

Proyectos..



Recetario Interactivo(Julian)

Descripción: Este proyecto es una aplicación de recetas donde los estudiantes podrán buscar recetas por ingredientes y ver información detallada de cada una, con la opción de guardarlas en favoritos

Requerimientos:

- Búsqueda de recetas: Permitir a los usuarios ingresar un ingrediente y obtener una lista de recetas (usando una API de recetas gratuita). OPCIONAL.
- Detalles de la receta: Al hacer clic en una receta, mostrar más detalles en un modal (ingredientes, pasos, imagen).
- Guardar en favoritos: Añadir un botón para guardar recetas favoritas en localStorage para que persistan entre recargas de la página.

Galería de Imágenes con Desestructuración y Modales

Descripción: Una aplicación de galería de imágenes que muestra fotos populares y permite al usuario ver detalles de cada imagen en un modal, usando desestructuración para organizar la información de la API.

Requerimientos:

- Muestra de imágenes: Cargar imágenes desde una API de fotos gratuita (como Unsplash) y mostrarlas en un grid.
- Modal de detalles: Al hacer clic en una imagen, abrir un modal con detalles como el nombre del fotógrafo, la descripción y el enlace a la imagen original.
- Favoritos: Permitir guardar imágenes en una lista de favoritos en localStorage.

Andres

Sistema de Encuestas en Tiempo Real

Descripción: Este proyecto es una aplicación de encuestas en la que los estudiantes pueden crear preguntas y opciones de respuesta, ver los resultados en tiempo real, y guardar las encuestas en localStorage. Además, este proyecto les ayudará a practicar el patrón Singleton para la gestión de encuestas.

Requerimientos:

1. Creación de encuestas: Los estudiantes pueden agregar preguntas y opciones de respuesta. Cada encuesta debe tener un título y varias opciones de respuesta.
2. Votación en tiempo real: Los usuarios pueden seleccionar una opción de respuesta, y los resultados deben actualizarse dinámicamente en la página.
3. Guardado de encuestas: Guardar las encuestas en localStorage para que persistan entre recargas de la página.
4. Vista de resultados: Mostrar los resultados de la encuesta en gráficos simples usando barras, mostrando el porcentaje de votos para cada opción en tiempo real.

Juan Andres

Trivia de Conocimientos Generales

Descripción: Este proyecto es una aplicación de trivia que presenta preguntas de opción múltiple, obtenidas de una API de preguntas y respuestas. Los estudiantes responderán a las preguntas y verán sus resultados al final de la trivia.

Requerimientos:

1. Generación de preguntas: Obtener preguntas de una API (por ejemplo, Open Trivia Database) y mostrarlas en la interfaz.
2. Interacción de usuario: Permitir que el usuario seleccione una respuesta y pase a la siguiente pregunta.
3. Puntuación final: Calcular la puntuación en función de las respuestas correctas y mostrar un mensaje al final.
4. Almacenar historial: Guardar los resultados en localStorage para que el usuario pueda ver su rendimiento en intentos previos.

Matias

Convertidor de Moneda en Tiempo Real

Descripción: Este proyecto es una aplicación simple de conversión de moneda que permite a los usuarios seleccionar dos monedas y ver la tasa de conversión actualizada en tiempo real. La aplicación obtendrá datos de conversión desde una API de divisas y actualizará la información al instante.

Requerimientos:

1. Selección de monedas: Permitir al usuario seleccionar una moneda de origen y una moneda de destino desde un menú desplegable.
2. Conversión en tiempo real: Mostrar la tasa de conversión en función de los valores obtenidos de la API.
3. Actualización automática: Cambiar automáticamente el resultado de la conversión si el usuario modifica alguna de las monedas o el monto.
4. Historial de conversiones: Guardar las conversiones recientes en localStorage para que el usuario pueda revisarlas al volver a abrir la página.

Comparador de Modelos de Autos

Descripción: Este proyecto es una aplicación web que permite a los usuarios comparar modelos de automóviles de diferentes marcas. Los usuarios pueden buscar modelos específicos, ver detalles técnicos y comparar características como precio, consumo de combustible, potencia, y otros datos. La información se obtendrá de una API de autos, y los datos se mostrarán en una interfaz organizada.

Requerimientos:

1. Búsqueda de modelos de autos: Permitir al usuario buscar autos por marca, modelo y año.
2. Comparación de características: Seleccionar hasta tres modelos y comparar características como el precio, consumo de combustible, potencia, tipo de motor, y otras especificaciones.
3. Visualización de datos: Organizar la información en una tabla de comparación o en tarjetas para facilitar la lectura.
4. Historial de comparaciones: Guardar las últimas comparaciones en localStorage, de manera que el usuario pueda acceder a ellas al recargar la página.
5. Filtros y ordenamiento: Incluir filtros como rango de precio o eficiencia de combustible, y permitir el ordenamiento ascendente o descendente.

Thank You

