**¿Aun sigues creyendo todo lo que te dicen?**

Alejandra Díaz Parra

Marzo de 2019

Universidad ICESI

Departamento de Tecnologías de Información y Comunicaciones

Algoritmos y Programación I

**Documentación del Laboratorio 3**

**Identificación del problema.**

Se estaa requiriendo una aplicación de software que modele los aspectos básicos de la veterinaria “Mi pequeña mascota”.

**Especificación de requerimientos funcionales y no funcionales.**

Requerimientos funcionales:

* Registro de clientes humanos y sus mascotas.
* Hospitalización de una mascota.
* Creación de la historia clínica de una mascota, con los datos de la mascota y los de su dueño, los detalles de la hospitalización y los medicamentos recetados.
* Consultar el historial de historias clínicas.
* Consultar los datos del cliente humano.
* Consultar los datos de una mascota.
* Calcular el costo de una hospitalización.
* Mostrar un informe de la historia clínica de la mascota.
* Cambiar el estado de un cliente (activo o inactivo).

Requerimientos no funcionales:

* Calcular el costo de una hospitalización teniendo en cuenta el tipo de animal, su peso y la cantidad de tiempo que vaya a ser hospitalizado.
* Tiene 8 cuartos habilitados.
* Mostrar mensajes que orienten al usuario final.
* No sobrecargar una clase con métodos que pueden y deberían realizar otras clases.

**Trazabilidad del Análisis al Diseño**

|  |  |  |
| --- | --- | --- |
| Requerimiento Funcional | Método(s) que lo satisface(n). | Clase en la que se encuentra el método. |
| -Registro de clientes y sus mascotas.  -Hospitalización de una mascota.  -Crear historia clínica.  -Consultar los datos del cliente.  -Consultar los datos de una mascota.  -Consultar el costo de una hospitalización.  -Mostrar la historia clínica.  -Cambiar estado del cliente.  -Buscar disponibilidad de cuarto. | -addPetM(), addPet(), addPetC()    -newHospitalizationM(), hospitalizationV()  -newClinicHistory(), newClinicHistoryV()  -searchClientM(), searchClientV(), reportClient()  -searchPetM(), searchPetV(), reportPet()  -newHospitalizationM(), hospitalizationCost(), hospitalizationCostR()  -showClinicHistory(), showClinicHistoryV(), reportCH()  - changeStateCM(),changeStateCV()  -busyRoom(), busyRoomV() | -Main, Vet, Client  -Main, Vet  -Main, Vet  -Main, Vet, Client  -Main, Vet, Pet  -Main, Vet, Room  -Main, Vet, Clinic History  -Main, Vet  -Main, Vet |