

## **Project 2: Prototype IR System Report**



**By**

Thanachot Onlamoon 6588062

Punnut Sawetwannakul 6588142

Bhurinat Kanchanasuwan 6588150

This project is partial fulfillment of the requirements for  
ITCS414 Information storage and retrieval  
Academic year 2024/ 1st semester  
Faculty of ICT, Mahidol university

## **Table of content**

<b>Topic</b>	<b>Page no.</b>
Introduction	3
Problems that we are trying to solve	4
Existing relevant systems	4
Implementation	5
Discussion	9
Conclusion	11

## **Introduction**

In this report, we present the development of a prototype information retrieval (IR) system focused on enhancing search functionality for information related specifically to Doraemon gadgets. Targeting both Doraemon enthusiasts and toy designers, this system aims to provide a streamlined solution for accessing detailed data on specific gadgets, which is currently challenging due to limited specialized search options. Using web scraping and indexing techniques with tools like GoColly, Elasticsearch, and Kibana, our team built an interface to support intuitive and efficient queries, offering a customized experience for users interested in this unique niche. This report outlines our process, from identifying the problem and analyzing existing systems to implementation and discussion of technical challenges, lessons learned, and system limitations.



## Problems that we are trying to solve

Problem that we are trying to solve is:

There are:

- Many doraemon fans and doraemon merchandise collectors who want to know more information that are related to some doraemon gadgets.
- Many toy designers and toy creators who also want to know more information that are related to some doraemon gadgets to design and create these gadgets in real life.

But they don't have a decent search system which can search for information that are related to doraemon gadgets only.



### Existing relevant systems

There are some existing relevant systems which can search for information that are related to doraemon but there are 0 existing relevant systems which can search for information that are related to doraemon gadgets only.

These are the main existing relevant systems that are mentioned above.

- [Doraemon Wiki](#)
- [Doraemon.com](#)

## **Implementation**

### **Data collection**

Our group uses “web scraping” as our data collection method. Web scraping is a type of data collection method that uses softwares to collect data such as texts, pictures and much more from a certain website.

Our group uses [Doraemon Wiki](#) as our source website.



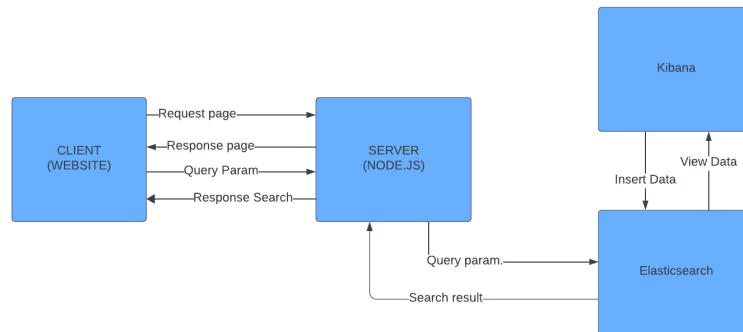
### **Software and Tool**

These are softwares and tools that our group uses:

- Data collection: our group uses “Go-Colly” to collect data.  
Go-Colly is a type of web scraping tool which is written in Go programming language and can automate a process of collecting data such as texts, pictures and much more from a website.
- Data indexing: our group uses “Elasticsearch (Okapi BM25)” and “Kibana” to index data.  
Elasticsearch (Okapi BM25) is a type of search engine which is a part of an elastic stack and can search a certain data from a large database/ store a certain data into a large database.  
Kibana is a type of data visualization tool that is also a part of an elastic stack and can view a large database which is stored in Elasticsearch through a graph or a dashboard.

- Search system interface designing: our group uses “Node.js”, “Express.js”, “HTML”, “CSS”, and “Javascript” to design our search system interface.

## System diagram



## System snapshots

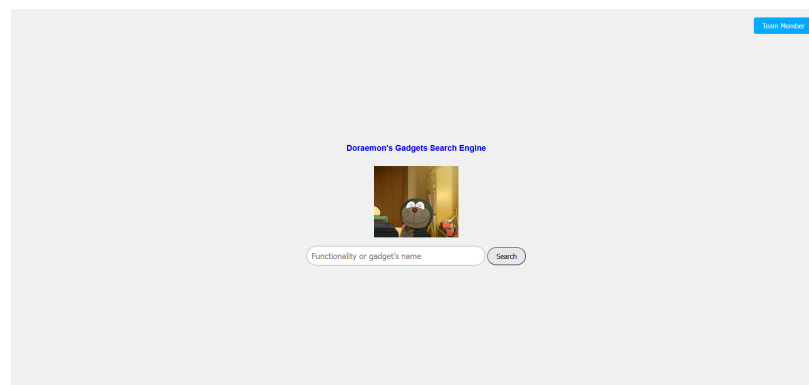
Link to the system: <https://doraemon.mahumeaw.com/>

Link to GitHub Repository:

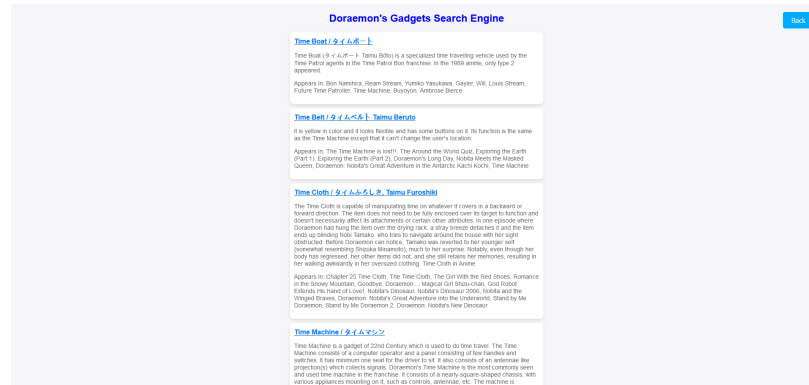
<https://github.com/Yobubble/doraemon-gadgets-search-engine.git>

These are our search system interface:

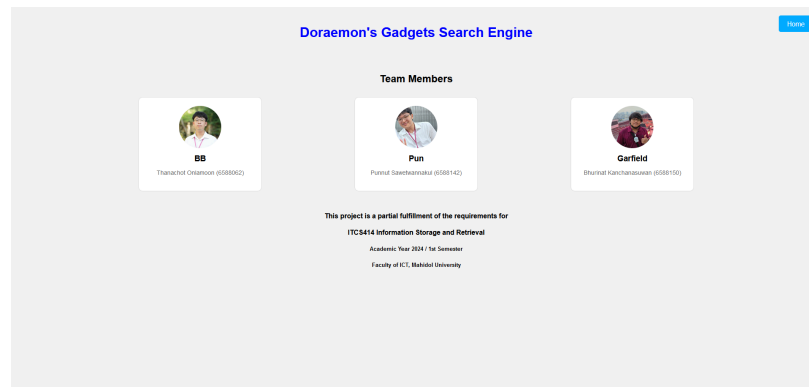
- Search page: this page is a page where users can search for a certain information by typing a query.



- Search results page: this page is a page where users can see a certain information from their typed query.



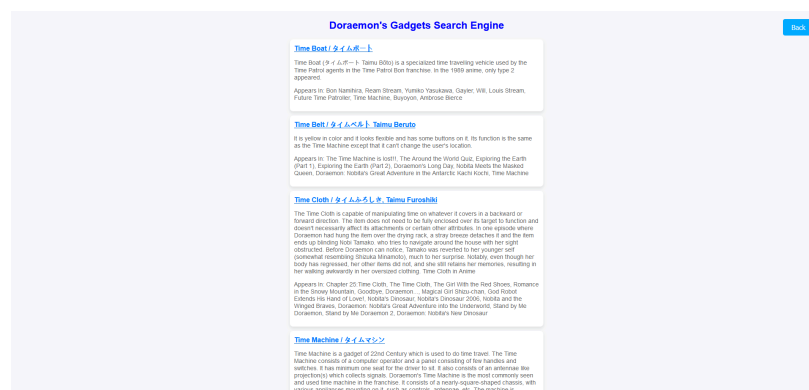
- Our group member page: this page is a page where users can see our group member information.



## System snapshots: query and rank

This is our search results page when:

- Users type “time” (an example of 1 word query) in our search page.







## **Discussion**

### **Technical difficulties/ Challenges and How to handle these technical difficulties/ challenges**

These are technical difficulties/ challenges that we face in this project:

- Data Scraping Issues
  - Unspecified HTML elements complicate web scraping since targeted classes or IDs are missing.
  - Data inconsistency, as Doraemon gadgets on the website have varying properties.
- Image URL Access Issues:
  - Images from the fandom site are restricted by “Content Security Policy” (CSP) and hotlinking blocks, preventing direct embedding.
- Elasticsearch Incompatibility:
  - Some team members cannot run Elasticsearch via Docker Compose for development.

How we handle these technical difficulties/ challenges that we face in this project:

- Data Scraping Issues:
  - Used advanced GO-Colly queries (e.g., `ul > li > a`) to target necessary elements.
  - Filtered out gadgets missing essential properties, reducing data from 700 to around 300 gadgets.
- Image URLs:
  - Switched to a different image source and re-scraped with the new links.
- Elasticsearch Mocking:
  - Created a mock API endpoint for testing, allowing frontend work without Elasticsearch.

## **Learned lessons**

These are learned lessons which our group learns from this project:

Our group learns:

- What is “web scraping” and how to use “GoColly” to do web scraping properly.
- What are “Elasticsearch (Okapi BM25)” and “Kibana” and how to use “Elasticsearch (Okapi BM25)” and “Kibana” to index data properly.
- How to use “Node.js”, “Express.js”, “HTML”, “CSS”, and “Javascript” to create a search system interface properly.
- What are “1 word query”, “multiple word query”, “partial match query”, and “ranking”.
- How to work as a team effectively.
- How to solve problems which occur during this project effectively.

## **System limitations**

These are our search system limitations:

- In our search results page, users can’t see some gadgets. This happens because our source website doesn’t have any information about some gadgets, so our group decided not to put these gadgets into our search system.
- In our search results page, users can’t see gadget’s data completely (all gadgets don’t have some data). This happens because our source website doesn’t have some information about some gadgets.
- In our search results page, users can’t see gadget’s pictures. This happens because our source website doesn’t allow others to collect gadget’s picture.

## **System limitations: future improvements**

About future improvements of our search system limitations, our group will use different source websites to make sure that users will see all gadgets (each gadget will have its complete data and its picture) in our search results page.

## **Conclusion**

In conclusion, this project successfully demonstrates the development of a prototype information retrieval system tailored to providing precise information on Doraemon gadgets. By employing web scraping with GoColly and indexing through Elasticsearch, the system offers an effective solution for Doraemon fans and toy designers seeking detailed gadget information. Despite challenges with data inconsistencies and image accessibility, our team implemented workarounds and learned valuable lessons in web scraping, database indexing, and frontend development. While current limitations remain, such as incomplete gadget data, future improvements, including alternative data sources and enhanced scraping tools, can further refine the system's capabilities and user experience.

