// =============================================================

// NEWSZOID CHRONICLE - COMPLETE SCRIPT.JS

// =============================================================

document.addEventListener('DOMContentLoaded', () => {

// =============================================================

// HARDCODED FALLBACK DATA

// =============================================================

const HARDCODED\_DATA = {

  specialDays: {

    "01-26": {

      "en": { "title": "Republic Day (India)", "desc": "Commemorating the adoption of the Constitution of India." },

      "hi": { "title": "गणतंत्र दिवस (भारत)", "desc": "भारत के संविधान को अपनाने के उपलक्ष्य में।" }

    },

    "08-15": {

      "en": { "title": "Independence Day (India)", "desc": "Celebrating India's independence from British rule." },

      "hi": { "title": "स्वतंत्रता दिवस (भारत)", "desc": "ब्रिटिश शासन से भारत की स्वतंत्रता का जश्न।" }

    },

    "10-02": {

      "en": { "title": "Gandhi Jayanti", "desc": "Marking the birth anniversary of Mahatma Gandhi." },

      "hi": { "title": "गांधी जयंती", "desc": "महात्मा गांधी की जयंती के उपलक्ष्य में।" }

    }

  },

  news: {

    india: [

      { "en": "Supreme Court to review key constitutional amendments.", "hi": "सुप्रीम कोर्ट प्रमुख संवैधानिक संशोधनों की समीक्षा करेगा।" },

      { "en": "Parliament passes landmark bill on digital privacy.", "hi": "संसद ने डिजिटल गोपनीयता पर ऐतिहासिक विधेयक पारित किया।" },

      { "en": "RBI announces new monetary policy framework.", "hi": "आरबीआई ने नई मौद्रिक नीति की घोषणा की।" }

    ],

    world: [

      { "en": "International climate summit reaches breakthrough agreements.", "hi": "अंतर्राष्ट्रीय जलवायु शिखर सम्मेलन समझौतों पर पहुंचा।" },

      { "en": "Global tech leaders announce new AI safety protocols.", "hi": "वैश्विक तकनीकी नेताओं ने नए AI प्रोटोकॉल की घोषणा की।" }

    ]

  },

  poll: {

    "id": "weeklyPoll1",

    "en": {

      "question": "Which topic are you most interested in this week?",

      "options": ["Global Politics", "Tech Innovations", "Climate Change", "Finance"]

    },

    "hi": {

      "question": "इस सप्ताह आपकी किस विषय में सबसे अधिक रुचि है?",

      "options": ["वैश्विक राजनीति", "तकनीकी नवाचार", "जलवायु परिवर्तन", "वित्त"]

    }

  },

  facts: [

    { "en": "Honey never spoils. Archaeologists found 3000-year-old pots still edible.", "hi": "शहद कभी खराब नहीं होता।" },

    { "en": "The human brain uses 20% of the body's total energy.", "hi": "मानव मस्तिष्क शरीर की कुल ऊर्जा का 20% उपयोग करता है।" }

  ],

  horoscope: {

    "aries": { "en": "A good day for new beginnings.", "hi": "नई शुरुआत के लिए अच्छा दिन है।" },

    "taurus": { "en": "Patience will bring rewards.", "hi": "धैर्य से लाभ मिलेगा।" },

    "gemini": { "en": "Communication opens doors.", "hi": "संवाद नए अवसर लाता है।" },

    "cancer": { "en": "Focus on family and home.", "hi": "पारिवार पर ध्यान दें।" },

    "leo": { "en": "Your creativity shines bright today.", "hi": "आपकी रचनात्मकता चमक रही है।" },

    "virgo": { "en": "Attention to detail pays off.", "hi": "बारीकियों पर ध्यान फायदेमंद है।" }

  },

  trending: ["Elections", "Cricket", "AI Technology", "Climate Change"],

  stockData: [

    { symbol: "RELIANCE", price: "₹2,847.50", change: "+1.2%", trend: "up" },

    { symbol: "TCS", price: "₹4,123.25", change: "+0.8%", trend: "up" },

    { symbol: "HDFC BANK", price: "₹1,645.75", change: "-0.3%", trend: "down" }

  ]

};

// =============================================================

// GLOBAL STATE

// =============================================================

const browserLang = navigator.language || navigator.userLanguage;

let currentLang = browserLang && browserLang.startsWith('hi') ? 'hi' : 'en';

let siteData = { ...HARDCODED\_DATA };

let userLocation = { city: 'Delhi', lat: 28.7041, lon: 77.1025 };

const elements = {

  darkModeToggle: document.getElementById('darkModeToggle'),

  translateBtn: document.getElementById('translateBtn'),

  locationBtn: document.getElementById('locationBtn'),

  loginBtn: document.getElementById('loginBtn'),

  logoutBtn: document.getElementById('logoutBtn'),

  loginModal: document.getElementById('loginModal'),

  locationModal: document.getElementById('locationModal'),

  loginForm: document.getElementById('loginForm'),

  userInfo: document.getElementById('userInfo'),

  userNameEl: document.getElementById('userName'),

  participateBtn: document.getElementById('participateBtn'),

  currentDate: document.getElementById('currentDate'),

  currentLocation: document.getElementById('currentLocation'),

  changeLocationBtn: document.getElementById('changeLocationBtn'),

  useCurrentLocationBtn: document.getElementById('useCurrentLocationBtn'),

  locationSearch: document.getElementById('locationSearch'),

  popularCities: document.getElementById('popularCities'),

  specialDayTitle: document.getElementById('specialDayTitle'),

  specialDayDescription: document.getElementById('specialDayDescription'),

  countdownTimer: document.getElementById('countdownTimer'),

  pollContainer: document.getElementById('poll-container'),

  indiaNews: document.getElementById('indiaNews'),

  worldNews: document.getElementById('worldNews'),

  stockData: document.getElementById('stockData'),

  factCard: document.getElementById('factCard'),

  horoscopeGrid: document.getElementById('horoscopeGrid'),

  trendingTags: document.getElementById('trendingTags'),

  localNewsContent: document.getElementById('localNewsContent'),

  localNewsBadge: document.getElementById('localNewsBadge'),

  weatherContent: document.getElementById('weatherContent'),

  weatherLocation: document.getElementById('weatherLocation')

};

// =============================================================

// INITIALIZATION

// =============================================================

function init() {

  loadSavedPreferences();

  setupEventListeners();

  initializeContent();

  startPeriodicUpdates();

  initializeAccessibility();

  handleKeyboardNavigation();

  initializeCommentSystem();

  console.log('Newszoid Chronicle initialized');

}

function loadSavedPreferences() {

  const savedTheme = localStorage.getItem('theme') || 'light';

  applyTheme(savedTheme);

  const savedLang = localStorage.getItem('language');

  if (savedLang && (savedLang === 'en' || savedLang === 'hi')) {

    currentLang = savedLang;

  }

  const loggedInUser = localStorage.getItem('loggedInUser');

  if (loggedInUser) showUserInfo(loggedInUser);

  const savedLocation = localStorage.getItem('userLocation');

  if (savedLocation) {

    try {

      userLocation = JSON.parse(savedLocation);

    } catch (e) {

      console.warn('Failed to parse saved location');

    }

  }

}

async function initializeContent() {

  try {

    await loadExternalData();

    setDate();

    setSpecialDay();

    populateNews();

    renderPoll();

    renderStockData();

    showRandomFact();

    renderHoroscope();

    renderTrending();

    startLiveCountdown();

    initializeSocialSharing();

    initializeCarousel();

    updateLanguageDisplay();

    updateLocationDisplay();

    loadLocalNews();

    loadWeather();

    populatePopularCities();

  } catch (error) {

    console.warn('Content initialization issues:', error);

  }

}

async function loadExternalData() {

  try {

    const response = await fetch('./data.json', { cache: 'no-store' });

    if (response.ok) {

      const externalData = await response.json();

      siteData = { ...HARDCODED\_DATA, ...externalData };

    }

  } catch (error) {

    console.warn('Using hardcoded data');

  }

}

// =============================================================

// THEME & LANGUAGE

// =============================================================

function applyTheme(theme) {

  document.body.classList.toggle('dark-mode', theme === 'dark');

  if (elements.darkModeToggle) {

    elements.darkModeToggle.textContent = theme === 'dark' ? '☀️ Light Mode' : '🌙 Dark Mode';

  }

}

function toggleLanguage() {

  currentLang = currentLang === 'en' ? 'hi' : 'en';

  localStorage.setItem('language', currentLang);

  updateLanguageDisplay();

  renderPoll();

  showRandomFact();

  renderHoroscope();

  setSpecialDay();

  populateNews();

}

function updateLanguageDisplay() {

  document.querySelectorAll('[data-en]').forEach(el => {

    const text = el.dataset[currentLang] || el.dataset.en;

    if (text) {

      if (el.tagName === 'INPUT') {

        if (el.type === 'submit') {

          el.value = text;

        } else if (el.hasAttribute('placeholder')) {

          const placeholderKey = currentLang + 'Placeholder';

          el.placeholder = el.dataset[placeholderKey] || el.placeholder;

        }

      } else {

        el.textContent = text;

      }

    }

  });

}

// =============================================================

// LOCATION

// =============================================================

function updateLocationDisplay() {

  if (elements.currentLocation) elements.currentLocation.textContent = userLocation.city;

  if (elements.localNewsBadge) elements.localNewsBadge.textContent = userLocation.city;

  if (elements.weatherLocation) elements.weatherLocation.textContent = userLocation.city;

}

function populatePopularCities() {

  if (!elements.popularCities) return;

  const cities = [

    { name: 'Delhi', lat: 28.7041, lon: 77.1025 },

    { name: 'Mumbai', lat: 19.0760, lon: 72.8777 },

    { name: 'Bangalore', lat: 12.9716, lon: 77.5946 },

    { name: 'Chennai', lat: 13.0827, lon: 80.2707 },

    { name: 'Kolkata', lat: 22.5726, lon: 88.3639 },

    { name: 'Hyderabad', lat: 17.3850, lon: 78.4867 }

  ];

  elements.popularCities.innerHTML = cities.map(city =>

    `<button class="city-btn" data-city='${JSON.stringify(city)}'>${city.name}</button>`

  ).join('');

  elements.popularCities.querySelectorAll('.city-btn').forEach(btn => {

    btn.addEventListener('click', () => {

      const cityData = JSON.parse(btn.dataset.city);

      setUserLocation(cityData);

    });

  });

}

function setUserLocation(locationData) {

  userLocation = locationData;

  localStorage.setItem('userLocation', JSON.stringify(locationData));

  updateLocationDisplay();

  loadLocalNews();

  loadWeather();

  if (elements.locationModal) elements.locationModal.style.display = 'none';

}

async function detectCurrentLocation() {

  if (!navigator.geolocation) {

    alert('Geolocation not supported');

    return;

  }

  navigator.geolocation.getCurrentPosition(

    (position) => {

      const { latitude, longitude } = position.coords;

      setUserLocation({

        city: `${latitude.toFixed(2)}, ${longitude.toFixed(2)}`,

        lat: latitude,

        lon: longitude

      });

    },

    (error) => {

      console.error('Geolocation error:', error);

      alert('Unable to detect location');

    }

  );

}

async function loadLocalNews() {

  if (!elements.localNewsContent) return;

  elements.localNewsContent.innerHTML = '<div class="loading-local-news">Loading local news...</div>';

  setTimeout(() => {

    const localNews = [

      { title: `${userLocation.city} Metro expansion approved`, description: 'New routes to improve connectivity', time: '2 hours ago' },

      { title: 'Local festival preparations begin', description: 'City gears up for celebrations', time: '5 hours ago' }

    ];

    elements.localNewsContent.innerHTML = localNews.map(item => `

      <div class="local-news-item">

        <h4>${item.title}</h4>

        <p>${item.description}</p>

        <div class="local-news-time">${item.time}</div>

      </div>

    `).join('');

  }, 1000);

}

async function loadWeather() {

  if (!elements.weatherContent) return;

  elements.weatherContent.innerHTML = '<div class="loading-weather">Loading weather...</div>';

  setTimeout(() => {

    const weather = { temp: 28, description: 'Partly cloudy', icon: '⛅', humidity: 65, windSpeed: 12, feelsLike: 30, pressure: 1013 };

    elements.weatherContent.innerHTML = `

      <div class="weather-current">

        <span class="weather-icon">${weather.icon}</span>

        <div class="weather-temp">${weather.temp}°C</div>

        <div class="weather-desc">${weather.description}</div>

      </div>

      <div class="weather-details">

        <div class="weather-detail">

          <span class="weather-detail-label">Feels Like</span>

          <span class="weather-detail-value">${weather.feelsLike}°C</span>

        </div>

        <div class="weather-detail">

          <span class="weather-detail-label">Humidity</span>

          <span class="weather-detail-value">${weather.humidity}%</span>

        </div>

        <div class="weather-detail">

          <span class="weather-detail-label">Wind</span>

          <span class="weather-detail-value">${weather.windSpeed} km/h</span>

        </div>

        <div class="weather-detail">

          <span class="weather-detail-label">Pressure</span>

          <span class="weather-detail-value">${weather.pressure} hPa</span>

        </div>

      </div>

    `;

  }, 1000);

}

// =============================================================

// AUTHENTICATION

// =============================================================

function showUserInfo(name) {

  if (elements.userNameEl) elements.userNameEl.textContent = name;

  if (elements.userInfo) elements.userInfo.style.display = 'flex';

  if (elements.loginBtn) elements.loginBtn.style.display = 'none';

}

function hideUserInfo() {

  if (elements.userInfo) elements.userInfo.style.display = 'none';

  if (elements.loginBtn) elements.loginBtn.style.display = 'block';

}

function handleLogin(email, password) {

  const users = JSON.parse(localStorage.getItem('newsUsers') || '[]');

  const user = users.find(u => u.email === email && u.password === password);

  if (user) {

    localStorage.setItem('loggedInUser', user.name);

    showUserInfo(user.name);

    if (elements.loginModal) elements.loginModal.style.display = 'none';

    if (elements.loginForm) elements.loginForm.reset();

    return { success: true };

  }

  return { success: false, message: currentLang === 'hi' ? 'अमान्य लॉगिन विवरण' : 'Invalid credentials' };

}

function handleLogout() {

  localStorage.removeItem('loggedInUser');

  hideUserInfo();

}

// =============================================================

// CONTENT RENDERING

// =============================================================

function setDate() {

  if (!elements.currentDate) return;

  const now = new Date();

  const locale = currentLang === 'hi' ? 'hi-IN' : 'en-US';

  const options = { weekday: 'long', year: 'numeric', month: 'long', day: 'numeric' };

  elements.currentDate.textContent = now.toLocaleDateString(locale, options);

}

function setSpecialDay() {

  if (!elements.specialDayTitle || !elements.specialDayDescription) return;

  const now = new Date();

  const key = `${String(now.getMonth() + 1).padStart(2, '0')}-${String(now.getDate()).padStart(2, '0')}`;

  const specialDay = siteData.specialDays[key];

  if (specialDay) {

    elements.specialDayTitle.textContent = specialDay[currentLang].title;

    elements.specialDayDescription.textContent = specialDay[currentLang].desc;

  } else {

    elements.specialDayTitle.textContent = currentLang === 'hi' ? 'बस एक सामान्य दिन' : 'Just a Regular Day';

    elements.specialDayDescription.textContent = currentLang === 'hi' ? 'आज कोई विशेष कार्यक्रम नहीं है।' : 'No special events today.';

  }

}

function populateNews() {

  populateNewsList('indiaNews', siteData.news.india);

  populateNewsList('worldNews', siteData.news.world);

}

function populateNewsList(listId, newsItems) {

  const listEl = document.getElementById(listId);

  if (!listEl || !newsItems) return;

  listEl.innerHTML = newsItems.slice(0, 5).map(item => `<li>${item[currentLang] || item.en}</li>`).join('');

}

function renderPoll() {

  if (!elements.pollContainer || !siteData.poll) return;

  const pollData = siteData.poll;

  const pollLangData = pollData[currentLang];

  const votes = JSON.parse(localStorage.getItem(pollData.id + '\_votes') || JSON.stringify(new Array(pollData.en.options.length).fill(0)));

  const hasVoted = localStorage.getItem(pollData.id + '\_voted') === 'true';

  const questionHtml = `<p class="poll-question">${pollLangData.question}</p>`;

  if (hasVoted) {

    const totalVotes = votes.reduce((acc, v) => acc + v, 0);

    const resultsHtml = pollLangData.options.map((option, index) => {

      const percentage = totalVotes === 0 ? 0 : Math.round((votes[index] / totalVotes) \* 100);

      return `

        <div class="poll-result-item">

          <div class="label"><span>${option}</span><span>${percentage}%</span></div>

          <div class="progress-bar"><div class="progress" style="width: ${percentage}%;"></div></div>

        </div>

      `;

    }).join('');

    elements.pollContainer.innerHTML = questionHtml + `<div class="poll-results">${resultsHtml}</div>`;

  } else {

    const optionsHtml = pollLangData.options.map((option, index) => `<button data-index="${index}">${option}</button>`).join('');

    elements.pollContainer.innerHTML = questionHtml + `<div class="poll-options">${optionsHtml}</div>`;

    elements.pollContainer.querySelectorAll('.poll-options button').forEach(button => {

      button.addEventListener('click', (e) => handleVote(e, pollData));

    });

  }

}

function handleVote(event, pollData) {

  const selectedIndex = parseInt(event.target.dataset.index, 10);

  const votes = JSON.parse(localStorage.getItem(pollData.id + '\_votes') || JSON.stringify(new Array(pollData.en.options.length).fill(0)));

  votes[selectedIndex]++;

  localStorage.setItem(pollData.id + '\_votes', JSON.stringify(votes));

  localStorage.setItem(pollData.id + '\_voted', 'true');

  renderPoll();

}

function renderStockData() {

  if (!elements.stockData || !siteData.stockData) return;

  const stockHtml = siteData.stockData.map(stock => `

    <div class="stock-card">

      <div class="stock-name">${stock.symbol}</div>

      <div class="stock-price">${stock.price}</div>

      <div class="stock-change stock-${stock.trend}">${stock.change}</div>

    </div>

  `).join('');

  elements.stockData.innerHTML = stockHtml;

}

function showRandomFact() {

  if (!elements.factCard || !siteData.facts) return;

  const randomFact = siteData.facts[Math.floor(Math.random() \* siteData.facts.length)];

  elements.factCard.textContent = randomFact[currentLang] || randomFact.en;

}

function renderHoroscope() {

  if (!elements.horoscopeGrid || !siteData.horoscope) return;

  elements.horoscopeGrid.innerHTML = '';

  Object.keys(siteData.horoscope).forEach(sign => {

    const horoscope = siteData.horoscope[sign];

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

    card.className = 'zodiac-card';

    card.innerHTML = `<h4>${sign.charAt(0).toUpperCase() + sign.slice(1)}</h4><p>${horoscope[currentLang] || horoscope.en}</p>`;

    elements.horoscopeGrid.appendChild(card);

  });

}

function renderTrending() {

  if (!elements.trendingTags || !siteData.trending) return;

  elements.trendingTags.innerHTML = siteData.trending.map(tag => `<span class="trending-tag">${tag}</span>`).join('');

}

// =============================================================

// INTERACTIVE FEATURES

// =============================================================

function startLiveCountdown() {

  if (!elements.countdownTimer) return;

  const getNextContestEndDate = () => {

    const now = new Date();

    let endDate = new Date(now.getFullYear(), now.getMonth() + 1, 0);

    endDate.setHours(23, 59, 59, 999);

    return (now > endDate) ? new Date(now.getFullYear(), now.getMonth() + 2, 0) : endDate;

  };

  let endDate = getNextContestEndDate();

  const updateTimer = () => {

    const distance = endDate - new Date();

    if (distance < 0) { endDate = getNextContestEndDate(); return; }

    const days = Math.floor(distance / (1000 \* 60 \* 60 \* 24));

    const hrs = Math.floor((distance % (1000 \* 60 \* 60 \* 24)) / (1000 \* 60 \* 60));

    const mins = Math.floor((distance % (1000 \* 60 \* 60)) / (1000 \* 60));

    const secs = Math.floor((distance % (1000 \* 60)) / 1000);

    elements.countdownTimer.textContent = `${days}d ${hrs}h ${mins}m ${secs}s`;

  };

  updateTimer();

  setInterval(updateTimer, 1000);

}

function handleCompetitionParticipation() {

  const userLoggedIn = localStorage.getItem('loggedInUser');

  if (!userLoggedIn) {

    alert(currentLang === 'hi' ? 'कृपया पहले लॉगिन करें!' : 'Please login first!');

    if (elements.loginModal) elements.loginModal.style.display = 'block';

    return;

  }

  if (confirm(currentLang === 'hi' ? 'क्या आप प्रतियोगिता में भाग लेना चाहते हैं?' : 'Participate in competition?')) {

    alert(currentLang === 'hi' ? 'धन्यवाद! आप सफलतापूर्वक शामिल हो गए हैं।' : 'Thank you! Successfully joined.');

    elements.participateBtn.textContent = currentLang === 'hi' ? 'पंजीकृत' : 'Registered';

    elements.participateBtn.disabled = true;

    elements.participateBtn.style.opacity = '0.6';

  }

}

function initializeSocialSharing() {

  document.querySelectorAll('article').forEach(article => {

    const shareContainer = article.querySelector('.social-share');

    if (!shareContainer) return;

    const articleUrl = encodeURIComponent(window.location.href);

    const headlineEl = article.querySelector('h2, h3');

    const articleTitle = encodeURIComponent(headlineEl ? headlineEl.textContent : 'Newszoid Article');

    const twitterBtn = shareContainer.querySelector('.share-btn.twitter');

    const facebookBtn = shareContainer.querySelector('.share-btn.facebook');

    const linkedinBtn = shareContainer.querySelector('.share-btn.linkedin');

    if (twitterBtn) { twitterBtn.href = `https://twitter.com/intent/tweet?url=${articleUrl}&text=${articleTitle}`; twitterBtn.target = '\_blank'; }

    if (facebookBtn) { facebookBtn.href = `https://www.facebook.com/sharer/sharer.php?u=${articleUrl}`; facebookBtn.target = '\_blank'; }

    if (linkedinBtn) { linkedinBtn.href = `https://www.linkedin.com/shareArticle?mini=true&url=${articleUrl}&title=${articleTitle}`; linkedinBtn.target = '\_blank'; }

  });

}

function initializeCarousel() {

  const track = document.querySelector('.carousel-track');

  if (!track) return;

  const items = Array.from(track.children);

  if (items.length === 0) return;

  const prevBtn = document.querySelector('.prev-btn');

  const nextBtn = document.querySelector('.next-btn');

  let currentIndex = 0;

  function updateCarousel() {

    const width = items.length > 0 ? items[0].getBoundingClientRect().width : track.clientWidth;

    track.style.transform = `translateX(-${currentIndex \* width}px)`;

  }

  if (nextBtn) nextBtn.addEventListener('click', () => { currentIndex = (currentIndex + 1) % items.length; updateCarousel(); });

  if (prevBtn) prevBtn.addEventListener('click', () => { currentIndex = (currentIndex - 1 + items.length) % items.length; updateCarousel(); });

  if (items.length > 1) setInterval(() => { currentIndex = (currentIndex + 1) % items.length; updateCarousel(); }, 5000);

  window.addEventListener('resize', updateCarousel);

  updateCarousel();

}

// =============================================================

// COMMENT SYSTEM

// =============================================================

function initializeCommentSystem() {

  function escapeHTML(str) {

    return String(str || '').replace(/[&<>"']/g, m => ({'&':'&amp;','<':'&lt;','>':'&gt;','"':'&quot;',"'":'&#39;'}[m]));

  }

  document.addEventListener('click', function(e) {

    const btn = e.target.closest('.share-btn.comment');

    if (!btn) return;

    e.preventDefault();

    const article = btn.closest('article');

    if (!article) return;

    let articleId = article.dataset.articleId;

    if (!articleId) {

      const headlineEl = article.querySelector('.story-headline, h2, h3');

      const headline = (headlineEl && headlineEl.textContent) ? headlineEl.textContent.trim() : '';

      const slug = headline.toLowerCase().replace(/\s+/g, '-').replace(/[^a-z0-9-]/g, '').slice(0, 60);

      articleId = slug ? `article\_${slug}` : `article\_${Date.now()}`;

      article.dataset.articleId = articleId;

    }

    let box = article.querySelector('.comment-box-inline');

    if (box) { box.remove(); return; }

    box = document.createElement('div');

    box.className = 'comment-box-inline';

    box.innerHTML = `

      <textarea class="comment-input" placeholder="Write a comment..." aria-label="Write a comment"></textarea>

      <div class="comment-actions">

        <button type="button" class="submit-comment-btn">Post</button>

        <button type="button" class="close-comment-btn">Close</button>

      </div>

      <div class="comments-list" aria-live="polite"></div>

    `;

    article.appendChild(box);

    const storageKey = 'comments\_' + articleId;

    let comments = [];

    try { comments = JSON.parse(localStorage.getItem(storageKey)) || []; } catch (err) { comments = []; }

    const listEl = box.querySelector('.comments-list');

    function renderComments() {

      if (!comments.length) {

        listEl.innerHTML = '<div class="comment-item">No comments yet - be the first.</div>';

        return;

      }

      listEl.innerHTML = comments.map(c => {

        const time = new Date(c.time || Date.now()).toLocaleString();

        return `<div class="comment-item"><div class="comment-meta"><span class="comment-author">${escapeHTML(c.author || 'Guest')}</span><span class="comment-time">${time}</span></div><div class="comment-body">${escapeHTML(c.text)}</div></div>`;

      }).join('');

    }

    renderComments();

    box.querySelector('.submit-comment-btn').addEventListener('click', function() {

      const ta = box.querySelector('.comment-input');

      const text = ta.value.trim();

      if (!text) return;

      const author = localStorage.getItem('loggedInUser') || 'Guest';

      const newComment = { text, time: Date.now(), author };

      comments.push(newComment);

      try { localStorage.setItem(storageKey, JSON.stringify(comments)); } catch (err) { console.warn('Failed to save comment'); }

      ta.value = '';

      renderComments();

    });

    box.querySelector('.close-comment-btn').addEventListener('click', function() {

      box.remove();

    });

  });

}

// =============================================================

// EVENT LISTENERS

// =============================================================

function setupEventListeners() {

  if (elements.darkModeToggle) {

    elements.darkModeToggle.addEventListener('click', () => {

      const newTheme = document.body.classList.contains('dark-mode') ? 'light' : 'dark';

      localStorage.setItem('theme', newTheme);

      applyTheme(newTheme);

    });

  }

  if (elements.translateBtn) {

    elements.translateBtn.addEventListener('click', toggleLanguage);

  }

  if (elements.locationBtn) {

    elements.locationBtn.addEventListener('click', () => {

      if (elements.locationModal) elements.locationModal.style.display = 'block';

    });

  }

  if (elements.changeLocationBtn) {

    elements.changeLocationBtn.addEventListener('click', () => {

      if (elements.locationModal) elements.locationModal.style.display = 'block';

    });

  }

  if (elements.useCurrentLocationBtn) {

    elements.useCurrentLocationBtn.addEventListener('click', detectCurrentLocation);

  }

  if (elements.loginBtn) {

    elements.loginBtn.addEventListener('click', () => {

      if (elements.loginModal) elements.loginModal.style.display = 'block';

    });

  }

  document.querySelectorAll('.modal .close').forEach(btn => {

    btn.addEventListener('click', () => {

      const modal = btn.closest('.modal');

      if (modal) modal.style.display = 'none';

    });

  });

  window.addEventListener('click', (event) => {

    if (event.target.classList.contains('modal')) {

      event.target.style.display = 'none';

    }

  });

  if (elements.loginForm) {

    elements.loginForm.addEventListener('submit', (event) => {

      event.preventDefault();

      const email = document.getElementById('loginEmail')?.value || '';

      const password = document.getElementById('loginPassword')?.value || '';

      if (email && password) {

        const result = handleLogin(email, password);

        if (!result.success) alert(result.message);

      }

    });

  }

  if (elements.logoutBtn) {

    elements.logoutBtn.addEventListener('click', handleLogout);

  }

  if (elements.participateBtn) {

    elements.participateBtn.addEventListener('click', handleCompetitionParticipation);

  }

  document.querySelectorAll('.social-btn').forEach(btn => {

    btn.addEventListener('click', () => {

      const provider = btn.classList.contains('google-login') ? 'Google'

        : btn.classList.contains('microsoft-login') ? 'Microsoft' : 'Facebook';

      alert(`${provider} login would be implemented here`);

    });

  });

}

// =============================================================

// PERIODIC UPDATES

// =============================================================

function startPeriodicUpdates() {

  setInterval(() => showRandomFact(), 30000);

  setInterval(() => updateStockPrices(), 120000);

  setInterval(() => setDate(), 60000);

  setInterval(() => {

    loadExternalData().then(() => populateNews()).catch(console.error);

  }, 1800000);

}

function updateStockPrices() {

  if (!siteData.stockData) return;

  siteData.stockData.forEach(stock => {

    const changePercent = (Math.random() - 0.5) \* 4;

    const currentPrice = parseFloat(stock.price.replace(/[₹,]/g, ''));

    const newPrice = currentPrice \* (1 + changePercent / 100);

    stock.price = `₹${newPrice.toLocaleString('en-IN', { minimumFractionDigits: 2, maximumFractionDigits: 2 })}`;

    stock.change = `${changePercent >= 0 ? '+' : ''}${changePercent.toFixed(1)}%`;

    stock.trend = changePercent >= 0 ? 'up' : 'down';

  });

  renderStockData();

}

// =============================================================

// ACCESSIBILITY

// =============================================================

function handleKeyboardNavigation() {

  document.addEventListener('keydown', (e) => {

    if (e.key === 'Escape') {

      document.querySelectorAll('.modal').forEach(modal => {

        if (modal.style.display === 'block') modal.style.display = 'none';

      });

    }

    if (e.altKey && e.key.toLowerCase() === 'l') {

      e.preventDefault();

      toggleLanguage();

    }

    if (e.altKey && e.key.toLowerCase() === 'd') {

      e.preventDefault();

      const newTheme = document.body.classList.contains('dark-mode') ? 'light' : 'dark';

      localStorage.setItem('theme', newTheme);

      applyTheme(newTheme);

    }

  });

}

function initializeAccessibility() {

  const skipLink = document.createElement('a');

  skipLink.href = '#main-content';

  skipLink.textContent = currentLang === 'hi' ? 'मुख्य सामग्री पर जाएं' : 'Skip to main content';

  skipLink.className = 'skip-link';

  skipLink.addEventListener('focus', () => { skipLink.style.top = '6px'; });

  skipLink.addEventListener('blur', () => { skipLink.style.top = '-40px'; });

  document.body.insertBefore(skipLink, document.body.firstChild);

  const mainContent = document.querySelector('main') || document.querySelector('.main-content');

  if (mainContent && !mainContent.id) mainContent.id = 'main-content';

}

// =============================================================

// FIREBASE INTEGRATION (OPTIONAL)

// =============================================================

if (window.firebaseAuth) {

  window.firebaseAuth.onAuthStateChanged(user => {

    if (user) {

      const name = user.displayName || user.email || 'User';

      localStorage.setItem('loggedInUser', name);

      showUserInfo(name);

    } else {

      localStorage.removeItem('loggedInUser');

      hideUserInfo();

    }

  });

  document.querySelectorAll('.google-login').forEach(btn => {

    btn.addEventListener('click', () => {

      const provider = new firebase.auth.GoogleAuthProvider();

      window.firebaseAuth.signInWithPopup(provider).catch(err => {

        console.error('Google sign-in failed', err);

        alert('Sign-in failed: ' + err.message);

      });

    });

  });

}

// =============================================================

// UTILITY FUNCTIONS

// =============================================================

function formatIndianCurrency(amount) {

  return new Intl.NumberFormat('en-IN', {

    style: 'currency',

    currency: 'INR',

    minimumFractionDigits: 2

  }).format(amount);

}

function getRelativeTime(date) {

  const now = new Date();

  const diff = now - new Date(date);

  const seconds = Math.floor(diff / 1000);

  const minutes = Math.floor(seconds / 60);

  const hours = Math.floor(minutes / 60);

  const days = Math.floor(hours / 24);

  if (days > 0) return `${days} day${days > 1 ? 's' : ''} ago`;

  if (hours > 0) return `${hours} hour${hours > 1 ? 's' : ''} ago`;

  if (minutes > 0) return `${minutes} minute${minutes > 1 ? 's' : ''} ago`;

  return 'Just now';

}

function sanitizeHTML(str) {

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

  temp.textContent = str;

  return temp.innerHTML;

}

function isMobile() {

  return window.innerWidth <= 768 || /Android|iPhone|iPad|iPod|BlackBerry|IEMobile|Opera Mini/i.test(navigator.userAgent);

}

function smoothScrollTo(elementId) {

  const element = document.getElementById(elementId);

  if (element) element.scrollIntoView({ behavior: 'smooth', block: 'start' });

}

// =============================================================

// STARTUP & GLOBAL EXPORTS

// =============================================================

init();

window.NewszoidChronicle = {

  toggleLanguage,

  updateStockPrices,

  showRandomFact,

  renderPoll,

  setUserLocation,

  currentLang: () => currentLang,

  siteData: () => siteData,

  userLocation: () => userLocation,

  version: '2.0.0'

};

console.log('🗞️ Newszoid Chronicle v2.0 loaded successfully!');

console.log('Debug functions: window.NewszoidChronicle');

}); // End DOMContentLoaded