SEARCH ENGINE

A PROJECT REPORT

Submitted by

HARINI M(2303811710422053)

In partial fulfillment of requirements for the award of the course

CGB1201 - JAVA PROGRAMMING

In

COMPUTER SCIENCE AND ENGINEERING

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE ,New Delhi)

SAMAYAPURAM–621112

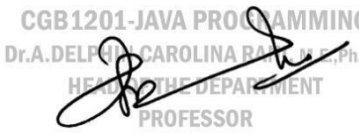
NOVEMBER-2024

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"SEARCH ENGINE"** is the bonafide work of **HARINI M (2303811710422053)** who carried out the project work during the academic year 2024-2025 under my supervision.



SIGNATURE

Dr.A.DelphinCarolinaRani,M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

PROFESSOR

Department of CSE

K.Ramakrishnan College of Technology(Autonomous)

Samayapuram–621112.

SIGNATURE

Mr. M. Saravanan,M.E.,

SUPERVISOR

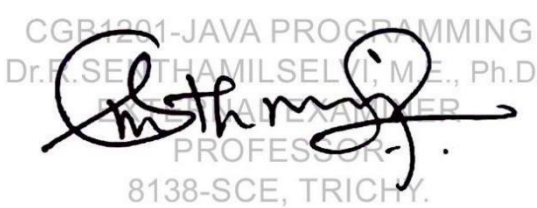
ASSISTANT PROFESSOR

Department of CSE

K.Ramakrishnan College of Technology(Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on 02.12.2024



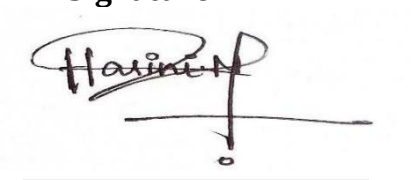
INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I declare that the project report on **"SEARCH ENGINE"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **CGB1201 - JAVA PROGRAMMING.**

Signature



HARINI M

Place:Samayapuram

Date:02.12.2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to our project and offering adequate duration in completing our project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr A. DELPHIN CAROLINA RANI, M.E.,Ph.D.,** Head of the department, **COMPUTER SCIENCE AND ENGINEERING** for providing her encourage pursuing this project.

I express our deep expression and sincere gratitude to our project guide **Mr. M. SARAVANAN, M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patience which motivated us to carry out this project.

I render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards

MISSION OF THE INSTITUTION

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of the industry and society.
- Be an institute with world class research facilities
- Be an institute nurturing talent and enhancing the competency of students to transform them as all-round personality respecting moral and ethical values

VISION OF DEPARTMENT

To be a center of eminence in creating competent software professionals with research and innovative skills.

MISSION OF DEPARTMENT

M1: Industry Specific: To nurture students in working with various hardware and software platforms inclined with the best practices of industry.

M2: Research: To prepare students for research-oriented activities.

M3: Society: To empower students with the required skills to solve complex technological problems of society.

PROGRAM EDUCATIONAL OBJECTIVES

1. PEO1: Domain Knowledge

To produce graduates who have strong foundation of knowledge and skills in the field of Computer Science and Engineering.

2. PEO2: Employability Skills and Research

To produce graduates who are employable in industries/public sector/research organizations or work as an entrepreneur.

3. **PEO3: Ethics and Values**

To develop leadership skills and ethically collaborate with society to tackle real-world challenges.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Domain Knowledge

To analyze, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2: Quality Software

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Innovation Ideas

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

A search engine is a complex software system designed to facilitate the retrieval of information from the internet by indexing vast amounts of web content and ranking it according to relevance. It works by crawling websites to gather data, which is then indexed in databases, allowing users to quickly access a wide range of information in response to specific queries. Search engines use sophisticated algorithms that analyze various factors, such as keywords, page content, backlinks, user behavior, and site authority, to deliver the most relevant results. In addition to traditional text-based results, modern search engines incorporate multimedia content, such as images, videos, and news, and support features like voice search and personalized recommendations. Search engines like Google, Bing, and Yahoo also employ advanced techniques such as natural language processing (NLP), artificial intelligence (AI), and machine learning to continuously refine results and improve user experience, offering faster and more accurate responses. As search technology evolves, search engines are increasingly incorporating deeper understanding of user intent and context, enhancing their ability to provide highly tailored and efficient results.

ABSTRACT WITH POs AND PSOs MAPPING

CO5:BUILD JAVA APPLICATIONS FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	Pos MAPPED	PSOs MAPPED
Building Java applications for solving real-time problems involves developing systems that can process, analyze, and respond to dynamic inputs instantly or within minimal delay.These applications can be applied across various domains such as finance, healthcare, e-commerce, and search engines. For example, in a real-time stock trading application, Java can be used to monitor and analyze live market data, executing trades automatically based on predefined strategies, while ensuring high availability and scalability. In real-time data processing systems**,** Java's powerful concurrency libraries and frame-works like Apache Kafka or Apache Flink can be employed to process large streams of data efficiently, enabling immediate insights.	**PO1-3** **PO2-3** **PO3-3** **PO4-3** **PO5-3** **PO6-3** **PO7-3** **PO8-3** **PO9-3** **PO10 -** **3PO11-** **3PO12-** **3**	**PSO1-3** **PSO2-3** **PSO3-3**

Note:1-Low,2-Medium,3-High

TABLE OF CONTENTS

CHAPTERNO.	TITLE	PAGENO.
	ABSTRACT	x
1	**INTRODUCTION**	2
	1.1 Objective	2
	1.2 Overview	2
	1.3 Java Programming concepts	3
2	**PROJECT METHODOLOGY**	4
	2.1 Proposed Work	4
	2.2 Block Diagram	4
3	**MODULE DESCRIPTION**	5
	3.1 CRAWLER(WEBSCRAPER)	5
	3.2 INDEXER	5
	3.3 QUERY PROCESSOR	5
	3.4 RANKING ALGORITHM	5
4	**CONCLUSION AND FUTURE SCOPE**	6
	4.1 Conclusion	6
	4.2 Future Scope	6
	REFERENCES	7
	APPENDIX A(SOURCECODE)	8
	APPENDIX B(SCREENSHOTS)	11

1.1 Objective

C

H

A

P

T

E

R

1

I

N

T

R

O

D

U

C

T

I

O

N

The objective of a search engine is to deliver fast, accurate, and relevant results to users by efficiently indexing vast amounts of web content and understanding the intent behind queries. Itaims to provide personalized results based on user preferences, search history, and context,while continuously improving its algorithms to enhance result ranking and relevance using AI and machine learning. The engine must be scalable, handling increasing volumes of data with-out sacrificing performance, and prioritize privacy and security by safeguarding user information. Ensuring fairness and reducing bias in search results is crucial, along with providing equal representation of diverse content. Additionally, the search engine seeks to integrate advanced features like multimedia search, voice search, and real-time crawling, offering a seam-less and user-friendly experience across all devices. It also focuses on optimizing crawling strategies to index the most up-to-date content from across the web efficiently. Collaboration with various data sources helps improve the comprehensiveness of search results, ensuring a wider range of content is accessible.Ultimately,the search engine strives to improve user satisfaction by continuously refining its performance and providing relevant,timely results.

- **T1.2 Overview**

A search engine is a software system designed to search for information on the World Wide Web or within a specific database, returning relevant results based on user queries. It operates by using web crawlers to explore and index vast amounts of web content, storing this information in a structured database that allows for fast retrieval when a user enters a query.The search engine uses complex algorithms and ranking systems to analyze and sort results by relevance, taking into account factors such as keyword matching, link quality, and con-tent authority. Advanced machine learning and AI technologies are employed to improve result relevance, adapt to user behaviour, and personalize experiences based on user preferences, location, and search history. Features such as multimedia search (images, videos, audio),voice recognition,and real-time updates further enhance the user experience by providing diverse and up-to-date information. Natural Language Processing (NLP) helps the engine understand complex queries, including conversational or long-tail questions, while query expansion techniques improve result accuracy by interpreting synonyms and related terms. Search engines also integrate local search capabilities, ensuring that users receive geographically relevant results. The system is built to be scalable, handling large amounts of data and user traffic without sacrificing performance. Ensuring privacy, security, and fairness in results is critical, with mechanisms to protect user data and a void bias in ranking.

Continuous updates to crawling algorithms,indexing strategies, and content filtering ensure that results

remain fresh, relevant, and free from spam or low-quality sources. Furthermore,user feedback and engagement metrics are often used to refine ranking models and improve overall search quality. Ultimately, the goal of a search engine is to deliver fast, accurate,and relevant content while providing a seamless and efficient experience for users to find the information they need.

Java Programming Concepts

The basic concept of Object-Oriented Programming (OOP) in the context of a search engine involves applying OOP principles—Encapsulation, Abstraction, Inheritance, and Polymorphism—to structure the code in a modular, maintainable,and scalable way.

Encapsulation: In a search engine, encapsulation is used to bundle related data and functions together. For example, a Search Engine class could encapsulate methods for quartering index, ranking results, and displaying them, while internal details like indexing and crawling are hidden from the user or other parts of the system.

Abstraction: Abstraction in search engines hides complex underlying processes from the user or other components. For instance, when a user submits a query, they only interact with a high-level search interface, which abstracts away the complexities of how the search engine crawls the web,indexes content,and ranks results.

Inheritance: Inheritance allows search engine components to be reused and extended. For example, abase Crawler class can be extended by specialized classes such as WebCrawler,Image Crawler, or Video Crawler that inherit the basic crawling functionality but add specific behaviour for handling different types of content.

Polymorphism: Polymorphism enables the search engine to handle different types of queries in a flexible manner. A common interface for query processing can process various types of queries(text,voice,image)by using polymorphic methods,where the implementa-tionvariesdependingonthequerytypebutfollowsthesameinterface.

In Object-Oriented Programming (OOP), class and object are core concepts that help structure and organize code.

Class: A class is like a blueprint or template for creating objects. It defines the properties(attributes) and behaviours (methods) that the objects created from it will have. A class out-lines what data an object will store and what actions it can perform, but it does not contain specific data itself. Think of it as a general definition, like a "Search Engine" class that specifiesit can crawl the web,store indexes,and rank search results.

Object: An object is an instance of a class. It is a specific realization of the class, with actual data and the ability to perform the behaviours defined in the class. For example, when you create a search engine object (like "Google" or "Bing"), you're creating an object from the "Search Engine" class. This object will have real data, such as the list of indexed pages or the search query it processes,and it can perform actions like crawling newebsites.

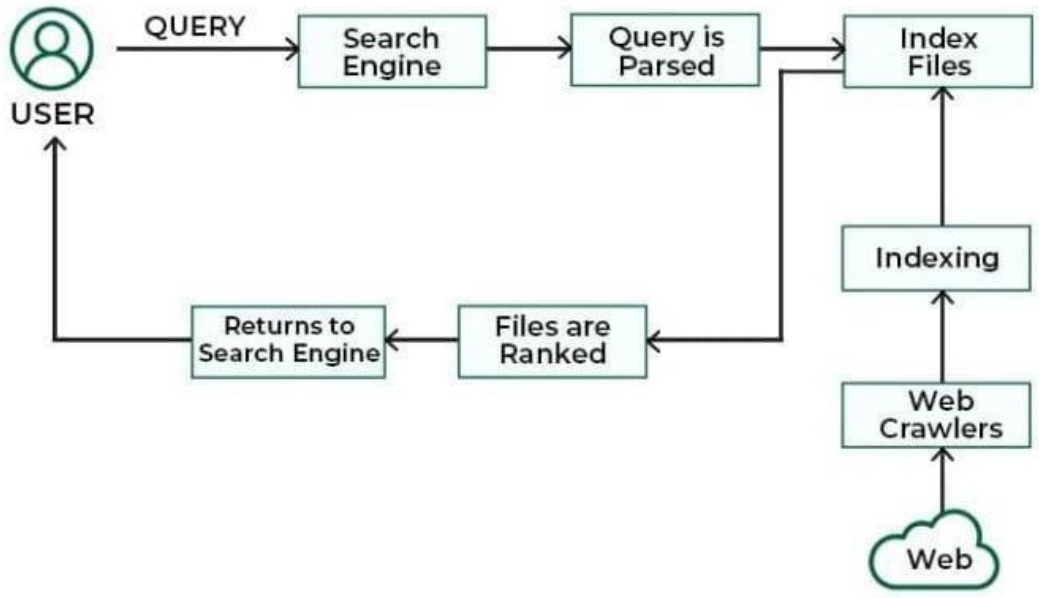
CHAPTER 2
PROJECT METHODOLOGY

2.1 Proposed Work

The proposed work in search engine development aims to enhance accuracy and user experience through AI and machine learning for better ranking and personalized results.Optimizing crawling and indexing processes will ensure faster,more comprehensive access to up-to-date content.

Integration of Natural Language Processing (NLP) will allow the engine to understand complex queries and user intent.Improved multimedia search capabilities ,such as image, video,and voice search, will meet growing user demands. Efforts will be made to ensure privacy, security, and fairness by addressing biases and protecting user data. Real-time updates and scalability will be prioritized to handle increasing data and traffic. Localization and contextual search will provide more relevant, location-aware results. Ultimately, the goal is to create a more efficient,personalized,and intelligent searchengine.

2.2 Block Diagram



CHAPTER 3

MODULE DESCRIPTION

3.1 Crawler(Web Scraper)

A crawler (or web scraper) is a program designed to automatically browse the web and collect data from websites. It works by navigating through web pages, following links, and downloading content, which is then stored for indexing or analysis. Crawlers help search engines gather information to build comprehensive indexes of the web. They can also be used for data extraction, monitoring website changes, or gathering large volumes of structured data for analysis.

3.2 Indexer

An indexer is a component of a search engine responsible for organizing and storing the data collected by a crawler. It processes the raw content from web pages, extracting key information such as keywords, meta data, and links. This information is then stored in an index, allowing for fast retrieval when a user submit a search query.The indexer helps optimize search performance by creating an efficient structure that enables quick searches and relevant result rankings.

3.3 Queryprocessor

A query processor is a key component of a search engine that interprets and processes user search queries. It analyzes the input to understand the intent behind the query, often applying techniques like tokenization, stemming, and synonym matching to refine the search. The query processor then matches the processed query against the indexed data to retrieve relevant results. It also ranks these results based on factors such as relevance, popularity, and user context, providing the most accurate answers to the user

3.4 Ranking Algorithm

A ranking algorithm in a search engine determines the order in which search results are displayed to the user.It evaluates various factors such as relevance of the content,quality of the web page,and authority(e.g.,back links or domain reputation).The algorithm assigns scores to the results based on these factors, with higher-scoring pages appearing first.

CHAPTER4
CONCLUSION & FUTURE SCOPE

4.1 CONCLUSION

In conclusion, the search engine effectively serves its primary purpose of indexing and retrieving relevant information based on user queries. Through its use of advanced algorithms and ranking techniques, the system is able to return accurate and timely results, offering users an efficient search experience. While the search engine performs well in terms of speed and relevance, there is always room for enhancement. Future developments could focus on improving query understanding, incorporating AI-driven personalization, and expanding its database to accommodate evolving user needs. Overall, the search engine proves to be a reliable tool, with the potential for further growth and optimization in response to emerging technologies and user expectations. This conclusion highlights the strengths, acknowledges areas for growth, and suggests next steps for enhancing the search engine.

4.2 FUTURESCOPE

The future scope of search engines is poised for significant evolution, driven by advancements in AI, machine learning, and natural language processing, which will enable smarter, more personalized, and context-aware search results. As voice search and conversational interfaces gain popularity, search engines will adapt to handle spoken queries and provide interactive, conversational experiences. Visual search and augmented reality will further enhance user engagement by enabling image based queries and immersive, real-time interactions. Additionally, semantic search and improved knowledge graphs will offer more accurate, intent-based results, while increased focus on privacy and data security will lead to more secure, decentralized search options. With the rise of niche vertical search engines, real-time search capabilities, and integration with IoT devices and other digital platforms, search engines will become even more interconnected and versatile, offering tailored, up-to-the-minute, and cross-platform experiences. As these innovations unfold, the search experience will continue to evolve into a smarter, more ethical, and immersive tool for users across a variety of contexts.

REFERENCES

JAVABOOKS

1. **"Lucene in Action"by Erik Hatcher and Otis Gospodnetic**
2. **Description**: This book provides a comprehensive guide to Apache Lucene, a powerful Java library for indexing and searching text. It covers the fundamentals of searching in development,from indexing data to building search functionality. Lucene is widely used in the industry, and this book offers practical insights into using it for creating searching in Java.
3. **Link**:LuceneinAction

4.3 "Mastering Elastic search"by Rafał Kućand Marek Rogoziński

1. **Description**: Elastic search, built on top of Apache Lucene, is a popular open source search engine widely used for full-text search and real-time analytics. This book covers the setup, configuration, and optimization of Elastic search, which is useful when building robust search engine applications usingJava.
2. **Link**:MasteringElasticsearch

WEBSITESLINK

1. **TutorialsPoint**

   https://www.tutorialspoint.com/

   TutorialsPoint offers a vast collection of tutorials on programming languages, data structures,algorithms,web development,and other technical topics.It provides simple explanations,code examples,and interactive tutorials to help learner satall levels.
2. **W3Schools**

   https://www.w3schools.com/

   W3Schools is a popular website for learning web development technologies like HTML,CSS,JavaScript,SQL,and more .It provide easy to understand tutorials,interactive examples,and quizzes to help users practice coding directly in their browser.
3. **LeetCode**

   https://leetcode.com/

   LeetCode is a platform for practicing coding problems, especially for interviews and algorithmic challenges. It offers a wide range of problems that cover data structures, algorithm

APPENDIX A

(SOURCECODE)

```
importjava.awt.*;import
java.awt.event.*;importj
ava.util.*;

publicclassSimpleSearchQueryApp{
 //Predefineddataforsimulation
private static final String[] sampleData
="Javaisaprogramminglanguage.","Java
AWTisa GUItoolkit.",
"Javaprogramming isfun.",
"AWTstandsforAbstractWindowToolkit.","S
earchengineshelpyou findinformation."
};
// Main GUI and Application
   Logicpublicstaticvoid
   main(String[]args){
      Frame frame=new
      Frame("SimpleSearchQueryApplication");frame.setSize(500,400
      );
      frame.setLayout(newFlowLayout());
      //LabelandTextFieldforuserinput
      LabelsearchLabel=newLabel("EnterSearchQuery:");TextField
      searchField= newTextField(30);

      //Buttontotrigger thesearch
      ButtonsearchButton=newButton("Search");
      //TextAreatodisplaysearchresults
      TextArea resultArea = new TextArea(10,
      40);resultArea.setEditable(false);
      //Addcomponentstoframefr
```

```
ame.add(searchLabel);fram
e.add(searchField);frame.ad
d(searchButton);frame.add(
resultArea);
//Searchbuttonactionlistener
searchButton.addActionListener(new ActionListener()
   {@Override
   publicvoidactionPerformed(ActionEvent e){
      Stringquery=searchField.getText().trim().toLowerCase();res
      ultArea.setText("");// Clearpreviousresults
      if(query.isEmpty()) {
         resultArea.append("Pleaseenterasearchquery.\n");
      }else{
         booleanfound= false;
         // Search through the predefined
         datafor(Stringdata :sampleData){
            if (data.toLowerCase().contains(query))
               {resultArea.append("Found: " + data +
               "\n");found=true;
            }
         }

          if(!found){
               resultArea.append("Noresultsfoundfor query:"+query+"\n");
         }
      }
   }
});
//Setupwindow
closingbehaviorframe.addWindowListener(newWin
```

```
    dowAdapter(){
        publicvoidwindowClosing(WindowEventwe){System.exit(0);
        }
    });
    //Makethe
    framevisibleframe.setVi
    sible(true);
  }
}
```

APPENDIX B

(SCREENSHOT)