

I need you to update ALL URLs in my entire project to use absolute URL structures instead of relative URLs. This includes:

I need you to update ALL URLs in my entire project to use absolute URL structures instead of relative URLs. This is critical for SEO, canonical URLs, and hreflang implementation. === SCOPE OF CHANGES === 1. ****HTML/Template Files:**** - All href attributes in <a> tags - All src attributes in , <script>, <iframe>, <video>, <audio>, <source>, <embed> tags - All action attributes in <form> tags - All data attributes containing URLs (data-src, data-url, etc.) - All srcset attributes in and <picture> tags - All poster attributes in <video> tags 2. ****Meta Tags & SEO Elements:**** - og:image, og:url, og:video, og:audio URLs - twitter:image, twitter:player URLs - Canonical link tags (<link rel="canonical">) - Alternate link tags (<link rel="alternate">) - Hreflang tags (<link rel="alternate" hreflang="...">) - RSS/Atom feed URLs - Sitemap references in meta tags - Any other meta property URLs (schema.org, JSON-LD) 3. ****CSS Files:**** - All url() references for background images, fonts, and other assets - @import statements - Both inline <style> tags and external .css files 4. ****JavaScript Files:**** - All URL strings and fetch/axios/XMLHttpRequest requests - Image sources set via JavaScript (innerHTML, createElement, etc.) - Dynamic link generation - Router configurations and navigation URLs - API endpoint references 5. ****Configuration & Data Files:**** - Sitemap.xml URLs - robots.txt references - .htaccess redirects - JSON configuration files - manifest.json file URLs - Package.json homepage/repository URLs - Any config files with URL definitions 6. ****Server-Side Code:**** - Template rendering URLs (if using server-side rendering) - Redirect rules - Email template URLs - RSS feed generation === URL SPECIFICATIONS === ****Base Domains:**** - Main website: https://www.yourdomain.com - CDN for images/assets: https://cdn.yourdomain.com - API endpoint base: https://api.yourdomain.com - Blog subdomain (if applicable): https://blog.yourdomain.com - Development environment: https://dev.yourdomain.com (or specify staging URL) - Production environment: https://www.yourdomain.com ****Conversion Rules:**** - Convert ALL relative URLs to absolute URLs - Use HTTPS protocol for all internal URLs - Preserve query parameters (e.g., ?id=123&utm_source=email) - Preserve URL fragments/anchors (e.g., #section-name) - Do NOT add trailing slashes unless the original URL had them - Keep external URLs (from other domains) unchanged - Maintain case sensitivity of original URLs ****Example Conversions:**** Relative → Absolute: - `/about` → `https://www.yourdomain.com/about` - `/about/` → `https://www.yourdomain.com/about/` - `../images/logo.png` → `https://www.yourdomain.com/images/logo.png` - `contact.html` → `https://www.yourdomain.com/contact.html` - `/blog/post?id=5` → `https://www.yourdomain.com/blog/post?id=5` - `/page#section` → `https://www.yourdomain.com/page#section` - `./style.css` → `https://www.yourdomain.com/style.css` - `/assets/img/hero.jpg` → `https://cdn.yourdomain.com/assets/img/hero.jpg` ****Special Cases:**** - Root path `/` → `https://www.yourdomain.com/` - Protocol-relative URLs `//example.com` → Convert to `https://example.com` - Empty href `href=""` or `href="#"` → Leave unchanged - JavaScript void `href="javascript:void(0)"` → Leave unchanged - Mailto/tel links `mailto:`, `tel:` → Leave unchanged - Data URLs `data:image/png;base64,...` → Leave unchanged === EXECUTION PROCESS === ****Step 1: Analysis**** Before making ANY changes, please: 1. Scan all files in the

project recursively

2. Identify all files containing URLs (list them by type)
3. Count total relative URLs found in each file
4. Provide a summary report showing:
 - Total files to be modified
 - Total URLs to be converted
 - Breakdown by file type (.html, .css, .js, etc.)
 - Any potential issues or edge cases identified

****Step 2: Backup**** - Create a git commit or backup before changes - Document the current state

****Step 3: Confirmation**** - Show me the analysis and wait for my approval before proceeding

****Step 4: Conversion**** - Convert all relative URLs to absolute URLs according to specifications - Maintain code formatting and indentation - Preserve comments and documentation

****Step 5: Validation**** After conversion, verify:

- All URLs are properly formatted
- No broken links introduced
- All absolute URLs use correct domain/subdomain
- HTML/CSS/JS syntax remains valid
- Special characters are properly encoded

****Step 6: Report**** Provide a final report with:

- Number of files modified
- Number of URLs converted
- List of any URLs that couldn't be converted automatically
- Any warnings or issues encountered

=== IMPORTANT NOTES ===

- ****Do NOT modify:**** External URLs (already pointing to other domains) - ****Preserve:**** All query strings, anchors, and URL parameters exactly as they are
- ****Check for:**** Dynamically generated URLs that might need special handling
- ****Watch for:**** URLs in JavaScript strings that might be concatenated
- ****Consider:**** Different domains for different asset types (main site vs CDN)
- ****Environment:**** If you find environment variables for URLs, update those too

=== DOMAIN CONFIGURATION === Please use these domains for conversion:

- Primary domain: [INSERT YOUR MAIN DOMAIN: <https://www.example.com>]
- CDN domain: [INSERT CDN DOMAIN or use primary: <https://cdn.example.com>]
- API domain: [INSERT API DOMAIN or use primary: <https://api.example.com>]
- Current environment: [Production/Development/Staging]

Begin with the analysis phase and wait for my confirmation before making any changes.