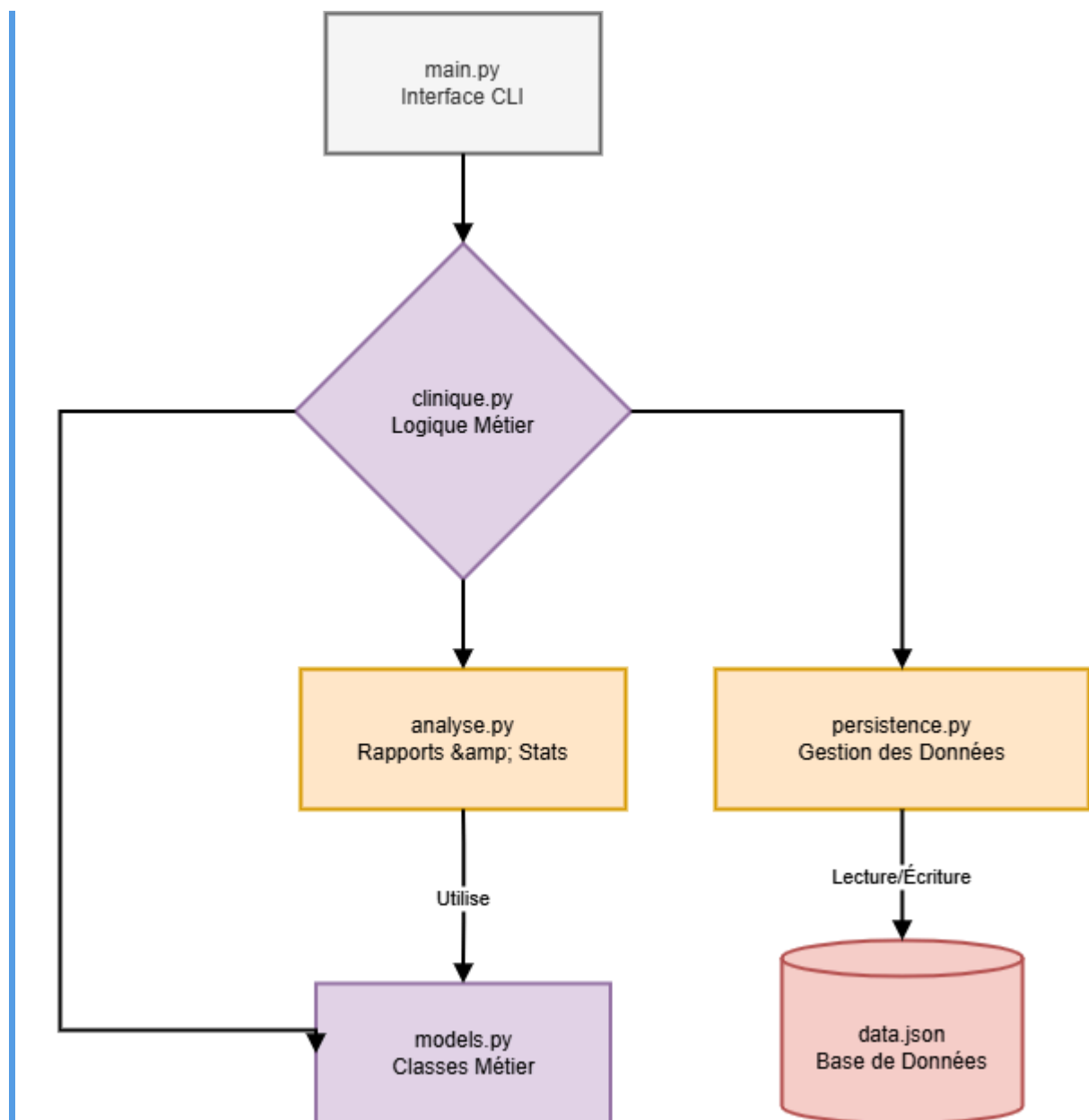
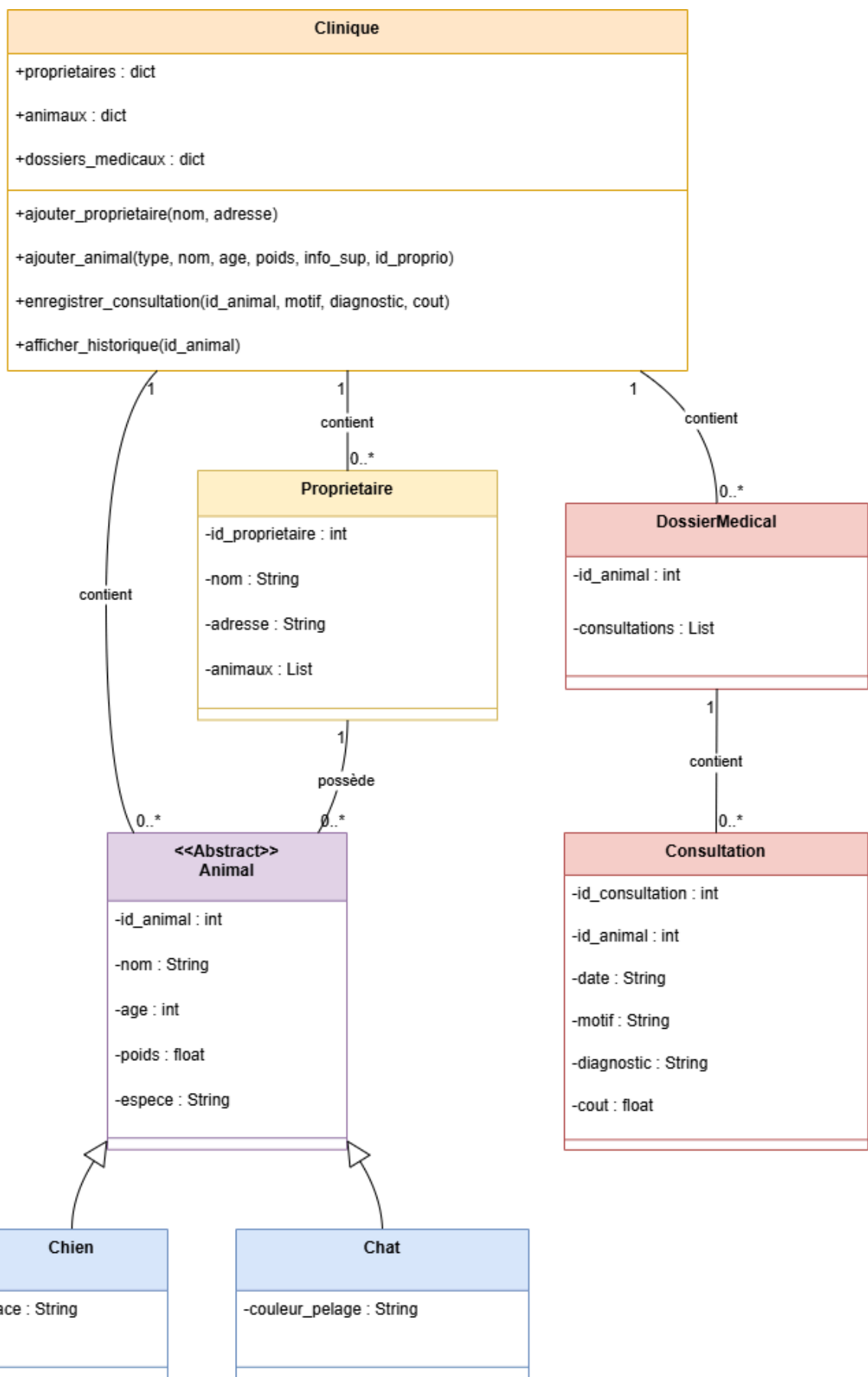


Diagramme de Classes

Modèle de Données (UML)

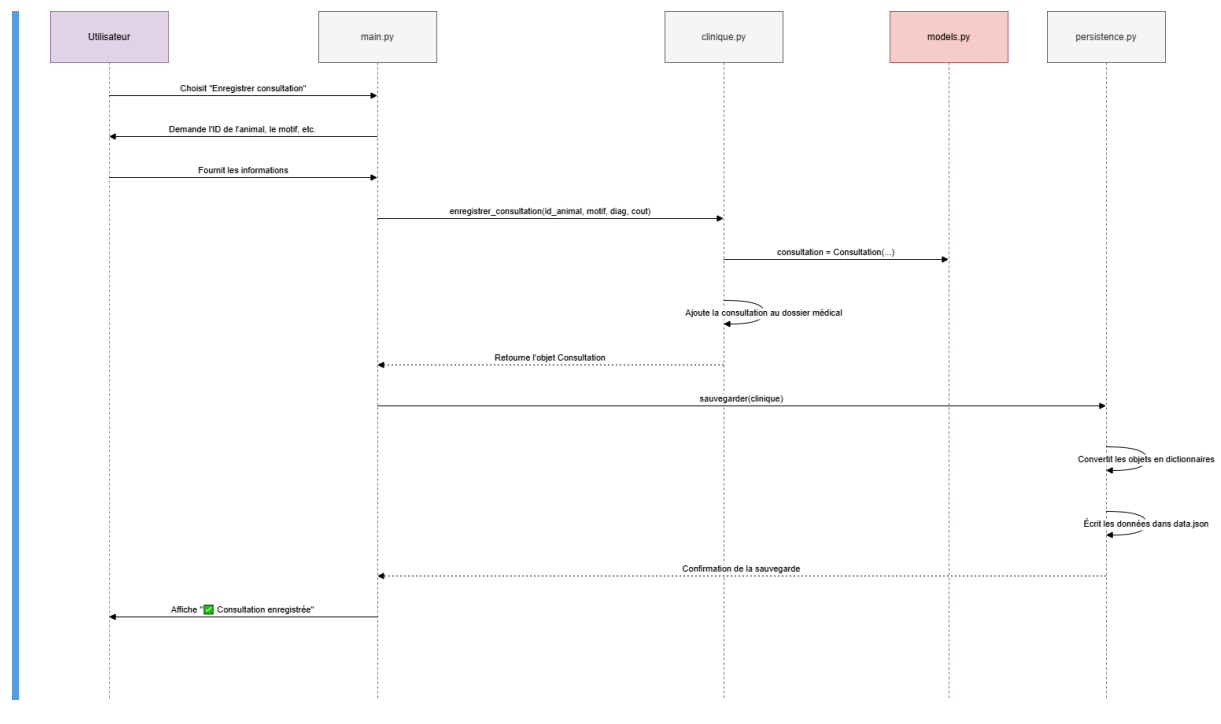
Le cœur de notre système est basé sur un ensemble de classes qui modélisent les entités de la clinique.



Diapositive 6 : Flux d'Interaction Utilisateur

Cas d'Usage : Enregistrer une Consultation

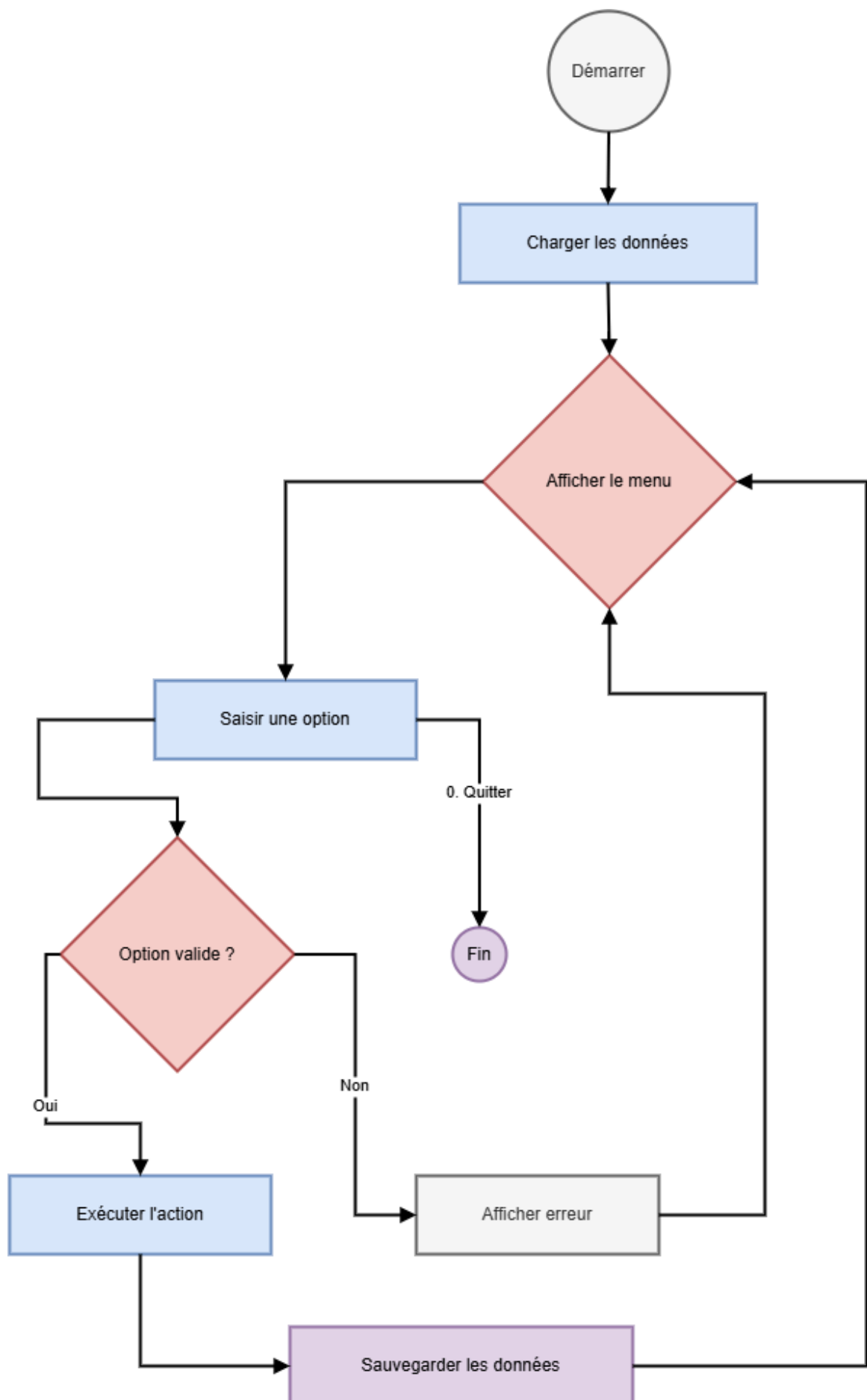
Ce diagramme de séquence illustre comment les différents modules interagissent lorsqu'un utilisateur enregistre une nouvelle consultation.



Diapositive 7 : Flux du Menu Principal

Logique de l'Interface (CLI)

Le point d'entrée `main.py` orchestre l'interaction avec l'utilisateur via un menu principal.



Diapositive 8 : Exécution du Programme

Captures d'Écran de l'Application

Menu Principal

```
===== CLINIQUE VÉTÉRINAIRE =====
1. Ajouter un propriétaire
2. Ajouter un animal
3. Lister tous les animaux
4. Lister animaux d'un propriétaire
5. Enregistrer une consultation
6. Historique d'un animal
7. Rechercher diagnostic
8. Générer rapport activité
0. Quitter
Choisir une option: 1
Nom du propriétaire (0 pour annuler): Amine
Adresse email (0 pour annuler): amine@gmail.com
✅ Propriétaire ajouté: ID: 6 Amine, amine@gmail.com, Animaux: []
```

Ajout d'un propriétaire et de son animal

```
0. Quitter
Choisir une option: 2
Types disponibles: Chien, Chat
Type animal (0 pour annuler): chien
Nom (0 pour annuler): Appa
Âge (ans) (0 pour annuler): 3
Poids (kg) (0 pour annuler): 9
Race (0 pour annuler): Berger
Propriétaires disponibles:
ID: 1 Youssef, youssef@gmail.com, Animaux: [1, 2]
ID: 2 Khadija, khadija@gmail.com, Animaux: [3, 4]
ID: 3 Omar, omar@yahoo.com, Animaux: [5, 6]
ID: 4 Amina, amina@hotmail.com, Animaux: [7]
ID: 5 Rachid, rachid@gmail.com, Animaux: [8, 9, 10]
ID: 6 Amine, amine@gmail.com, Animaux: []
ID du propriétaire (0 pour annuler): 6
✅ Animal ajouté: ID: 11 Appa, Chien, 3 ans, 9.0 kg, Race: Berger
```

Enregistrement d'une consultation

```
===== CLINIQUE VÉTÉRINAIRE =====
1. Ajouter un propriétaire
2. Ajouter un animal
3. Lister tous les animaux
4. Lister animaux d'un propriétaire
5. Enregistrer une consultation
6. Historique d'un animal
7. Rechercher diagnostic
8. Générer rapport activité
0. Quitter
Choisir une option: 1
Nom du propriétaire (0 pour annuler):
```

Analyse et Rapports

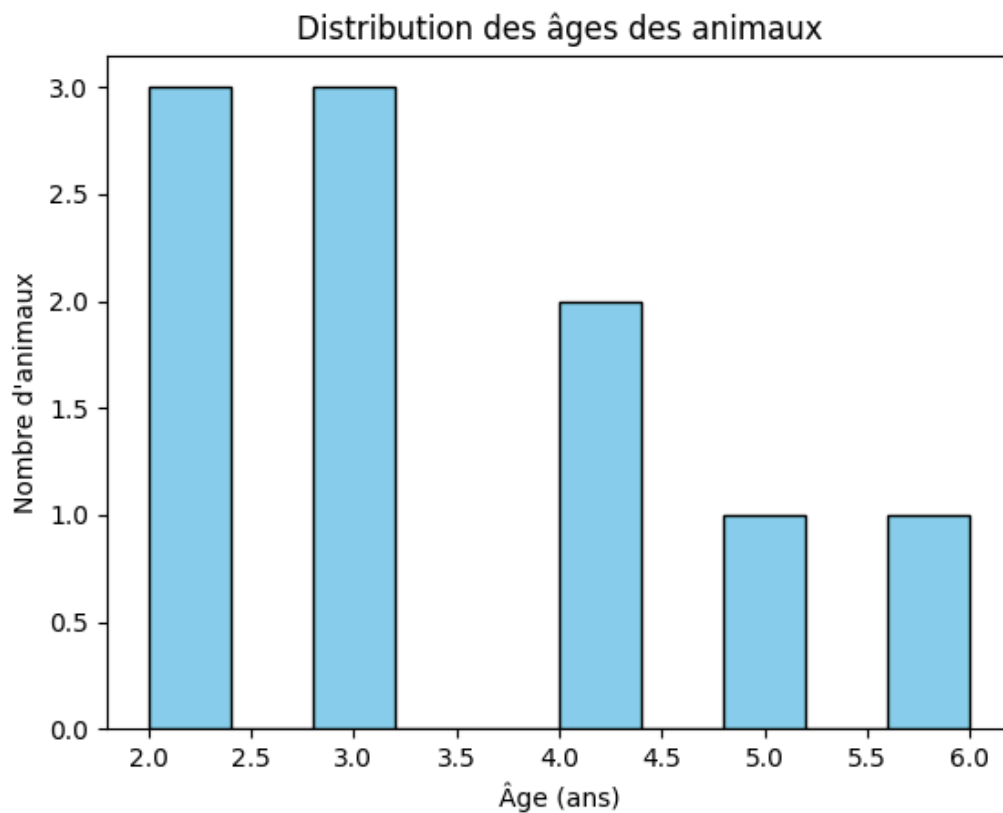
Génération de Rapports d'Activité

Le module `analyse.py` utilise **Pandas** pour traiter les données de consultations et **Matplotlib** pour générer des visualisations pertinentes. 15

- **Rapports Générés :**

- Revenu total par mois.
- Top 5 des motifs de consultation les plus fréquents.
- Histogramme de la distribution des âges des animaux.

Exemple de Graphique Généré



Conclusion et Perspectives

Conclusion et Évolutions Futures

- **Résultats Atteints :** Le projet a abouti à un système CLI fonctionnel, robuste et répondant à tous les objectifs du cahier des charges. La persistance des données est fiable et les capacités d'analyse offrent une réelle valeur ajoutée.
- **Perspectives d'Évolution :**

- **Interface Graphique :** Développer une interface plus conviviale avec Tkinter, PyQt ou une application web.
- **Base de Données :** Migrer de JSON vers un système de gestion de base de données relationnelle comme SQLite pour de meilleures performances et une meilleure intégrité des données.
- **API REST :** Exposer la logique métier via une API pour permettre une utilisation multi-postes ou l'intégration avec d'autres systèmes.