

Veterinaria

Análisis del Sistema (30 puntos)

1.1 Requisitos funcionales del sistema (5 pts)

Escribir los puntos exactos que debe cumplir tu sistema.

Registrar dueños con sus datos de contacto.

Registrar mascotas, los datos como: nombre, especie, raza, edad, dueño.

Registrar consultas, con datos como: fecha, motivo, tratamiento, mascota atendida.

Manejar un número indefinido de dueños, mascotas y consultas mediante arreglos dinámicos.

Generar reportes:

Especie más atendida.

Mascota con mayor número de consultas.

Promedio de consultas por mes.

1.2 Clases necesarias y su propósito (5 pts)

Clase	Propósito
Dueno	Es el propietario de una o varias mascotas y se guardan los datos de contacto.
Mascota	Es la mascota con especie, raza, edad y dueño
Consulta	Registra la atención médica con datos como: fecha, motivo, tratamiento y mascota
Clinica	Administra a los dueños, mascotas y consultas, además de generar reportes.
Main	Interactúa con el usuario, muestra el menú y controla el flujo.

1.3 Atributos de cada clase (10 pts)

Clase: Dueño

Atributo	Tipo de dato	Visibilidad	Propósito
Nombre	String	Private	El nombre del dueño de las mascotas
Telefono	String	Private	El telefono del dueño de las mascotas
Correo	String	private	El correo del dueño de las mascotas

Clase: Mascota

Atributo	Tipo de dato	Visibilidad	Propósito
Nombre	String	Private	Nombre de la mascota
Especie	String	Private	Que animal es
Raza	String	Private	Raza del animal
Edad	Int	Private	Edad del animal
Dueno	Dueño	Private	Dueño de la mascota
Consultas	ArrayList	Private	Lista de consultas por mascota

Clase: Consulta

Atributo	Tipo de dato	Visibilidad	Propósito
Fecha	String	Private	Fecha de la consulta
Motivo	String	Private	Motivo de la consulta
Tratamiento	String	Private	Tratamiento prescrito para la mascota
Mascota	Mascota	Private	Mascota que recibio la consulta

Clase: Clinica

Atributo	Tipo de dato	Visibilidad	Propósito
Duenos	ArrayList	Private	Fecha de la consulta
Mascotas	ArrayList	Private	Motivo de la consulta
Consultas	ArrayList	Private	Tratamiento prescrito para la mascota

1.4 Métodos de cada clase (10 pts)

Clase: Dueño

Método	Parametros : tipos de dato	Valor de retorno	Visibilidad	Propósito
getNombre	0	String	Public	Pide el nombre al dueño
getTelefono	0	String	Public	Pide el teléfono del dueño
getCorreo	0	String	Public	Pide el correo del dueño
getMascotas	0	ArrayList	Public	Listado de las mascotas
agregarMascota	Mascota	void	public	Agrega mascota al dueño

Clase: Mascota

Método	Parametros : tipos de dato	Valor de retorno	Visibilidad	Propósito
getNombre	0	String	Public	Nombre de la mascota
getEspecie	0	String	Public	Que animal es la mascota
getRaza	0	String	Public	Raza de la mascota
getEdad	0	int	Public	Edad de la mascota

getDueño	0	Dueno	Public	Dueño de la mascota
getConsultas	0	ArrayList	Public	Lista de consultas de mascota
agregarConsulta	Consulta	void	Public	Consulta a la lista de mascotas

Clase: Consulta

Método	Parametros : tipos de dato	Valor de retorno	Visibilidad	Propósito
getFecha	0	Void	Public	Fecha de la consulta
getMotivo	0	String	Public	Motivo de la consulta
getTratamiento	0	String	Public	Tratamiento de la consulta
getMascota	0	Mascota	Public	Mascota en consulta

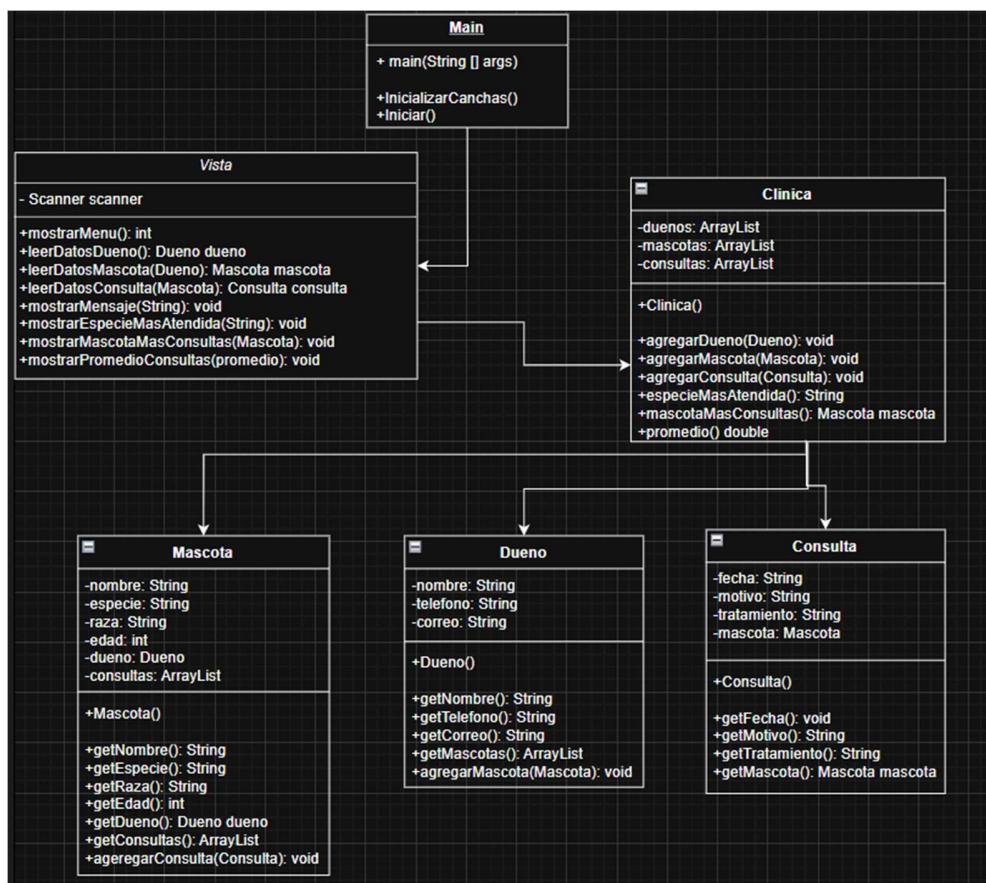
Clase: Clinica

Método	Parametros : tipos de dato	Valor de retorno	Visibilidad	Propósito
agregarDueño	Dueno	void	Public	Pide el nombre al dueño
agregarMascota	Mascota	void	Public	Pide el teléfono del dueño
agregarConsulta	Consulta	void	Public	Pide el correo del dueño

EspecieMasAtendida	0	String	Public	Listado de las mascotas
MascotaMasConsulta	0	Mascota	Public	Mascota con más consultas
Promedio	0	double	public	Promedio de mascotas por mes

2. Diseño: Diagrama de Clases (30 puntos)

- Asegúrate de mostrar atributos y métodos con visibilidad (+, -).
- Indica relaciones entre clases (asociación, agregación, etc.).
- Incluye el driver program (Main).



3. Programa (40 puntos)

En cada archivo `java`, asegurarse de incluir:

- Las clases necesarias.
- Uso adecuado de objetos.

Menú que debe implementar el driver program:

1. Opcion 1
2. Opcion 2
3. ...
4. Salir

GitHub: colocar aquí la URL:

[https://github.com/_____](https://github.com/)

Checklist antes de entregar

- Está claro el análisis?
- El diagrama tiene los elementos UML correctamente?
- Subiste tu código a GitHub con todo lo necesario?