

## III Proyecto Programado

### I. Introducción

El proyecto programado tiene como objetivo fundamental la aplicación de los conocimientos adquiridos en clases y a través de las asignaciones de tareas, entre otros manejos de estructuras tipo listas y manejo de archivos. Asimismo, promover la investigación como parte del proceso de formación profesional.

### II. Objetivo general

Incentivar al estudiante a la aplicación de los conocimientos e investigación, a partir de enfrentarse con un problema de la vida real o muy cercana a esta.

### III. Aspectos administrativos

1. A partir de la fecha de entrega usted dispondrá de tres semanas para desarrollar el proyecto. Jueves 04 mayo entrega 25 mayo
2. La entrega se realizará utilizando el tecdigital
3. Documentación adjuntar la documentación (manual de usuario) y plan de pruebas.
4. Para la revisión del proyecto, se darán citas. La asistencia a la cita es de **carácter obligatorio**, por tanto, si un estudiante no se presenta a la cita, aplicará la perdida automática de 30 pts de la nota del proyecto, y se revisará sin que el estudiante esté presente. La nota máxima posible sería un 70 en caso de tener todo el proyecto completo y funcional. Lo mismo aplica con las tardías.
5. Cualquier intento de fraude implica directamente obtener un 0 como calificación y que se aplican las medidas normativas correspondientes, de acuerdo con el reglamento interno institucional.
6. El proyecto se llevará a cabo parejas o individual
7. El lenguaje de programación a utilizar será python
8. **Recuerde que no puede cambiar el nombre de los archivos ni el formato de los archivos.**
9. No se pueden eliminar estructuras, no interfaz gráfica. Debe mantener todo lo solicitado. En caso de no presentar alguna parte del proyecto, se analizará la penalización de la nota, según el criterio del profesor, tomando en cuenta lo que el afecta la parte no realizada. Esto se puede reflejar en varias partes del proyecto. Los requerimientos no son independientes.
10. En caso de trabajo grupal se debe entregar una tabla que indique la distribución del trabajo y la forma de repartir la nota. Se aplicará regla de 3, para obtener la nota.

11. La entrega del manual y la tabla de distribución e  
de

## Funciones del sistema

### Leer Archivos:

- A. Leer Archivos: Se contará con 7 archivos con la información necesaria para que el sistema funcione correctamente. Los archivos son Países.txt, Ciudades.txt, Mascotas.txt, Clientes.txt, Visitas.txt, Tratamiento.txt y Medicacion.txt, a los cuales se definirá un estándar. Estos se leerán y serán almacenados diferentes estructuras según las indicaciones dadas. **No debe modificar el formato ni la información de los archivos, ni el nombre del archivo.**

Archivo	Estructura
Países.txt	Lista
Ciudades.txt	Lista
Clientes.txt	Lista
Mascotas.txt	Lista
Visitas.txt	Lista
Tratamientos.txt	Lista
Medicación.txt	Lista

### Formato de los Archivos: La información repetida se descarta la segunda aparición

**Países.txt** (N registro de países, no puede haber repetidos)

**Codpais;**Nombre

(Llave de búsqueda codpais).

En caso de existir código de país con el código repetido, se descarta la segunda aparición, pero no se elimina del archivo, simplemente no se carga a la lista en memoria.

**Ciudades.txt** No hay código de ciudades repetidas

**codpais;**codeciudadNombrelista

(Llave de búsqueda codeciudad)

El país se debe validar en lista en memoria, si existe un codciudad, pero el codpais no existe, no se puede cargar la ciudad a la lista. Un mismo país puede tener varias ciudades. No puede existir un mismo país con una ciudad repetida.

**Cientes.txt** No pueden existir identificación de cliente repetidos

**NumCliente**;nombre;dirección;codpais;codciudad;Telefono;**Fechaultimavisita**;  
**descuento**; **saldo**

(Llave de búsqueda Numcliente)

En caso de existir numcliente con el código repetido, se descarta la segunda aparición, pero no se elimina del archivo, simplemente no se carga a la lista en memoria.

**Fecha de ultima visita debe ser menor qua la fecha de nacimiento**

**Mascotas.txt** No pueden existir códigos de mascotas

**NumCliente**;IdAnimal;Nombre;tipomascota;raza;fechanacimiento;sexo;color; castrado;  
**fechaultimavisita**

(Llave de búsqueda IdAnimial)

El NumCliente se debe validar en lista en memoria, si existe un Idanimal, pero el numcliente no existe, no se puede cargar la mascota a la lista.

**Fecha de ultima visita debe ser menor qua la fecha de nacimiento, se valida con la fecha del cliente**

**Visitas.txt** No pueden existir visitas con código repetido.

**CodVisita**;IdAnimal; **Fechaultimavisita**;Totalfactura;formadepago(01 contado 02 crédito)

(Llave de búsqueda codvisita)

**Fecha de ultima visita debe ser menor qua la fecha de nacimiento y compara con la fecha de la mascota**

La mascota debe validar en lista en memoria, si existe un codvisita, pero el IdAnimal no existe, no se puede insertar la visita a la lista. Una misma mascota puede tener varias visitas.

**Tratamientos.txt** No pueden existir tratamientos con código repetido.

**CodTratamiento**;nombre;preciounitario, **cantidad**

(Llave de búsqueda codTratamiento)

**No se puede vender más de los que hay**

**Medicación.txt** No pueden existir medicaciones repetidas para una misma mascota para una misma fecha.

IdAnimal;codmedicacion;**fechaultimavisita**, lista de medicamentos, costo unitario;cantidad;costo total, **cantidad recetada se debe rebajar de la cantidad que se tiene en la clinica**

**El costo total debe validar que el costo total este correcto de acuerdo con el precio unitario del medicamento y la cantidad del mismo, es que por ejemplo si se intenta agregar una medicación con un costo total de 300 y el precio unitario es de 150 pero la cantidad es de 4**

Facturación se debe crear una factura de cada cliente y se tiene un registro de las facturas del cliente(cedula)

### Orden de lectura de Archivos

1. Países	Lista Países
2. Ciudades	Lista Ciudades (valida país)
3. Clientes	Lista de Clientes (valida país y ciudad)
4. Mascota	Lista de Mascotas (valida número de cliente)
5. Visitas	Lista de Visitas (valida la idanimal)
6. Tratamientos	Lista de Tratamientos (valida codtratamiento)
7. Medicación	Lista de Medicacion(valida Idanimal y codtratamiento)
8. Fechas	Fechaultima visita debe ser posterior a fecha de nacimiento. Formato de las fecha 23,04,23(día,mes,año)
9. Pagos	Formas de pago implican que si es crédito y tiene saldo se debe acumular
10. Descuentos	1-99999 tiene 3% 100000 tiene 5% 101000 a 150000 tiene 10%
11. Fecha	Fecha de ultima visita de vista debe ser igual a la ultima fecha de tratamiento
12. Fecha	Primero dos deben estar entre 1 y 31 Segundo entre 1 y 12 Tercero No se restringe
13. Saldo	Si paga de crédito se suma al saldo

### Nota

NO se puede modificar ningún archivo, borrar información, cambiar el nombre del archivo, cambiar ; por ,

SI puede leer archivo

Le corresponde al estudiante realizar un plan de pruebas. Debe venir junta con la documentación. Ese plan de prueba no puede estar basado en los archivos de ejemplo, solo tiene 3 líneas, debe utilizar archivos más amplios.

IMPOTANTE: si el proyecto no carga los archivos al inicio, el proyecto no se revisa. No se adaptan los archivos a las necesidades del proyecto, no cambia el nombre ni el formato.

## Opciones del Menú

Menú	
1. Leer Archivos	
2. Mantenimiento de la Base de Datos	
Inserción <ul style="list-style-type: none"> <li>• País</li> <li>• Ciudad</li> <li>• Cliente</li> <li>• Mascota</li> <li>• Visita</li> <li>• Tratamiento</li> <li>• Medicación</li> </ul>	<p>Se valida país no sea repetido</p> <p>Se valida que país exista y codciudad no sea repetido</p> <p>Se valida que ciudad y país existan, además que cliente no sea repetido</p> <p>Se valida que el cliente exista y que el IdAnimal no sea repetido</p> <p>Se valida el idanimal que exista y que el código de visita sea repetido</p> <p>Se valida que el código de tratamiento no sea repetido</p> <p>Se valida que el idanimal exista, que el código de medicación no sea repetido en una misma visita, que la fecha de ultima visita sea posterior a la fecha de nacimiento</p>
Búsqueda <ul style="list-style-type: none"> <li>• País</li> <li>• Ciudad</li> <li>• Cliente</li> <li>• Mascota</li> </ul>	<p>Busca un país específico</p> <p>Se busca una ciudad de un país específico.</p> <p>Se busca que ciudad y país existan, se busca el cliente.</p> <p>Se busca que el cliente exista, y se busca el IdAnimal</p>

<ul style="list-style-type: none"> <li>• Visita</li> <li>• Tratamiento</li> <li>• Medicación</li> </ul>	<p>Se busca el idanimal, si existe se busca el codigovisita, se da la información de la visita</p> <p>Se busca el codtratamiento, presenta la información</p> <p>Se da el idanimal, si existe se da el código de medicación, si existe se presenta la última medicación (lista)</p>
<p>Eliminar</p> <ul style="list-style-type: none"> <li>• País</li> <li>• Ciudad</li> <li>• Cliente</li> <li>• Mascota</li> <li>• Visita</li> <li>• Tratamiento</li> <li>• Medicación</li> </ul>	<p>Se indica código, se elimina todo lo relacionado</p> <p>Se indica país, código de ciudad, se modifica primero la residencia del cliente y luego se elimina la ciudad</p> <p>Se indica el numero de cliente y se marca como eliminado colocando un 1, al inicio de la lista con la información del cliente.</p> <p>Si se elimina el cliente, se marca como eliminada la mascota, colocando un 1 delante de la información de la mascota (en la lista)</p> <p>Se marca la mascota como eliminada</p> <p>Se solicita un código de medicamento y si existe se elimina</p> <p>No se elimina nada.</p>
<p>Modificar</p> <ul style="list-style-type: none"> <li>• País</li> <li>• Ciudad</li> <li>• Cliente</li> <li>• Mascota</li> </ul>	<p>Se modifica el nombre, indica el país se busca por código</p> <p>Se indica el código de país, código de ciudad y se modifica el nombre</p> <p>Indica el numcliente, si existe puede modificar nombre, dirección, codpais, codciudad o Telefono</p> <p>Se indica el numcliente, se valida, el idanimal se valida, si existe se puede modifica el campo del nombre y castrado</p>

<ul style="list-style-type: none"> <li>• Visita</li> <li>• Tratamiento</li> <li>• Medicación</li> </ul>	<p>Se indica CodVisita y IdAnimal, se existen se puede modificar la forma de pago</p> <p>Se busca el Codtratamiento y si existe se puede modificar el precio</p> <p>Se valida el codmedicacion,idanimal, y si existe se puede modificar la cantidad de algún medicamento y su precio y el costo total.</p>
<p>3. Reportes</p> <ul style="list-style-type: none"> <li>• Paises</li> <li>• Ciudades de un país</li> <li>• Clientes</li> <li>• Mascotas de un cliente</li> <li>• Visitas de una mascota</li> <li>• Tratamientos</li> <li>• Tratamiento de una mascota</li> <li>• Cliente con más saldo</li> <li>• Clientes de crédito</li> <li>• Clientes con más descuento</li> <li>• Ultimo tratamientos</li> <li>• Tratamiento más utilizado</li> <li>• Cliente que más facturo</li> </ul>	<p>Imprime toda la lista</p> <p>Solicita el código de un país para imprimir la lista de ciudades</p> <p>Imprime toda la lista de clientes</p> <p>Solicita el numcliente y se buscan las mascotas de ese cliente</p> <p>Se solicita idanimal y se indican las visitas de esa mascota</p> <p>Toda la lista</p> <p>Se indica el idanimal y se indica el ultimo Tratamiento</p> <p><b>NUEVO</b></p>



## Manual de usuario

Como parte del proyecto se espera se desarrolle un manual de usuario, el cual debe de estar dirigido a el usuario meta. Los reportes deben ser a un Archivo con un formato de reporte.

Coloque un nombre a su proyecto, considere que sea un nombre que llame la atención y un slogan. Ejemplo: En primer lugar numar