**Programa de Formación:** Análisis y Desarrollo de Software.

**ID ficha:** 2669959

**Competencia:** Diseñar la solución de software de acuerdo con procedimientos y requisitos técnicos

**Resultado de Aprendizaje:** Estructurar el modelo de datos del software de acuerdo con las especificaciones del análisis.

**Actividad de Aprendizaje EA1:** **Taller de casos de estudio de consumir desde el Frontend servicios de una API REST con JavaScript.**

**Instructivo: Consumir desde el Frontend con React servicios de una API REST con JavaScript.**

**(Actividad Individual)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Descripción de la Actividad | **Presentación**:  **«Tu mejor amigo en casa»** es una organización encarga de recoger mascotas que han sido abandonadas, en su gran mayoría perros y gatos, **«Tu mejor amigo en casa»** se encarga de darles un nuevo hogar transitorio con todas las comodidades como refugio, alimentación y cuidados veterinarios, esta organización requiere servicios de desarrollador ***back-end y Front-end*** para gestionar una aplicación adaptada a móvil donde puedan realizar un CRUD (adicionar, consultar, modificar y eliminar las mascotas). | | | | | | | |
| **Listado de recursos, materiales, equipos, herramientas** | Guía de aprendizaje  Diapositivas modelo entidad relación  Diapositiva modelo lógico  Equipo de computo | | | | | | | |
| **Tipo de Evidencia** | Desempeño |  | Conocimiento |  | Producto | **X** | No produce entrega de evidencia |  |
| **Producto y Forma de Entrega** | **Producto**:  requiere servicios de desarrollador ***front-end*** para maquetar una aplicación móvil donde las personas puedan buscar y adoptar estas mascotas, además se debe de construir el backend que permita la funcionalidad de los requerimientos solicitados.  Todos los archivos de diseño que incluyen imágenes, tipografía y colores serán proporcionados para realizar dicha tarea, se busca que el maquetado final sea en React y Tailwind Css sea idéntico o lo más fiel posible al diseño proporcionado.  El resultado deberá funcionar en los navegadores modernos y se probará en google chrome. Deberá prestar mucha atención a los detalles, tamaño y peso de los textos, distancias horizontales y verticales entre elementos, eventos y transiciones.  **Pantallas de diseño:**  El resultado deberá funcionar en los navegadores modernos y se probará en google chrome. Deberá prestar mucha atención a los detalles.  **Ingreso (Login)**  El usuario podrá iniciar sesión, si los datos son incorrectos no podrá ingresar (validación del lado del servidor), los datos para el ingreso deben estar almacenados en la base de datos según diagrama proporcionado y la contraseña debe estar encriptada, además se debe generar un token en el Backend.    **Listar Mascotas (Admin)**  El usuario después de iniciar sesión vera la lista de mascotas, esta lista de mascotas debe ser cargada desde la base de datos, el usuario podrá Adicionar, Consultar, Modificar y Eliminar mascotas, esta página debe estar protegida y no se permite la visualización si no ha iniciado sesión, también es posible cerrar sesión desde el icono superior derecho aplica para las páginas de Listas, Adicionar, Consultar y Modificar.    **Adicionar Mascotas (Create)**  Solo los usuarios que han iniciado sesión podrán adicionar mascotas, todos los datos solicitados son obligatorios y se validaran del lado del servidor, los datos de Raza, Categoría y Genero deben ser cargados desde las base de datos, al guardar los datos se almacenaran en la base de datos y la foto de la mascota se subirá al servidor local en una carpeta llamada img/ en el directorio /public.    **Consultar Mascotas (Read)**  Solo los usuarios que han iniciado sesión podrán consultar mascotas, todos los datos deben ser cargados desde las base de datos y que correspondan a la mascota seleccionada, importante que estos datos no rompan el maquetado proporcionado.    **Modificar Mascotas (Update)**  Solo los usuarios que han iniciado sesión podrán modificar mascotas, todos los datos deben ser cargados desde las base de datos y que correspondan a la mascota seleccionada, los datos solicitados son obligatorios y se validaran del lado del servidor, los datos de Raza, Categoría y Genero deben ser cargados desde las base de datos y se debe mostrar el dato que corresponda a la información de dicha mascota, al modificar los datos se reemplazaran en la base de datos, si se cambia la foto de la mascota se subirá al servidor local en una carpeta llamada uploads/ en el directorio principal de la aplicación.    **Eliminar Mascotas (Delete)**  Solo los usuarios que han iniciado sesión podrán eliminar mascotas, al presionar el botón rojo con el icono de basurero debe elminar la mascota seleccionada en la base de datos y se debe actualizar la lista.    Modelo Lógico de la base de datos:    **Nota**: Para el desarrollo de esta actividad usted debe de utilizar para el Fornt-end Reacj, Tailwind Css, Java Script y para el Back-end debe de utilizar MongoDb como gestor de base de datos y Nodejs para el desarrollo de la Api, la Api debe permitir ver la documentación por la ruta : localhost:3000/document  **Forma de Entrega**: Se debe enviar el enlace del proyecto en Github y subir junto con el instrumento de evaluación firmado por el instructor al driver suministrado.  **Nota: Hoy un lector, mañana un líder. -Margaret Fuller.** | | | | | | | |
| **Fecha de Inicio:** | 21/05/2023 | | | | | | | |
| **Fecha de Fin:** | 04/06/2023 | | | | | | | |
| **Intensidad /duración** | 32 horas | | | | | | | |