**Crawlability and Robots.txt Analysis Report for Al Jazeera**

**1. Introduction**

This report documents the crawl permissions and restrictions for the Al Jazeera website, based on an analysis of the robots.txt file, sitemap URLs, RSS feeds, and API endpoints. The goal is to ensure polite and compliant web crawling.

**2. Robots.txt Rules**

* The robots.txt file specifies which paths are allowed or disallowed for web crawlers (user-agents).
* The wildcard user-agent \* applies to all bots unless otherwise specified.

**2.1 General Rules for All User-agents (\*)**

| **Rule Type** | | **Path** | **Description** |
| --- | --- | --- | --- |
| Disallow | /api | | Crawlers are blocked from accessing any /api paths. |
| Disallow | /asset-manifest.json | | The specific file asset-manifest.json is disallowed. |
| Allow | /search/$ | | Exact URL /search/ is allowed. |
| Disallow | /search/ | | Any URL starting with /search/ (except exact /search/) is disallowed. |
| Disallow | /home/search?q= | | URLs starting with /home/search?q= are disallowed. |

**2.2 Complete Disallow for Specific User-agents**

The following user-agents are fully blocked (disallowed from crawling any page):

* anthropic-ai
* ChatGPT-User
* ClaudeBot
* Claude-Web
* cohere-ai
* GPTBot
* PerplexityBot
* Bytespider

**3. Sitemap URLs**

Multiple sitemap files are provided to help crawlers discover public content:

* <https://www.aljazeera.com/sitemap.xml>
* <https://www.aljazeera.com/news-sitemap.xml>
* <https://www.aljazeera.com/sitemaps/article-archive.xml>
* <https://www.aljazeera.com/sitemaps/article-new.xml>
* <https://www.aljazeera.com/sitemaps/video-archive.xml>
* <https://www.aljazeera.com/sitemaps/video-new.xml>

These sitemaps are organized by content type (news, articles, videos) and include URLs with last modification dates for efficient crawling.

**4. Crawl Delay**

* No explicit Crawl-delay directive is specified in the robots.txt.
* It is recommended to implement a polite crawling delay, such as 2 seconds between requests, to avoid overloading the server.

**5. RSS Feeds and API Endpoints**

**5.1 RSS Feeds**

* Al Jazeera provides official RSS feeds for different content categories, such as:
  + General news: <https://www.aljazeera.com/xml/rss/all.xml>
  + Economy: <https://www.aljazeera.com/xml/rss/economy.xml>
  + Opinion: <https://www.aljazeera.com/xml/rss/opinion.xml>
  + Sports: <https://www.aljazeera.com/xml/rss/sport.xml>
* These RSS feeds provide structured updates suitable for data collection and are recommended over scraping HTML pages or using private APIs.

**5.2 API Endpoints**

* API paths (e.g., /api) are disallowed in robots.txt and should not be accessed by crawlers.
* Attempts to crawl or scrape private API endpoints violate site policy and may cause blocking.
* Instead, use sitemaps and RSS feeds for collecting content data.

**6. Crawling Implementation**

* Strictly obey robots.txt rules and avoid disallowed paths.
* Identify your crawler with a clear User-Agent string.
* Introduce delays between requests (at least 2 seconds) to reduce server load.
* Use sitemaps for discovering URLs rather than guessing or scraping.
* Prefer RSS feeds to get the latest content updates reliably.

A Python script was implemented to demonstrate crawling of Al Jazeera's public content sections. The script visits five main content URLs, extracts the <title> tag from each page using BeautifulSoup, and introduces a delay of 2 seconds between requests to avoid overwhelming the server. All HTTP requests are made with a realistic User-Agent string.

Introducing a 2-second delay (time.sleep(2)) between each request ensures the crawler behaves responsibly by simulating human-like browsing behavior. This practice helps prevent triggering anti-bot mechanisms and reduces the load on the website’s servers. Such delays are considered best practice for ethical and legal web scraping, especially when no explicit crawl-delay directive is provided in the robots.txt file.

All visited sections returned titles in a consistent format:  
"[Section Name] | Today's latest from Al Jazeera"  
This consistency is due to Al Jazeera’s use of a uniform HTML <title> template across major content sections. Although the overall structure is the same, the section name at the beginning of the title distinguishes the topic of each page. For example:

* News → "News | Today's latest from Al Jazeera"
* Economy → "Economy | Today's latest from Al Jazeera"
* Sport → "Sport | Today's latest from Al Jazeera"

**2-Articles Crawling**

The **HTML Content Extractor** is a Python-based tool that scrapes article data—such as titles, dates, authors, and text—from news websites using **Beautiful Soup** or **Scrapy**. It supports **pagination**, cleans and structures the content, and saves it to **CSV** or **SQLite**. The tool includes **error handling** to retry failed requests and is designed to collect **50+ articles** efficiently for analysis.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

3- **JavaScript Content**

This module handles JavaScript-rendered content using tools like **Selenium** or **Playwright**. It ensures effective scraping of dynamic pages, supports stealth techniques, and helps with JS-based pagination.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.