|  |  |  |
| --- | --- | --- |
|  | Plantilla requerimientos  **Garous** | Formato  Presentación  Pág. 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Lugar: | Reunión vía sincrónica | **Fecha:** | *01* | *11* | *2021* | |
| **Hora (24):** | | *10* | *0* | *0* |
| Proyecto: | Desarrollo de Aplicaciones Web | | | | | |
|

|  |  |  |  |
| --- | --- | --- | --- |
| **Participantes** | | | |
| **ID** | **Nombre** | **Cédula** | **Grupo** |
| 1. | Miguel Andres Cardenas | 1000796629 | 4 |
| 2. | Juan Pablo Cifuentes | 1193563244 | 4 |
| 3. | Edgar David Veru Garzón | 1019019414 | 4 |
| 4. | Angie Tatiana Murillo | 1019602360 | 4 |
| 5. | Arleris Gracia Moreno | 1053793088 | 4 |

|  |
| --- |
| **Objetivo** |
| 1. Desarrollar un software (aplicación Web) para solventar las necesidades veterinarias desde casa para mascotas, con el fin de dar una hospitalización domiciliaria según los síntomas del paciente/mascota. |

|  |
| --- |
| **Descripción del detalle o historia del proyecto según el usuario** |
| 2 crear una aplicación web para la Ucaldas con el propósito de ayudar al tema de hospitalización de mascotas, donde el veterinario debe evaluar la condición del paciente y de acuerdo con su diagnóstico, decidirá si es apto para seguir su tratamiento en casa.  2.1. El dueño de la mascota debe facilitar los datos de la mascota como son: Id, Nombre del Paciente, Fecha de Nacimiento, Color, Sexo y Estado Salud.  La aplicación debe proveer las siguientes funcionalidades:  2.2. CRUD (Propietario). Para este registro, el usuario debe ingresar a la plataforma y luego seleccionar la opción registrar, en donde deberá suministrar la siguiente información para crear su cuenta: Usuario, Contraseña y correo electrónico. CRUD (Autenticación), crear, guardar y actualizar Datos.    2.3. Proveer una función del programa donde los usuarios puedan acceder al sistema, si se autentican, usando Nombre de Usuario y contraseña, para acceso exitoso.    2.4. Ofrecer una vista para el registro de mascotas por el usuario, cuando esté autenticado. Es decir, el usuario puede hacer CRUD (mascotas). En la creación de una mascota debería ingresar una referencia (un identificador del propietario) y nombre del paciente.  2.5. Para el registro de la historia veterinaria, el propietario deberá poder registrar los síntomas y tomar los signos vitales, asignar un médico veterinario, auxiliar veterinario para darle el soporte virtual. (Todo desde el programa para mascotas).  2.6. Analizar los datos registrados del paciente para dar soporte a los veterinarios en la toma de decisiones en el cuidado de la mascota con hospitalización o visita en casa  2.7. El Acudiente o dueño debe llevar un Histórico de los registros de los signos vitales (de manera manual) de la mascota para informar al auxiliar veterinario.  2.8. Permitir la consulta de la historia Clinica de cada paciente asignado para el veterinario y el auxiliar veterinario y adicional que las sugerencias de cuidado queden registradas, pero puedan ser visibles por el acudiente o dueño de la Mascota vía web desde cualquier dispositivo.  2.9. Ofrecer una vista amigable con el usuario, una interfaz sencilla y estilo minimalista. |

|  |  |  |
| --- | --- | --- |
|  | Plantilla requerimientos  **Garous** | Formato  Análisis Sistema  Pág. 2 |

**DESCRIPCIÓN DEL SISTEMA**

|  |
| --- |
| **REQUERIMIENTOS** |
| (PROPIETARIO REGISTRAR).  Para este registro, el usuario debe ingresar a la plataforma y luego seleccionar la opción REGISTRAR en donde debera suministrar la siguiente información para registrar una nueva cuenta: Nombre de usuario, contraseña y Correo Electronico. |
| (PROPIETARIO EDITAR).  El usuario puede ingresar a la plataforma y luego seleccionar la opción EDITAR en donde puede cambiar la siguiente información de su cuenta: Nombre de usuario, contraseña y correo electrónico. |
| (PROPIETARIO ELIMINAR).  El usuario ingresa a la plataforma y puede elegir la opción ELIMINAR en donde puede suprimir su cuenta, quedando fuera de la base del sistema. Elimina la información de los campos: Nombre de usuario, contraseña y correo electrónico. |
| (PROPIETARIO VALIDAR).  Este debe VALIDAR el ingreso al Sistema, proveer un portal de acceso, donde los usuarios puedan acceder al sistema, donde se autentiquen, usando, usuario y contraseña. En caso de que el propietario no este registrado lo enviará a registro Web.  PROPIETARIO REGISTRAR, donde ingresará los datos y verificará que los campos no estén vacíos, en caso de no estar llenos, este volverá a pedir los datos, en caso contrario, enviará a VALIDAR, para su ingreso y aquí validará si los datos ingresados son correctos o no y permitirá el acceso. |
| (REGISTRAR).  Ofrecer una vista para el registro del usuario, cuando esté autenticado. Es decir, el usuario puede hacer CRUD (Autenticación). En la creación de una cuenta se deberá ingresar los datos y una referencia o ID (un identificador para la mascota), Nombre del paciente, Fecha de Nacimiento, Color, Sexo y Estado Salud. |
| (EDITAR).  El usuario puede ingresar a la plataforma y luego seleccionar la opción EDITAR en donde puede cambiar la siguiente información: nombre de usuario, contraseña, correo electrónico, o ID (un identificador para la mascota), Nombre de la mascota, Fecha de Nacimiento, Color, Sexo y Estado Salud. |
| (ELIMINAR).  El usuario ingresa a la plataforma y puede elegir la opción ELIMINAR en donde puede suprimir su cuenta o eliminar datos que no sean correctos, quedando fuera de la aplicación web y no accediendo a usar el programa. Elimina la información de los campos como son: nombre de usuario, contraseña, correo electrónico, ID (un identificador para la mascota), Nombre de la mascota, Fecha de Nacimiento, Color, Sexo y Estado Salud. |
|  |
| (ADMIN CONSULTAR): Para las consultas de Mascotas y agendamientos el auxiliar veterinario verificara la información de los pacientes, Datos Animales, datos del propietario, Telefono de contacto y (latitud/longitud) dirección georreferenciada para informar al veterinario (Fecha, Dia y hora de Visita). |
| (ADMIN CAMBIAR): El auxiliar veterinario podrá realizar los cambios de Citas, veterinarios, Edición de datos de los usuarios y modificación de agendas de los veterinarios sean citas virtuales o presenciales. |
| (ADMIN GUARDAR): El auxiliar veterinario podrá realizar los cambios de datos Globales de la aplicación web y al finalizar presionar GUARDAR para que los datos queden almacenados en las bases de datos. |

**INPUT-OUTPUT-PROCESS ACTIVITY**

|  |  |  |  |
| --- | --- | --- | --- |
| **DATOS** | **ENTRADA** | **SALIDA** | **PROCESO** |
| **Propietario** | - **Registrar** los datos, por primera vez: Nombre de Usuario, Contraseña, email | - Verificar campos no vacíos | - Agregar Cuenta de Usuario |
| **Propietario** | - **Actualización** de los datos existentes: Nombre de Usuario, Contraseña, email | - Confirmar ingreso cambios del Propietario | - Edición de Propietario |
| **Propietario** | - **Eliminar** un propietario activo, se suprimirá de la base de datos. | - Confirmar eliminación, ¿Está seguro? | - Edición de Propietario Activo |
| **Propietario** | - **Login** solicita los datos para ingresar a la plataforma: name User + Password. | - Confirmar datos de ingreso, si los datos NO son correctos ¿Datos incorrectos? Usuario o Contraseña no corresponden | - Login propietario |
| **Propietario** |  | - **Genera** una lista Veterinarios disponibles | - Lista propietario, muestra el contenido de la base de datos como son: Fechas y veterinarios. |
| **Propietario/Admin** | - **Registrar** los datos de los **Paciente** que se agendaran con los sgtes datos: ID, nombre del paciente, Fecha de Nacimiento, Color, Sexo y Estado Salud. | - Verificar campos no vacíos | - Agregar **Paciente** |
| **Propietario/Admin** | - **Actualiza** la información existente del **Paciente**: ID, nombre del paciente, Fecha de Nacimiento, Color, Sexo y Estado Salud. | - Confirmar ingreso cambios del **Paciente** ¿Está seguro de realizar los cambios? | - Edición de **Paciente** |
| **Admin** | - **Eliminar** un **Paciente** que no se atenderá, se suprimirá de la lista de pacientes (Paciente Inactivo). | - Confirmar eliminación, ¿Está seguro, desea eliminar paciente? | - Eliminación de **Paciente** Inactivo |
| **Admin** | -El Auxiliar Consulta y Realiza cambios en los formularios de los pacientes via web y presiona Guardar una vez finalizada todas las modificaciones. | Confirmar Guardar  ¿Está Seguro de que quiere realizar los cambios del Formulario actual? | - Consulta y Guardar **Admin** |
| **Sistema** |  | - **Crea una notificación** para el propietario con los datos del: ID, nombre del paciente, Fecha de Visita, Dirección y Telefono de contacto, datos que llegan via movil e email. | -Notifica Sistema |
| **Sistema** |  | - **Crea una notificación** para el propietario que indican la atención finalizada, datos que llegan via movil e email del propietario. | - Notifica Fin Visita |

|  |  |  |  |
| --- | --- | --- | --- |
| **Admin** | Veterinario | - **Crea las historias Clínicas de los pacientes**. Filtra los datos necesarios para su labor. Agenda las reuniones para las visitas. | - Creación de Historia |
| **Admin** | Veterinario | - **Asigna el auxiliar veterinario para envio de consulta al domicilio y/o consulta virtual.** | - Asigna auxiliar |

**COMPORTAMIENTO DEL SISTEMA**

|  |  |  |
| --- | --- | --- |
| **ACTIVIDAD SISTEMA** | **VISTA** | **TAREA** |
| Creación Propietario | Pantalla Propietario solicitar datos para ingresar nuevo Propietario. Botón Guardar | Llenar datos en campo de texto: nombre de usuario, contraseña y correo electrónico, agregar a la base de datos. Botón Guardar, verificar: si los campos no están llenos (Verificar campos PROPIETARIO) |
| Edición Propietario | Pantalla Propietario solicitar datos para ingresar nuevo Propietario. | Ingresar dato nombre de usuario: contraseña y correo electrónico, agregar a la base de datos |
| Eliminación Propietario  (ADMIN) | Pantalla Propietario solicita usuario para eliminarlo del registro de Propietario s activos | Pedir datos: nombre de usuario, Botón Activo |
| Verificación (Ingresar Propietario) | Pantalla pedir contraseña |  |
| Verificar campos PROPIETARIOS | Pantalla emergente | Revisar si, campo de datos está lleno o vacío. Si alguna de las casillas está vacía, envía SMS emergente “Llenar todos los campos” y vuelve al primer texto. “Nombre de Usuario”, para completar llenado. Pero si los campos están llenos y es por primera vez, retorna a la Pantalla Login. Si está dentro de la plataforma y accede a la Pantalla Propietario, y va a Agregar o Actualizar y se a introducido algún dato, se verifican todas las casillas, si están todas con información, limpia los campos de texto y vuelve al campo “Nombre de Usuario”. |
| Creación Paciente | Pantalla Paciente solicitar datos para ingresar nuevo Paciente. Botón Guardar | Llenar datos en campo de texto: ID, nombre del paciente, Fecha de Nacimiento, Color, Sexo y Estado Salud., agregar a la base de datos. Botón Guardar, verificar: si los campos no están llenos (Verificar campos PACIENTES) |
| Edición Paciente | Pantalla Paciente solicitar datos para ingresar nuevo ítem para la visita | Ingresar dato nombre del Paciente, para modificar los datos de: ID, Fecha de Nacimiento, Color, Sexo y Estado Salud. |
| Eliminación Paciente | Pantalla Paciente solicita nombre paciente para eliminarlo del registro de Pacientes para la visita. | Pedir datos: nombre de Paciente, Botón Activo |
| Verificar campos PACIENTES | Pantalla emergente | Revisar si, campo de datos está lleno o vacío. Si están todos los campos llenos, retorna al campo nombre y limpia las casillas, de lo contrario envía SMS emergente |
| Sistema | En Pantalla e impresa. Botón Aceptar | Unir datos para notificación de Usuario, agregar a la base de datos y enviar SMS. |
| Sistema | En Pantalla Home. Botón Finalizar. | Almacena una notificación que contiene los datos de la visita, esta tiene; fecha, día, datos Animales y reporte de síntomas y signos vitales mas la sugerencia de cuidado.  Nota: Una vez guardados todos los datos la notificación la genera el sistema y debe ir directamente un SMS via movil y via email al propietario. |
| Veterinario | Pantalla Veterinario Visualiza datos de visita (Fecha, día, hora, Dirección georreferenciada, Telefono de Contacto, Nombre del propietario y Nombre del paciente). | Registra una notificación que contiene los datos de la visita, esta tiene; fecha, día, Datos Animales Dirección georreferenciada, Telefono de Contacto, Nombre del propietario y Nombre del paciente.  Nota: Una vez guardados todos los datos la notificación la genera el sistema y debe ir directamente un SMS via movil y via email al propietario. |

|  |  |  |
| --- | --- | --- |
| Veterinario | Pantalla Veterinario tiene check (Estados de Salud) del paciente para ir creando la historia veterinaria del paciente. | Registra una notificación que contiene los datos de la visita, esta tiene: Datos Animales y reporte de síntomas y signos vitales más la sugerencia de cuidado. Una vez guardados todos los datos la notificación El propietario recibe un SMS. |
|  | Plantilla requerimientos  **Garous** | Formato  Casos de Uso  Pág. 3 |

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

|  |  |  |
| --- | --- | --- |
|  | Plantilla requerimientos  **Garous** | Formato  Wireframe  Pág. 4 |

**INTERFAZ GRÁFICA – WIREFRAME**

Interfaz de usuario gráfica

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

|  |  |  |
| --- | --- | --- |
|  | Plantilla requerimientos  **Garous** | Formato  Diagramas UML  pág. 4 |

**Diagrama UML**

Diagrama

Descripción generada automáticamente

|  |  |  |
| --- | --- | --- |
|  | Plantilla requerimientos  **Garous** | Formato  DataBase  pág. 5 |

**Modelo Entidad-Relación DB**

**Diagrama

Descripción generada automáticamente**

**Creación DB y tablas en Azure**

|  |  |
| --- | --- |
| **- Relación: Tabla Propietario - Tabla Sistema**  Un propietario puede aparecer en muchos Sistemas | **- Relación: Tabla Sistema - Tabla Paciente**  Un Sistema puede tener muchos pacientes  (Detalles en el Sistema) |

|  |  |  |  |
| --- | --- | --- | --- |
| **CREATE DATABASE Garous** | | **CREATE TABLE Garous Sistema (**  idProd int AUTO\_INCREMENT PRIMARY KEY,  tipo varchar(12) NOT null,  codigo varchar(20) NOT null,  producto varchar(50) NOT null,  stock int NOT null DEFAULT 0,  valor double NOT null,  venta varchar(2) NOT null  ) | |
| **CREATE TABLE Garous.Propietario(**  idEmp int AUTO\_INCREMENT PRIMARY KEY,  identificacion varchar(30) NOT null,  nombres varchar(30) NOT null,  correo varchar(50) NOT null,  usuario varchar(30) NOT null,  clave varchar(20) NOT null,  activo varchar(2) NOT null  ) | | **CREATE TABLE Garous Sistema(**  idFact int AUTO\_INCREMENT PRIMARY KEY,  numFact varchar(10) NOT null,  fecha date NOT null,  (usuario) idEmp int NOT null,  (producto) idProd int NOT null,  cantProd int NOT null DEFAULT 0,  valorProd double NOT null,  FOREIGN KEY(idEmp) REFERENCES empleado(idEmp),  FOREIGN KEY(idProd) REFERENCES producto(idProd)  )  CREATE TABLE posfazzy.factura( idFact int AUTO\_INCREMENT PRIMARY KEY, numFact varchar(10) NOT null, fecha varchar(10) NOT null, usuario varchar(50) NOT null, producto varchar(30) NOT null, cantProd int NOT null DEFAULT 0, vrVenta double NOT null ); | |
|  | Plantilla requerimientos  **Garous** | | Formato  CRUD + Código  pág. 7 |

**Código del CRUD**

**Crud Paciente**

<!DOCTYPE html>

<html>

<head>

<title> #HospiEnCasaMascotas </title>

<link rel="icon" href="icono.ico">

</head>

<body>

<header>

<nav>

<ul>

<li> <a href="login.html"> <b> ingresar </b> </a></li>

<li> <a href="inicio.html"> <img src="../icono.ico"></a></li>

<li> <a href="registro.html"> <b> registrarse </b> </a></li>

<li> <a href="aboutus.html"> <b> acerca de nosotros </b> </a></li>

</ul>

</nav>

</header>

<article>

<section>

<h2> Acerca De Nosotros</h2>

<p> <b> Garous </b> es un proyecto el cual nacio de la necesidad de atender a nuestras mascotas y mantener un control de su salud periodico para prevenir posibles enfermedades, todo esto desde su hogar para poder mitigar el contagio del COVID-19</p>

</section>

</article>

<aside>

<h2> contenido </h2>

<p> Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod

tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,

quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo

consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse

cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non

proident, sunt in culpa qui officia deserunt mollit anim id est laborum. </p>

</aside>

<center>

<footer>

<h4>redes sociales </h4>

<p> para obtener informacion reciente, siguemos en nuestras <b> redes sociales </b></p>

<a href="">Facebook</a>

<a href="">instagram</a>

<a href="">twitter</a>

</footer>

</center>

</body>

</html>

\* {

padding: 0px;

margin: 0px;

}

body {

width: 100%;

height: 100%;

font-family: 'Open Sans', sans-serif;

font-weight: 300;

color: #666;

background-color: #ddd;

font-size: 16px;

line-height: 1.6em;

}

html {

width: 100%;

height: 100%;

}

label,

input,

button,

select,

textarea {

font-family: 'Open Sans', sans-serif;

font-size: inherit;

line-height: inherit;

}