Vamos a combinar las dos versiones de novedades.html para que:

1. **Mantenga la funcionalidad dinámica** de cargar noticias desde el backend (como en la segunda versión).
2. **Incluya las noticias estáticas** de la primera versión como datos iniciales en la base de datos.
3. **Muestre el tema principal y el link de la fuente** (como en la segunda versión).
4. **Sea compatible con la carga de noticias desde Excel** (como ya se implementó en cargar-noticias.html).

Además, ajustaremos el backend (server.js) para precargar las noticias estáticas en la base de datos al iniciar el servidor, de modo que las noticias de la primera versión se muestren junto con las nuevas que se carguen manualmente o desde Excel.

**Paso 1: Actualizar el Backend (**server.js**) para Precargar Noticias**

Modificaremos server.js para insertar las 10 noticias estáticas de la primera versión de novedades.html en la base de datos al iniciar el servidor, pero solo si la tabla está vacía. Esto asegura que las noticias no se dupliquen cada vez que reiniciemos el servidor.

Reemplaza el contenido de server.js con el siguiente código:

javascript

const express = require('express');

const sqlite3 = require('sqlite3').verbose();

const cors = require('cors');

const path = require('path');

const multer = require('multer');

const XLSX = require('xlsx');

const app = express();

const port = 3000;

// Configuración de multer para manejar la carga de archivos

const storage = multer.memoryStorage();

const upload = multer({ storage: storage });

// Middleware

app.use(cors());

app.use(express.json());

app.use(express.static(path.join(\_\_dirname, '.'))); // Sirve archivos estáticos (HTML, CSS, JS)

// Conectar a la base de datos SQLite

const db = new sqlite3.Database('noticias.db', (err) => {

if (err) {

console.error('Error al conectar a la base de datos:', err);

} else {

console.log('Conectado a la base de datos SQLite');

}

});

// Crear tabla de noticias si no existe

db.run(`

CREATE TABLE IF NOT EXISTS noticias (

id INTEGER PRIMARY KEY AUTOINCREMENT,

titulo TEXT NOT NULL,

contenido TEXT NOT NULL,

tema\_principal TEXT,

fuente TEXT,

fecha TEXT NOT NULL

)

`);

// Noticias estáticas para precargar

const noticiasIniciales = [

{

titulo: "Nueva Ley de Contabilidad para Pymes en 2025",

contenido: "El gobierno anunció una nueva normativa que simplifica los requisitos contables para pequeñas y medianas empresas, permitiendo una mayor flexibilidad en la presentación de balances.",

tema\_principal: "Contabilidad",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Reforma Tributaria: ¿Cómo Afectará a las Pequeñas Empresas?",

contenido: "La reforma tributaria de 2025 introduce cambios en las tasas impositivas para pymes, con beneficios para quienes inviertan en tecnología y sostenibilidad.",

tema\_principal: "Impuestos",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Inflación en Argentina: Proyecciones para el Segundo Semestre",

contenido: "Economistas prevén una desaceleración de la inflación hacia finales de 2025, pero advierten sobre la importancia de políticas fiscales más estrictas.",

tema\_principal: "Economía",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Digitalización Contable: El Futuro de las Pymes",

contenido: "La adopción de software contable basado en la nube está transformando la forma en que las pequeñas empresas gestionan sus finanzas, reduciendo costos y errores.",

tema\_principal: "Contabilidad",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Exenciones Fiscales para Nuevos Emprendimientos",

contenido: "El gobierno lanzó un programa de exenciones fiscales para startups que operen en sectores estratégicos, como tecnología y energías renovables.",

tema\_principal: "Impuestos",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Análisis de Tendencias: ¿Es el Momento de Invertir en Criptomonedas?",

contenido: "Bitcoin y Ether han mostrado una tendencia alcista en 2025, impulsados por la adopción institucional. Sin embargo, expertos advierten sobre la volatilidad de monedas como Bocacoin y USDT. Se recomienda diversificar con activos más estables como Cardano y Solana.",

tema\_principal: "Criptomonedas",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Acciones Tecnológicas: ¿Oportunidad o Riesgo?",

contenido: "Las acciones de empresas tecnológicas como Apple y Microsoft han subido un 15% en lo que va del año, pero los analistas advierten sobre una posible corrección en el mercado.",

tema\_principal: "Acciones",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Bonos del Tesoro: Una Opción Segura en Tiempos de Incertidumbre",

contenido: "Con la incertidumbre económica global, los bonos del tesoro de EE.UU. se han convertido en un refugio para inversores que buscan estabilidad.",

tema\_principal: "Bonos",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Software Contable: Las Mejores Herramientas para 2025",

contenido: "Plataformas como QuickBooks y Xero lideran el mercado de software contable, ofreciendo soluciones integradas para pymes que buscan eficiencia.",

tema\_principal: "Tecnología/Software",

fuente: "",

fecha: new Date().toISOString()

},

{

titulo: "Mercado de Capitales: Oportunidades en Mercados Emergentes",

contenido: "Los mercados emergentes, como Brasil y la India, están atrayendo a inversores con altas tasas de crecimiento y oportunidades en sectores como la tecnología y la energía.",

tema\_principal: "Mercado de Capitales",

fuente: "",

fecha: new Date().toISOString()

}

];

// Precargar noticias iniciales si la tabla está vacía

db.get('SELECT COUNT(\*) as count FROM noticias', (err, row) => {

if (err) {

console.error('Error al verificar la tabla de noticias:', err);

return;

}

if (row.count === 0) {

const stmt = db.prepare('INSERT INTO noticias (titulo, contenido, tema\_principal, fuente, fecha) VALUES (?, ?, ?, ?, ?)');

noticiasIniciales.forEach((noticia) => {

stmt.run(noticia.titulo, noticia.contenido, noticia.tema\_principal, noticia.fuente, noticia.fecha);

});

stmt.finalize(() => {

console.log('Noticias iniciales cargadas en la base de datos');

});

} else {

console.log('La base de datos ya contiene noticias, no se precargarán las iniciales');

}

});

// Ruta para obtener todas las noticias

app.get('/api/noticias', (req, res) => {

db.all('SELECT \* FROM noticias ORDER BY fecha DESC', [], (err, rows) => {

if (err) {

res.status(500).json({ error: err.message });

return;

}

res.json(rows);

});

});

// Ruta para agregar una nueva noticia manualmente

app.post('/api/noticias', (req, res) => {

const { titulo, contenido, tema\_principal, fuente } = req.body;

const fecha = new Date().toISOString();

if (!titulo || !contenido) {

res.status(400).json({ error: 'Título y contenido son requeridos' });

return;

}

db.run(

'INSERT INTO noticias (titulo, contenido, tema\_principal, fuente, fecha) VALUES (?, ?, ?, ?, ?)',

[titulo, contenido, tema\_principal || '', fuente || '', fecha],

function (err) {

if (err) {

res.status(500).json({ error: err.message });

return;

}

res.json({ id: this.lastID, titulo, contenido, tema\_principal, fuente, fecha });

}

);

});

// Ruta para cargar noticias desde un archivo Excel

app.post('/api/noticias/excel', upload.single('excelFile'), (req, res) => {

try {

if (!req.file) {

return res.status(400).json({ error: 'No se ha subido ningún archivo' });

}

// Leer el archivo Excel desde el buffer

const workbook = XLSX.read(req.file.buffer, { type: 'buffer' });

const sheetName = workbook.SheetNames[0];

const sheet = workbook.Sheets[sheetName];

const noticias = XLSX.utils.sheet\_to\_json(sheet);

if (noticias.length === 0) {

return res.status(400).json({ error: 'El archivo Excel está vacío' });

}

const stmt = db.prepare('INSERT INTO noticias (titulo, contenido, tema\_principal, fuente, fecha) VALUES (?, ?, ?, ?, ?)');

const fecha = new Date().toISOString();

noticias.forEach((noticia) => {

const { Titulo, Contenido, TemaPrincipal, Fuente } = noticia;

if (Titulo && Contenido) {

stmt.run(Titulo, Contenido, TemaPrincipal || '', Fuente || '', fecha);

}

});

stmt.finalize((err) => {

if (err) {

res.status(500).json({ error: err.message });

return;

}

res.json({ message: `Se cargaron ${noticias.length} noticias desde el archivo Excel` });

});

} catch (error) {

res.status(500).json({ error: 'Error al procesar el archivo Excel: ' + error.message });

}

});

// Iniciar el servidor

app.listen(port, () => {

console.log(`Servidor corriendo en http://localhost:${port}`);

});

**Cambios realizados**:

* Agregamos un arreglo noticiasIniciales con las 10 noticias estáticas de la primera versión de novedades.html.
* Verificamos si la tabla noticias está vacía al iniciar el servidor. Si lo está, insertamos las noticias iniciales.

**Paso 2: Actualizar** novedades.html

Combinaremos las dos versiones de novedades.html para que:

* Mantenga el diseño y el carrusel de cotizaciones de ambas versiones.
* Muestre las noticias dinámicamente desde la base de datos, incluyendo el tema principal y el link de la fuente.
* Incluya el enlace a cargar-noticias.html en el menú de navegación.

Aquí está la versión actualizada de novedades.html:

html

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="Últimas noticias y consejos para pequeñas empresas por Estudio Maniac 370.">

<title>Novedades - Estudio Maniac 370</title>

<link rel="stylesheet" href="styles.css">

<link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@700&family=Open+Sans&display=swap" rel="stylesheet">

</head>

<body>

<!-- Encabezado -->

<header>

<div class="logo">Estudio Maniac 370</div>

<nav>

<ul>

<li><a href="index.html">Inicio</a></li>

<li><a href="quienes-somos.html">Quiénes Somos</a></li>

<li><a href="servicios.html">Servicios</a></li>

<li><a href="franquicias.html">Franquicias</a></li>

<li><a href="novedades.html">Novedades</a></li>

<li><a href="contacto.html">Contacto</a></li>

<li><a href="miembros.html">Miembros</a></li>

<li><a href="planes.html">Planes</a></li>

<li><a href="cargar-noticias.html">Cargar Noticias</a></li>

</ul>

</nav>

</header>

<!-- Barra Lateral Izquierda (Mercado de Capitales) -->

<div class="left-sidebar">

<h3>Mercado de Capitales</h3>

<ul>

<li><a href="indice-sp500.html">Índice S&P 500</a></li>

<li><a href="bonos-tesoro.html">Bonos del Tesoro</a></li>

<li><a href="acciones-tech.html">Acciones Tecnológicas</a></li>

<li><a href="mercados-emergentes.html">Mercados Emergentes</a></li>

<li><a href="tendencias-2025.html">Tendencias 2025</a></li>

</ul>

</div>

<!-- Barra Lateral Derecha -->

<div class="sidebar">

<h3>Opciones</h3>

<ul>

<li><a href="#estado-contables">Estado Contables</a></li>

<li><a href="#informes">Informes</a></li>

<li><a href="#inversores">Inversores</a></li>

<li><a href="#acciones">Acciones</a></li>

<li><a href="#costos-proyectados">Costos Proyectados</a></li>

<li><a href="#bienes">Bienes</a></li>

<li><a href="#deuda">Deuda</a></li>

<li><a href="#cuenta-corriente">Cuenta Corriente</a></li>

</ul>

</div>

<!-- Sección de Novedades -->

<section id="novedades">

<h1>Novedades</h1>

<!-- Contenedor de Noticias -->

<div class="news-container" id="news-container">

<!-- Las noticias se cargarán dinámicamente aquí -->

</div>

<!-- Carrusel de Cotizaciones -->

<div class="carousel-container">

<div class="carousel">

<div class="carousel-item">

<h4>Dólar Oficial</h4>

<p>$950</p>

</div>

<div class="carousel-item">

<h4>Dólar Blue</h4>

<p>$1,200</p>

</div>

<div class="carousel-item">

<h4>Bitcoin</h4>

<p>$70,000</p>

</div>

<div class="carousel-item">

<h4>USDT</h4>

<p>$1.00</p>

</div>

<div class="carousel-item">

<h4>Bocacoin</h4>

<p>$0.05</p>

</div>

<div class="carousel-item">

<h4>Ether</h4>

<p>$3,500</p>

</div>

<div class="carousel-item">

<h4>Cardano</h4>

<p>$0.45</p>

</div>

<div class="carousel-item">

<h4>Solana</h4>

<p>$150</p>

</div>

<!-- Duplicar los elementos para un carrusel continuo -->

<div class="carousel-item">

<h4>Dólar Oficial</h4>

<p>$950</p>

</div>

<div class="carousel-item">

<h4>Dólar Blue</h4>

<p>$1,200</p>

</div>

<div class="carousel-item">

<h4>Bitcoin</h4>

<p>$70,000</p>

</div>

<div class="carousel-item">

<h4>USDT</h4>

<p>$1.00</p>

</div>

<div class="carousel-item">

<h4>Bocacoin</h4>

<p>$0.05</p>

</div>

<div class="carousel-item">

<h4>Ether</h4>

<p>$3,500</p>

</div>

<div class="carousel-item">

<h4>Cardano</h4>

<p>$0.45</p>

</div>

<div class="carousel-item">

<h4>Solana</h4>

<p>$150</p>

</div>

</div>

</div>

</section>

<!-- Pie de Página -->

<footer>

<p>© 2025 Estudio Maniac 370. Todos los derechos reservados.</p>

<p><a href="contacto.html">Contáctanos</a> | <a href="mailto:info@estudiomaniac370.com">info@estudiomaniac370.com</a></p>

</footer>

<script src="scripts.js"></script>

<script>

// Función para cargar las noticias

async function loadNews() {

try {

const response = await fetch('http://localhost:3000/api/noticias');

const noticias = await response.json();

const newsContainer = document.getElementById('news-container');

newsContainer.innerHTML = ''; // Limpiar el contenedor

noticias.forEach(noticia => {

const newsItem = document.createElement('div');

newsItem.classList.add('news-item');

newsItem.innerHTML = `

<h3>${noticia.titulo}</h3>

<p><strong>Tema Principal:</strong> ${noticia.tema\_principal || 'Sin tema'}</p>

<p>${noticia.contenido}</p>

${noticia.fuente ? `<p><strong>Fuente:</strong> <a href="${noticia.fuente}" target="\_blank">${noticia.fuente}</a></p>` : ''}

<small>Publicado el: ${new Date(noticia.fecha).toLocaleDateString()}</small>

`;

newsContainer.appendChild(newsItem);

});

} catch (error) {

console.error('Error al cargar las noticias:', error);

}

}

// Cargar las noticias al iniciar

loadNews();

// Actualizar las noticias cada 30 segundos

setInterval(loadNews, 30000);

</script>

</body>

</html>

**Cambios realizados**:

* Eliminamos las noticias estáticas del HTML, ya que ahora se cargan desde la base de datos.
* Mantuvimos el carrusel de cotizaciones y el diseño general.
* Incluimos el enlace a cargar-noticias.html en el menú de navegación.
* Mostramos el tema principal y el link de la fuente para cada noticia, como en la segunda versión.

**Paso 3: Verificar la Compatibilidad con la Carga desde Excel**

La funcionalidad de carga desde Excel ya está implementada en cargar-noticias.html y server.js. Para asegurarnos de que todo funcione correctamente:

1. **Formato del Archivo Excel**: El archivo Excel debe tener las siguientes columnas:

|  |  |  |  |
| --- | --- | --- | --- |
| Titulo | Contenido | TemaPrincipal | Fuente |
| Noticia 1 | Contenido 1 | Contabilidad | [https://ejemplo.com](https://ejemplo.com/) |
| Noticia 2 | Contenido 2 | Criptomonedas | [https://otro.com](https://otro.com/) |

* + **Título** y **Contenido** son obligatorios.
  + **TemaPrincipal** y **Fuente** son opcionales.

1. **Probar la Carga**:
   * Inicia el servidor ejecutando node server.js.
   * Abre cargar-noticias.html (http://localhost:3000/cargar-noticias.html).
   * Carga un archivo Excel con el formato indicado o agrega una noticia manualmente.
   * Abre novedades.html (http://localhost:3000/novedades.html) y verifica que las noticias se muestren correctamente, incluyendo las 10 noticias iniciales y las nuevas.

**Paso 4: Ajustar** styles.css **(si es necesario)**

Asegúrate de que styles.css incluya los estilos necesarios para el diseño de las noticias. Si no los has agregado previamente, añade lo siguiente al final de styles.css:

css

.news-item p {

margin: 5px 0;

}

.news-item a {

color: #EC407A;

text-decoration: none;

}

.news-item a:hover {

text-decoration: underline;

}

.news-item small {

display: block;

margin-top: 10px;

color: #666;

}

Esto asegura que el tema principal, el contenido, el link de la fuente y la fecha se muestren correctamente.

**Paso 5: Probar Todo el Sistema**

1. **Iniciar el Servidor**:
   * En la terminal, desde la carpeta del proyecto, ejecuta:

bash

node server.js

* + Deberías ver el mensaje: Servidor corriendo en http://localhost:3000.

1. **Verificar las Noticias Iniciales**:
   * Abre novedades.html (http://localhost:3000/novedades.html).
   * Deberías ver las 10 noticias iniciales (las de la primera versión) cargadas desde la base de datos, con su tema principal y sin fuente (ya que no se especificó en las noticias estáticas).
2. **Cargar Nuevas Noticias**:
   * Abre cargar-noticias.html (http://localhost:3000/cargar-noticias.html).
   * **Carga Manual**:
     + Ingresa un título, contenido, tema principal y link de la fuente, y haz clic en "Publicar Noticia". Verás un mensaje de confirmación.
   * **Carga desde Excel**:
     + Crea un archivo Excel con el formato indicado.
     + Selecciona el archivo y haz clic en "Cargar Noticias desde Excel". Verás un mensaje indicando cuántas noticias se cargaron.
3. **Verificar la Actualización**:
   * Vuelve a novedades.html y espera un máximo de 30 segundos (o recarga la página). Deberías ver las nuevas noticias junto con las iniciales, ordenadas por fecha (las más recientes primero).

**Notas y Mejoras**

1. **Evitar Duplicados**:
   * Si reinicias el servidor varias veces, las noticias iniciales no se duplicarán porque verificamos si la tabla está vacía antes de insertarlas.
2. **Filtrado por Tema**:
   * Podrías agregar un filtro en novedades.html para que los usuarios puedan ver noticias por tema principal (por ejemplo, un dropdown con opciones como "Contabilidad", "Impuestos", etc.).
3. **Paginación**:
   * Si la lista de noticias crece mucho, podrías implementar paginación para mostrar, por ejemplo, 10 noticias por página.
4. **Seguridad**:
   * Agrega autenticación para que solo operadores autorizados puedan acceder a cargar-noticias.html.

**Siguientes Pasos**

* **Filtrado por Tema**: Agregar un dropdown para filtrar noticias por tema principal.
* **Paginación**: Implementar paginación para manejar muchas noticias.
* **Imágenes**: Permitir cargar imágenes para las noticias, tanto manualmente como desde el Excel.

¿Te gustaría implementar alguna de estas mejoras o necesitas ayuda con otro aspecto? ¡Avísame!