

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | |
|  | | Assignment 2 – Web Crawler | | | | |  | |
|  |  | | | | | | |  |
|  | | | |  |  | | | |
|  | | | | **Web Spider**  **PHP** |  | | | |
|  | | | | Name: JazibCMS: 394006Class: BESE-12Section: A4th December 2023Web-EngineeringDr. Fahad Satti |  | | | |
|  | | |  | | |  | | |

**Introduction:**

This web-spider, which is also called as crawler is implemented with all the given requirements through PHP. PHP comes up with a Document Object Model (DOM) API which can be easily accessed to get the HTML tree of any document given its URL.

**Tech Stack Used:**

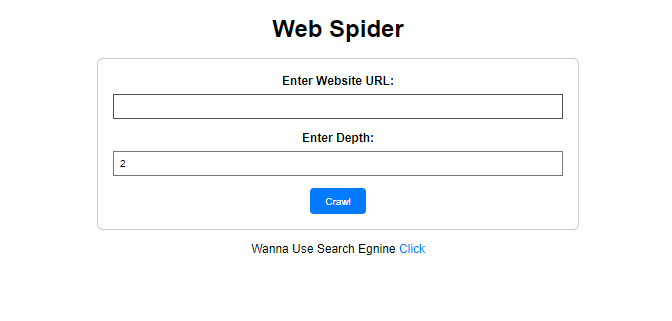
1. PHP for Crawling

2. HTML for Document Front-end

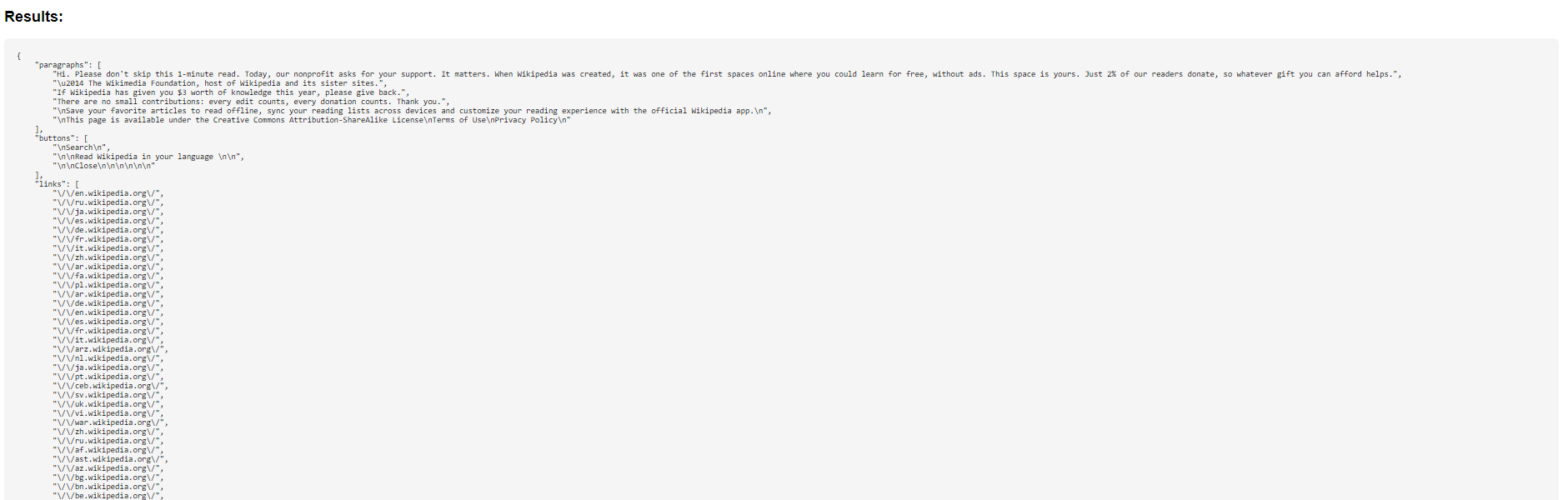
3 .CSS for beautifying (Embedded CSS)

**GitHub Repo:** <https://github.com/Jazib-2004/Web-Crawler-PHP-/>

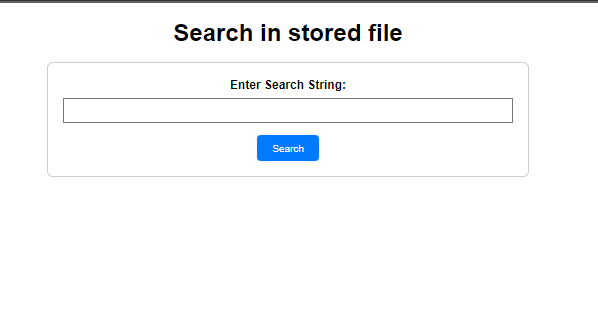
**Snapshots of the Interface:**

****

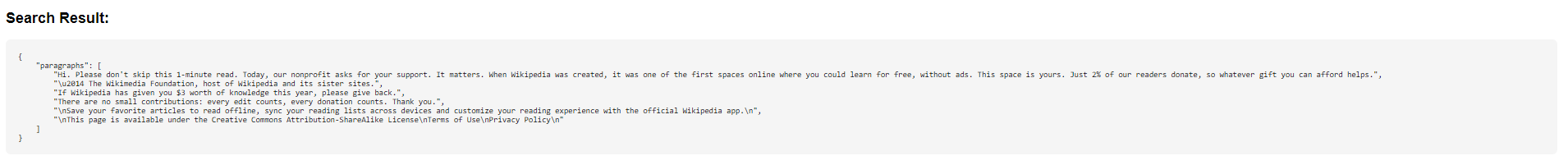
**Output Results after Crawling:** [**https://www.wikipedia.org/**](https://www.wikipedia.org/)

****

**Search Engine:**

****

**Output after searching: “Wikipedia”.**

****

**Requirements completed:**

1. URL Queue
2. **Crawling**
3. **HTML Parsing**
4. **URL Extraction**
5. **Depth Limit**
6. **Output**
7. **Content Search Module**
8. **Error Handling**

**Bonus Requirements:**

1. **Persistent Storage**