Mahatma Education Society's
Pillai HOC College of Engineering & Technology, Rasayani, Raigad

MINI PROJECT - GROUP 04

Academic Year- 2021-22

FING SEARCH ENGINE

TEAM MEMBERS:

1. Sahil Sheikh - 68
2. Pranjal Sonawane - 69
3. Sayali Thombare - 70

Guided By:
Ms. Rajashree Gadhve

INTRODUCTION

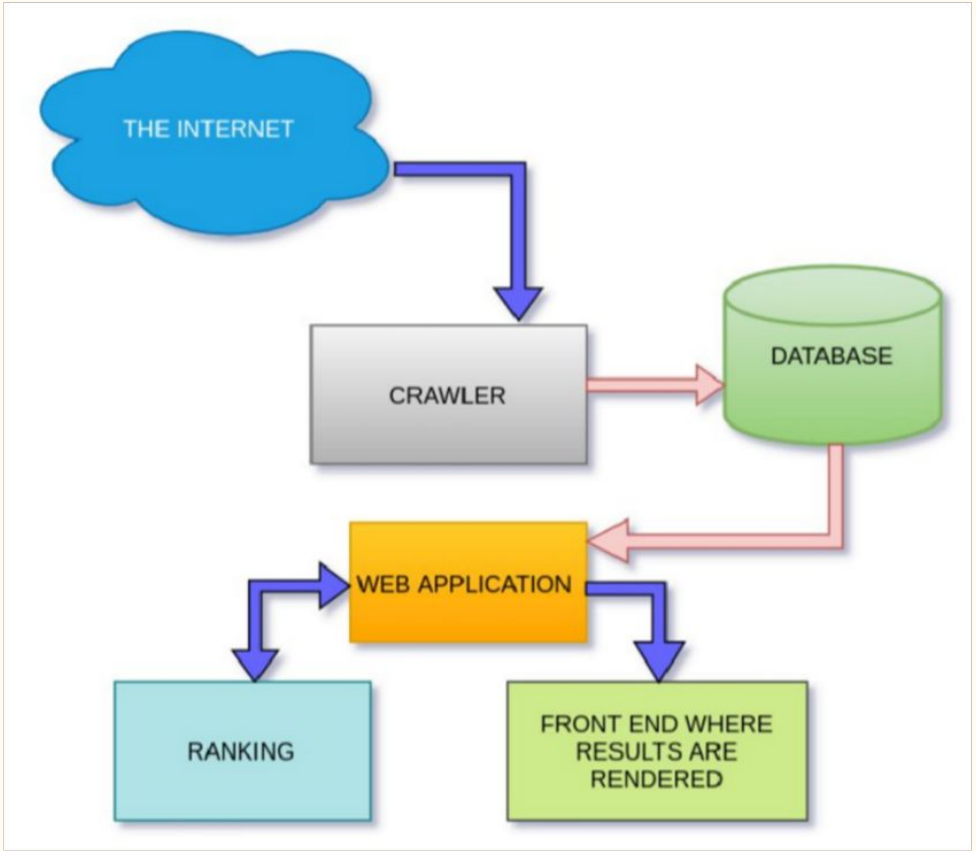
Fing search engine is a software system that is designed to carry out web searches. They search the web in a systematic way for particular information specified in a textual web search query. In this project we are going to build a search engine which can show us the search results it fetched from a internet . Fing provides all basic need for searching requirement to the user .

OBJECTIVES

Because every time, every time more and more web pages are added to the Internet to achieve top Ranking Search Engine.

The relevance of search engine algorithms are proprietary. Because of the mysterious nature of search engine relevancy algorithms, the process of achieving a higher ranking in organic search engine results.

ARCHITECTURE / FRAMEWORK



ALGORITHM & PROCESS DESIGN

APPROACH FOR BUILDING THE SEARCH ENGINE:

A search engine performs four basic processes:

Crawling, Indexing, Searching, Ranking.

CRAWLING:

Web search engines get their information by crawling from site to site. The crawler is provided with an entrypoint from which it starts collecting the links and text data and storing them in the database.

INDEXING:

Indexing means associating the data found on the web pages with the domain it was found on and HTML fields. The way data is stored in the database is a major contributor to the efficiency of the search engine.

SEARCHING:

As the name implies searching means to search the database for relevant results to the search query.

RANKING:

Ranking means to rank the search results found from the above operation in order of their relevance to the user. The better ranking system results in a better search experience.

WEB CRAWLER

The word “crawler” itself can be intimidating to many people but it is basically a script having a few lines of code.

A web crawler, spider, or search engine bot downloads and indexes content from all over the Internet. They're called web crawlers because crawling is the technical term for automatically accessing a website and obtaining data via a software program.

The crawler goes from page to page and stores the data fetched from it in the database, so that the information can be retrieved when it's needed.

APPROACH TO BUILD THE CRAWLER

We are going to use the following python libraries to achieve the task

1. requests library to fetch the pages.
2. beautifulsoup4 to parse the response received from the response object.
3. pymongo to connect to mongodb where we are going to store the data.

Yes that's it, that's all we need.
We will build a python class named Crawler inside the crawler.py file.
The first thing we want to do is to make a connection with our database using "pymongo" library.

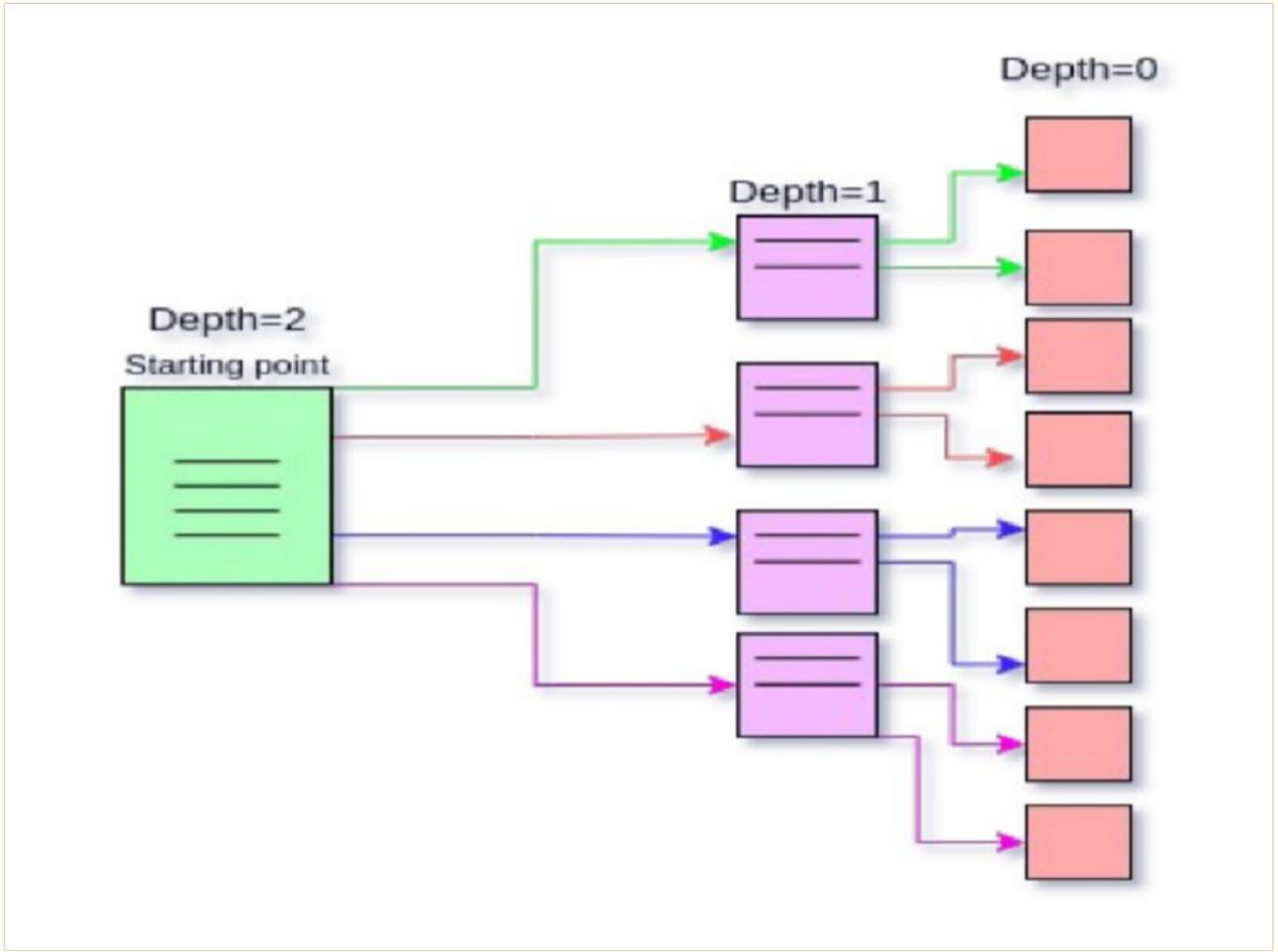
```
client = pymongo.MongoClient(connect_url_to_mongodb)

db = client.name_of_database
```

After the connection is made we are going to define two methods inside the class "Crawler" named start_crawling and crawl.

Both of the methods mentioned above are going to take two arguments:

url (string containing the url to the page we want to parse)
depth(integer parameter to control the number of pages your program crawls)


Depth=0
Depth=1
Depth=2
Starting point

RANKING MECHANISM

Once the search results are fetched from the database, next comes sorting them in order of relevance. To achieve this, first, the search query is separated into different keywords, after which, each search result is checked for the number of keywords present in it, and ranked according to it.

ALGORITHM

- Get the search query after the preprocessing and store them in an array keywords
- Get the search results from the database and store them in an object results.
- Check for the number of keywords present in each result in the object results.
- For each keyword in the title of the result, it is given score +2, and for each keyword in the description of the result, it is given score +1.
- Sort the results object in the descending order of score of each result.

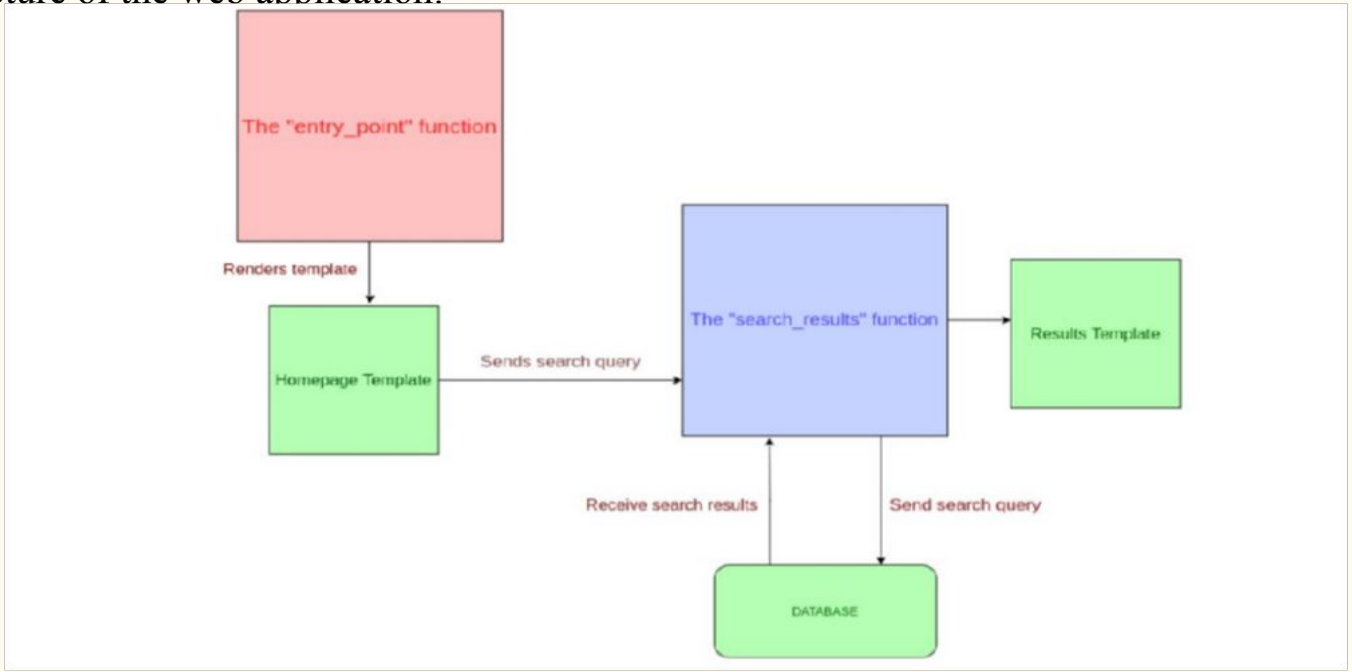
PageRank Algorithm

Page Rank algorithm is based on the assumption that more popular websites are to receive more links from other websites. You can apply the pagerank algorithm along with the algorithm mentioned above for effective search results. One way can be to add a field named pagerank_score in the object that you insert in the database, and while giving the search results sort the results in order of pagerank_score and score.

The Web App

We are going to make the web app using flask library in python. There will be two pages one the homepage and the other the search results page.

Basic structure of the web application:



Hardware and Software Requirement

The project required minimal processing capability.

Software

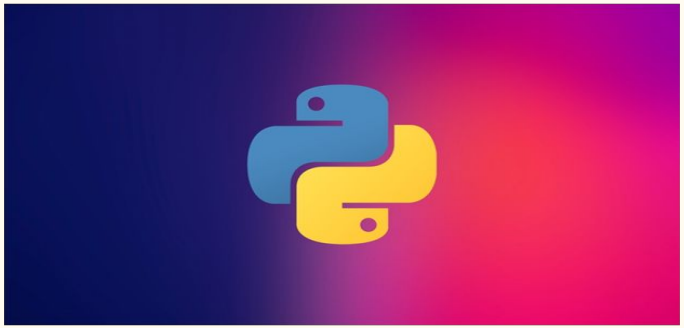
Python

Hardware:

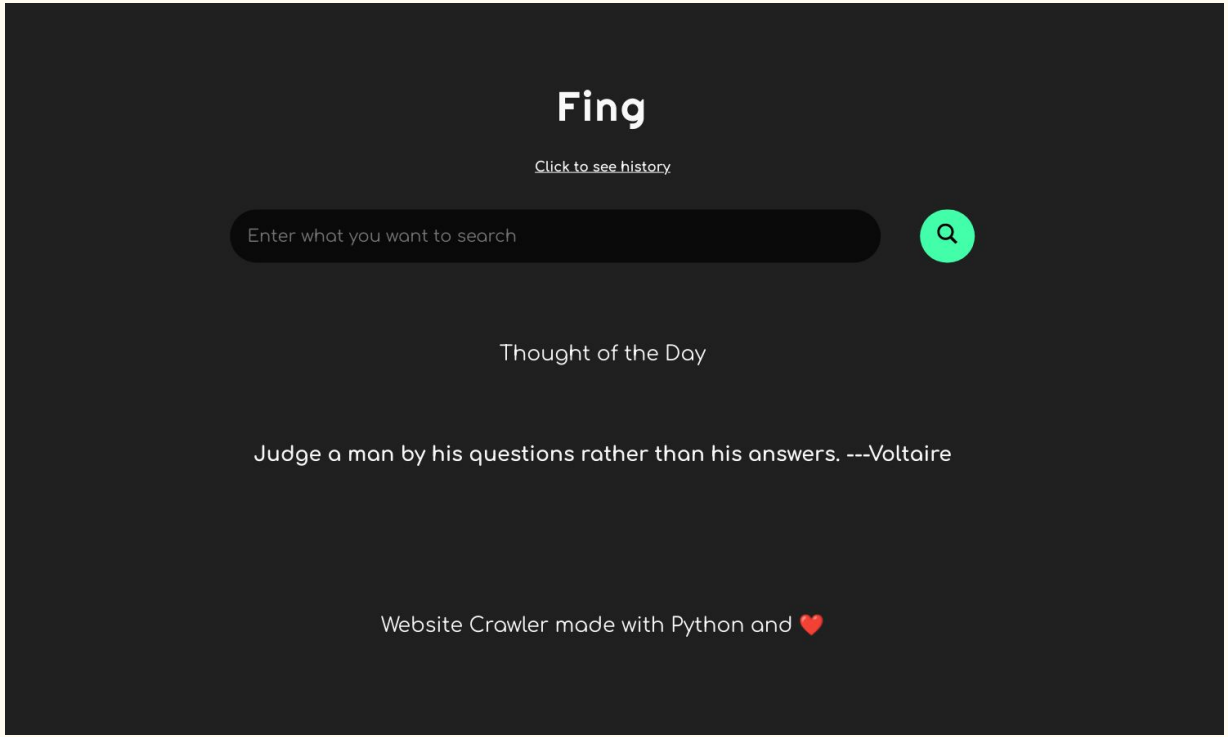
Any system with minimum Pentium processor and above

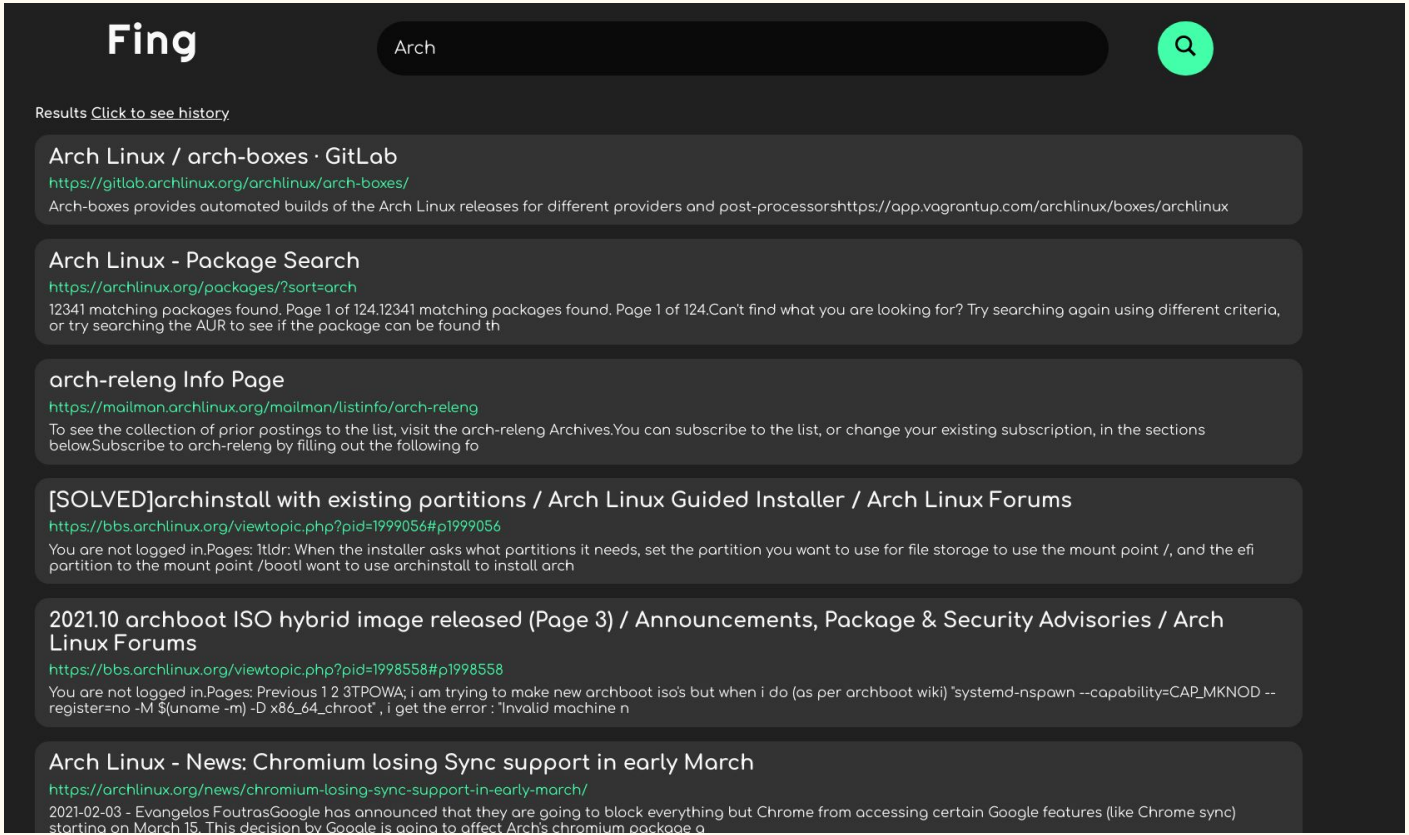
Technologies Used In Project

- Python
- Falsk
- Html
- Css
- MongoDB



EXPERIMENT AND RESULTS




Fing
Arch
Results Click to see history
Arch Linux / arch-boxes · GitLab
https://gitlab.archlinux.org/archlinux/arch-boxes/
Arch-boxes provides automated builds of the Arch Linux releases for different providers and post-processorshttps://app.vagrantup.com/archlinux/boxes/archlinux
Arch Linux - Package Search
https://archlinux.org/packages/?sort=arch
12341 matching packages found. Page 1 of 124.12341 matching packages found. Page 1 of 124.Can't find what you are looking for? Try searching again using different criteria, or try searching the AUR to see if the package can be found th
arch-releng Info Page
https://mailman.archlinux.org/mailman/listinfo/arch-releng
To see the collection of prior postings to the list, visit the arch-releng Archives.You can subscribe to the list, or change your existing subscription, in the sections below.Subscribe to arch-releng by filling out the following fo
[SOLVED]archinstall with existing partitions / Arch Linux Guided Installer / Arch Linux Forums
https://bbs.archlinux.org/viewtopic.php?pid=1999056#p1999056
You are not logged in.Pages: 1tldr: When the installer asks what partitions it needs, set the partition you want to use for file storage to use the mount point /, and the efi partition to the mount point /bootI want to use archinstall to install arch
2021.10 archboot ISO hybrid image released (Page 3) / Announcements, Package & Security Advisories / Arch Linux Forums
https://bbs.archlinux.org/viewtopic.php?pid=1998558#p1998558
You are not logged in.Pages: Previous 1 2 3TPOWA; i am trying to make new archboot iso's but when i do (as per archboot wiki) "systemd-nspawn --capability=CAP_MKNOD --register=no -M $(uname -m) -D x86_64_chroot" , i get the error : "Invalid machine n
Arch Linux - News: Chromium losing Sync support in early March
https://archlinux.org/news/chromium-losing-sync-support-in-early-march/
2021-02-03 - Evangelos FoutrasGoogle has announced that they are going to block everything but Chrome from accessing certain Google features (like Chrome sync) starting on March 15. This decision by Google is going to affect Arch's chromium package a

CONCLUSION & FUTURE WORK

Fing provide internet-based search services, providing accessibility to the world's online information.This project met all its original intended requirements and goals and was overall a great success .In Future work we will use ML and NLP and different indexing algorithm so user will more and more accurate result every time they searches also including voice assistance and many more features .

REFERENCES

- https://www.deepcrawl.com/knowledge/technical-seo-library/how-do-search- engines-work/
- https://flask.palletsprojects.com/en/2.0.x/
- https://en.wikipedia.org/wiki/PageRank
- https://pythonhosted.org/Flask-paginate/
- https://www.mongodb.com/blog/post/getting-started-with-python-and- mongodb
- https://en.wikipedia.org/wiki/PageRank
- http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf

Thank You!