Technical seo

Technical SEO, which is all about optimizing your website's technical aspects to ensure it is accessible, indexable, and crawlable by search engines. The goal is to make sure that search engines can efficiently discover, understand, and rank your content.

Just like we did with **On-Page** and **Off-Page SEO**, we'll break down **Technical SEO** into key elements and provide examples and best practices for each, with a focus on how they contribute to better SEO performance.

What is Technical SEO?

Technical SEO focuses on improving the **back-end structure** and **performance** of your website. It ensures that your website is easy for search engines (like Google) to crawl and index. It also improves user experience, which is an important ranking factor. Key elements of **Technical SEO** include website speed, mobile-friendliness, secure connections (HTTPS), site architecture, and more.

Key Elements of Technical SEO

1. **Website Speed Optimization**

Website speed is crucial because faster-loading websites provide a better user experience and are preferred by search engines.

- **Recommended Page Load Time**: **Under 3 seconds**.
  - **Reason**: Google has stated that faster pages are better for users. Pages that take longer than 3 seconds to load tend to have higher bounce rates and lower rankings.
- **Best Practices**:
  - Compress images (use **WebP** format or **JPEG 2000**).
  - Minimize **JavaScript** and **CSS** files.
  - Use **browser caching** to speed up loading for repeat visitors.
  - Implement **Content Delivery Networks (CDN)** to speed up content delivery.

Example of Image Optimization:

html

Copy

```
<img src="vegan-chocolate-cake.jpg" alt="Vegan Chocolate Cake" width="800" height="600">
```

- The image is compressed and optimized for faster loading.

2. **Mobile-Friendliness**

With **mobile-first indexing**, Google uses the mobile version of a website for ranking. A website that is not optimized for mobile will suffer in search rankings.

- **Best Practices**:
  - Ensure your website uses a **responsive design** (adapts to different screen sizes).
  - Optimize touch elements (buttons, navigation) for **easy use** on mobile devices.
  - Test your website's mobile performance using **Google's Mobile-Friendly Test**.

Example of Responsive Design (CSS):

css

Copy

```
/* This ensures that your site looks good on all screen sizes */@media screen and (max-width: 768px) {
body {    font-size: 14px;  }  .container {    padding: 10px;  }}
```

- The **@media query** adjusts the font size and padding for smaller screens like mobile phones.

3. **Secure Website (HTTPS)**

HTTPS (HyperText Transfer Protocol Secure) is a ranking factor, and Google gives a preference to websites that have secure connections.

- **Best Practices**:
  - Install an **SSL certificate** on your website.
  - Redirect HTTP traffic to HTTPS using 301 redirects.
  - Ensure that all internal and external links on your site are HTTPS.

Example of SSL Certificate in Action:

html

Copy

```
<!-- This is an HTTPS link --><a href="https://www.example.com/vegan-chocolate...">Check out the recipe here</a>
```

- Your website should have the "**https://**" protocol to ensure it's secure.

4. **XML Sitemap**

An **XML sitemap** is a file that lists all the important pages on your website, helping search engines understand the structure of your site and discover content to index.

- **Recommended Sitemap Size**: **Under 50,000 URLs per sitemap**.
  - **Reason**: Google recommends breaking up large sites into multiple sitemaps for better crawl efficiency.
- **Best Practices**:
  - Ensure your sitemap is **up-to-date** and includes all important pages (blog posts, product pages, etc.).
  - Submit the sitemap to **Google Search Console** and **Bing Webmaster Tools** for indexing.

Example of XML Sitemap:

xml

Copy

```
<?xml version="1.0" encoding="UTF-8"?><urlset xmlns="www.sitemaps.org - /schemas/sitemap/0.9/">  <url>
   <loc>https://www.example.com/vegan-chocolate...</loc>    <lastmod>2025-02-12</lastmod>
<changefreq>monthly</changefreq>    <priority>0.8</priority>  </url>  <!-- More URLs go here -->
</urlset>
```

- This **XML sitemap** includes a URL to the **vegan chocolate cake recipe page** with metadata like last modification date and priority.

5. **Robots.txt File**

The **robots.txt** file is used to control search engine crawlers' access to certain parts of your site. You can use it to block certain pages (like admin or duplicate content) from being crawled.

- **Best Practices**:
  - Ensure you're not blocking important pages from being crawled by search engines.
  - Use the **robots.txt** file for parts of your site that are irrelevant to SEO (e.g., admin pages, duplicate pages).
  - Use the **"Disallow"** directive wisely to prevent crawlers from wasting crawl budget on non-important pages.

Example of robots.txt:

plaintext

Copy

```
User-agent: *Disallow: /admin/Disallow: /cart/Allow: /vegan-chocolate-cake-recipe/
```

- This **robots.txt** file blocks crawlers from indexing the **/admin/** and **/cart/** pages but allows the **vegan chocolate cake recipe page** to be crawled.

6. **Structured Data (Schema Markup)**

Schema markup is a form of code you can add to your pages to help search engines better understand the content and context of your site, enhancing search results with rich snippets (like star ratings, images, etc.).

- **Best Practices**:
  - Use **JSON-LD format** for schema markup (Google recommends it).
  - Add structured data for things like **recipes**, **reviews**, **events**, and **products**.

Example of Schema Markup for Recipe:

html

Copy

```
<script type="application/ld+json">{  "@context": "Schema.org - Schema.org",  "@type": "Recipe", "name": "Vegan Chocolate Cake",  "recipeYield": "8 servings",  "recipeIngredient": [    "1 cup flour",   "1/2 cup cocoa powder",    "1 cup sugar",    "1 tsp baking soda"  ],  "recipeInstructions": [ "Preheat oven to 350°F.",    "Mix dry ingredients in a bowl.",    "Bake for 30 minutes."  ]}</script>
```

- This **schema markup** helps search engines understand that the page contains a **recipe** with ingredients and instructions.

7. **Crawl Errors & Site Audits**

Crawl errors occur when search engines cannot access or index parts of your website. It's crucial to regularly check for **crawl errors** and resolve them.

- **Best Practices**:
  - Use tools like **Google Search Console** to identify **404 errors**, broken links, or server issues.
  - Ensure all important pages are accessible by search engines and that there are no **redirect chains** or **broken links**.

Example of Identifying Crawl Errors:

- Use **Google Search Console** to check for crawl errors under **Coverage** and ensure all your pages are being indexed correctly.

8. **URL Canonicalization**

Canonicalization refers to making sure there's a single preferred version of a page to prevent duplicate content issues. Use **rel="canonical"** tags to tell search engines which version of a page is the main one.

- **Best Practices**:
  - Use **canonical tags** to indicate the preferred version of pages with similar or duplicate content.
  - Avoid **duplicate content issues** caused by URL parameters or multiple versions of the same content.

Example of Canonical Tag:

html

Copy

```
<link rel="canonical" href="https://www.example.com/vegan-chocolate... ">
```

- This tells search engines that the page https://www.example.com/vegan-chocolate... is the **preferred version**.

Summary of Recommended Technical SEO Elements

	Element	Recommended Best Practices
1	**Website Speed Optimization**	Load under **3 seconds**; optimize images, JavaScript, and CSS
2	**Mobile-Friendliness**	Use **responsive design**; test mobile performance regularly
3	**Secure Website (HTTPS)**	Ensure **SSL certificate** is installed and all links are HTTPS
4	**XML Sitemap**	Submit to **Google Search Console**; update regularly
5	**Robots.txt File**	Block non-essential pages; avoid blocking important pages
6	**Structured Data (Schema)**	Use **JSON-LD** format for content like recipes, reviews, etc.
7	**Crawl Errors & Site Audits**	Regularly check for **404 errors** and **broken links**
8	**Canonicalization**	Use **rel="canonical"** tags for duplicate content management

---

Conclusion

Technical SEO is about creating a strong foundation for your website so search engines can easily access, crawl, and index your content. By ensuring your website is fast, secure, mobile-friendly, and free from crawl errors, you'll improve both user experience and search engine visibility. Keep optimizing technical aspects as search engines and user expectations evolve to maintain a high-performing website.