University of South-Eastern Norway
Faculty of Science, Technology, Natural Sciences and Maritime Sciences
–

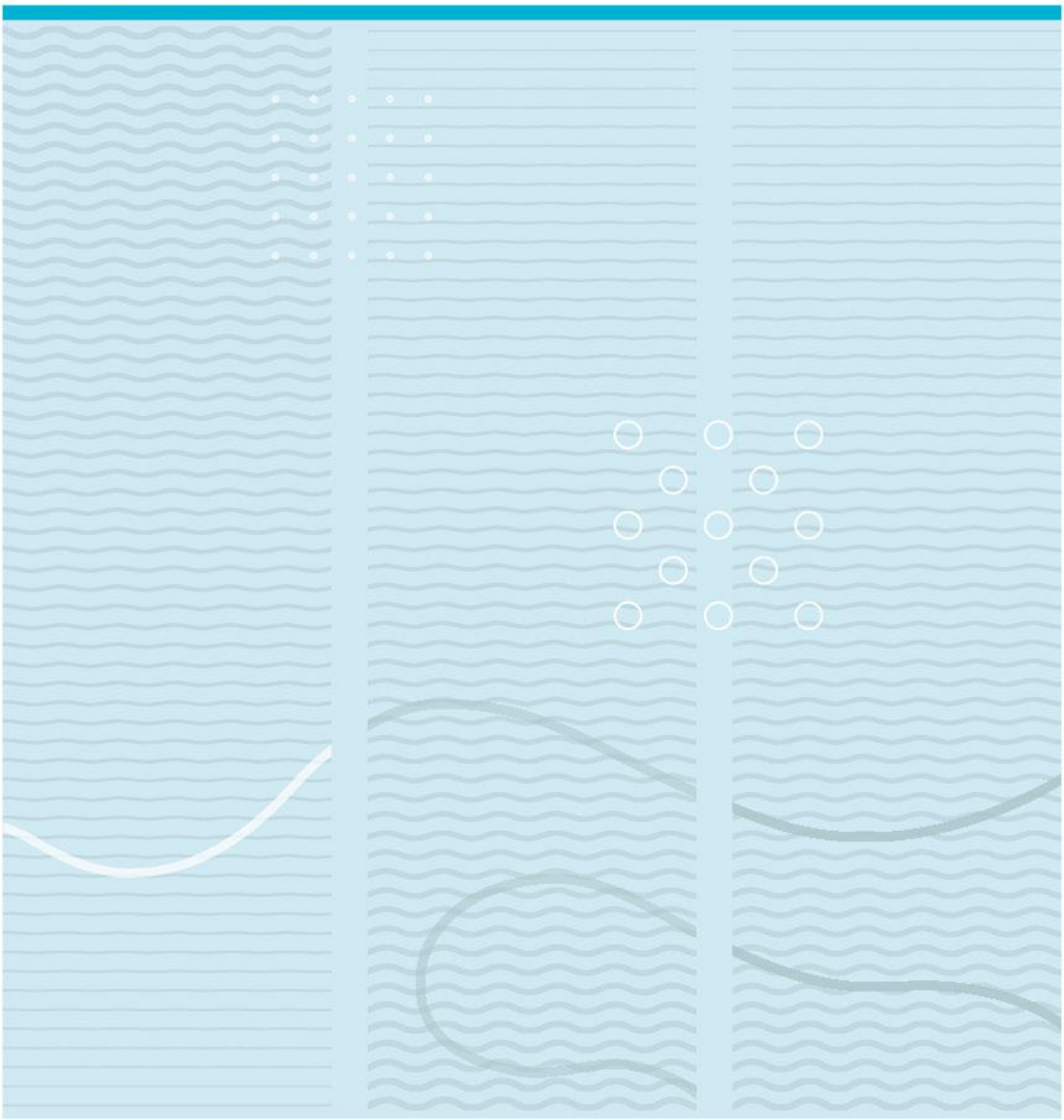
CS4010: Data Management

Study programme: Master of Science in Computer Science

Spring/Autumn 2021

246108
250822

Elastic Search Integration



University of South-Eastern Norway
Faculty of Science, Technology, Natural Sciences and Maritime Sciences
PO Box 235
NO-3603 Kongsberg, Norway

http://www.usn.no

This project is worth xx study points

Summary/Abstract

The requirement of organization (ThrowNo More) was to add search function in their existing application, which we carried out as our project in the Data Management course. To perform this, we chose elasticsearch because the requirement was to search the information by text (product name) over huge amount of data quickly. We received mysql dump file as data input to our project. Then, we created a database and generated tables inside that database. After that, we imported the tables into elasticsearch and created index there. Finally, we created a search interface with the help of appsearch, that produced satisfied result over the search query.

We could also host elasticsearch in our own server and develop the similar search engine there Which we want to do as our future work.

Contents

1 Introduction 5
1.1 Problem statement 5
1.2 Project objective 5
1.3 Assumptions and limitations 6
1.4 Project management 6
2 Theoretical background and Literature review 8
2.1 Theoretical background 8
2.2 Literature review 9
3 Research methodology 10
4 Results 11
5 Discussion and conclusions 12
5.1 Discussion 12
5.2 Conclusions 12

1 Introduction

With the development of applications, the daily task of people has become easier. We use search feature of applications, almost in our all-daily activities. For example, searching for transport, searching goods on e-commerce sites, checking weather forecast, searching for shops and offer, etc. Every time people are searching something and something. So, the search feature has become one of the basic requirements in any web applications and essential to integrate within.

Elasticsearch is a search engine which provides quick full-text search over huge amount of unstructured data (Han & Zhu, 2020). The elasticsearch is not only database, but also a competent search engine which is horizontally scalable, distributed and high fault-tolerant. With increase of business data , it is possible to add more servers and the components can be scattered over various servers in the network. The advantage of distribution is that if the system fails in one server, it will not affect the system running on other server (Divya & Goyal, 2013).

1.1 Problem statement

The project is proposed by an organization, Throwno More which is working to reduce food wastages in shops through an application named, “Throw No More” which provides an overview of goods at reduced price, because of close expiry date or package leakage. To use the app, the user should input his/her current location and the app will display all the shops nearby having offer goods. Then the user can select any shop, go inside and check the goods at discounted price.

The problem is that the user cannot search directly by product name he/she requires, they can only check shop-wise.

1.2 Project objective

The objective of our project is to add search function to the existing application so that the users can search directly by the product, they want and have better search experience.

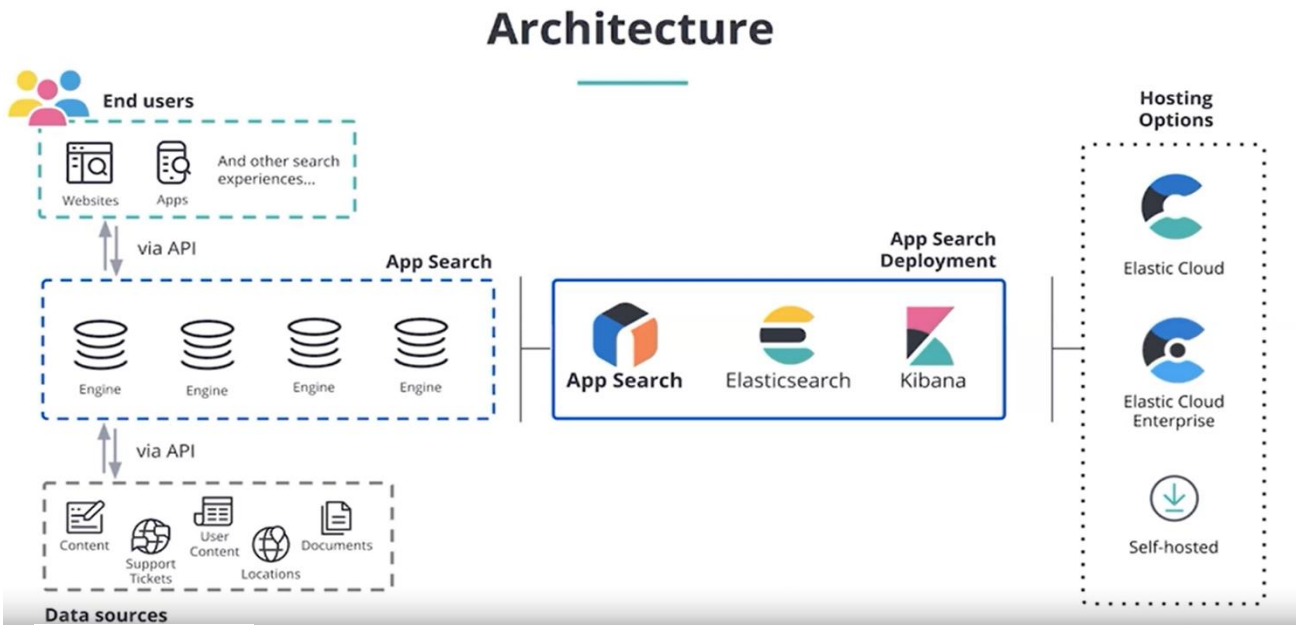
1.3 Assumptions and limitations

Our final goal was to install elastic search in Linux environment and integrate our search engine into Throw no more apps. Unfortunately, our two group members could not continue this course because of late arrival. So, we decided to continue our work in elastic cloud to complete our project before the deadline. There was data size limitation and few restrictions in free version of elastic cloud.

1.4 Project management

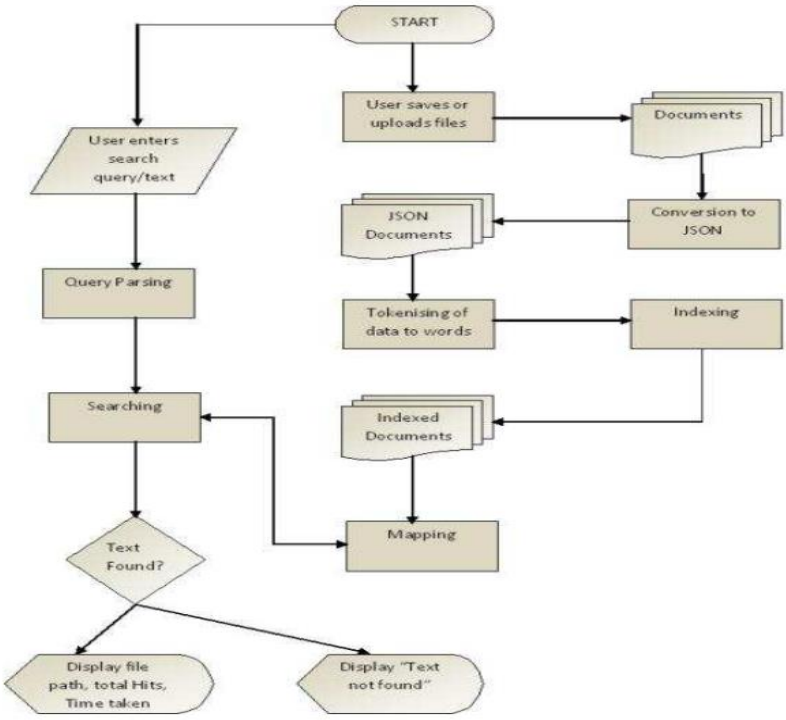
We two people work equally to complete the project. We distributed our work as mentioned below:

- Communication with organization (ThrowNo More)- Nirajan Karki
- Software installation- Both
- Creating Database, Data import- Md. Ohidul Islam
- Search operation and output generation- Both
- Assumption & Limitation, Research Methodology, Result - Md. Ohidul Islam
- Report writing – Nirajan Karki



Source: elastic.co

Figure 1- Elasticsearch stack architecture



Source: (Divya & Goyal, 2013)

Figure 2- Elasticsearch basic workflow

2 Theoretical background and Literature review

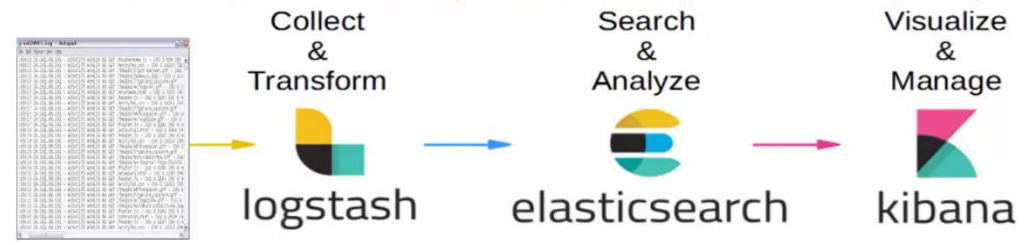
2.1 Theoretical background

Elasticsearch ELK stack composed of three open-source projects:

E – Elasticsearch, is a search and analytics engine

L – Logstash, is a server-side data processing pipeline for ingesting data from multiple sources into the elasticsearch

K- Kibana, is a browser-based tool for visualization of data



Source: ProgrammingKnowledge

Figure 3- ELK stack diagram

2.1.1 Elasticsearch

Elasticsearch is the core component of ELK stack based on Apache Lucene. It is an open-source project developed in Java which supports full-text search that is completely based on documents rather than tables and schemas. So, it can perform and combine many types of searches like structured, unstructured, geo, metrics, etc.

Normally, ES is used for single page application projects.

Terminology and the concept of elasticsearch can be better understood by comparing with the relational database system.

Node - Server

Index - Database

Type – Table

Document/Id – Record

Here, a document is a basic unit of information in JSON format which can be indexed. Many such documents having common fields can be stored together in a type. An index is a collection of types; we can define more than one type in our index. A node is a single server containing many indexes. On the top of index, there comes a cluster which is a collection of one or more nodes that together holds entire data.
There are also two more terms in elasticsearch, shards and replicas. ES can subdivide the index into multiple pieces called shards. Each shard is in itself fully-functional index that can be hosted in any node. ES allows us to make the copy of shard's index which is known as replicas(Han & Zhu, 2020).

2.2 Literature review

The history of development of elasticsearch is very interesting. It started from making of search engine for growing list of receipes by Shay Banon for his wife who was attending cooking school in UK. He created a Compass as his first iteration in 2004. During release of third version of the Compass, he thought to rewrite the big parts of Compass to create a scalable search solution. So, he made a solution from the ground up to be distributed using JSON which provides the flexibility for programming in different languages other than Java as well. In this way, he released the first version of Elasticsearch in 2010 (Divya & Goyal, 2013).

3 Research methodology

At first, we need to have ELK software's on our local PC to perform search operations. We installed all of these in windows as well as in Linux environment. We used 'Ubuntu' open-source operating system of Linux for our project. The best thing about elasticsearch is its extensive Rest API. Indexed data can be integrated, managed and query in different ways by using these API. Then we need data to run into these systems. In case of our project, we need not research for our data, as we received Mysql data directly from the organization. We used php admin to create the database as well as export data in the required format (JSON).

Figure 4- Screenshot of table

Then we look for hosting environment for elasticsearch (ELK stack) whether we should install it in our local computers or use elastic cloud. Due to time and manpower limitation, we choose elastic cloud. To install and be familiar with these software, we read different software documentation and watched youtube videos.

4 Results

For testing our project, we performed search query by product name , so entered the name of some products name as input. For example, we searched yoghurt and it displayed all kinds of yoghurts which are on the sale at our nearby shops as shown in the result figure-1,

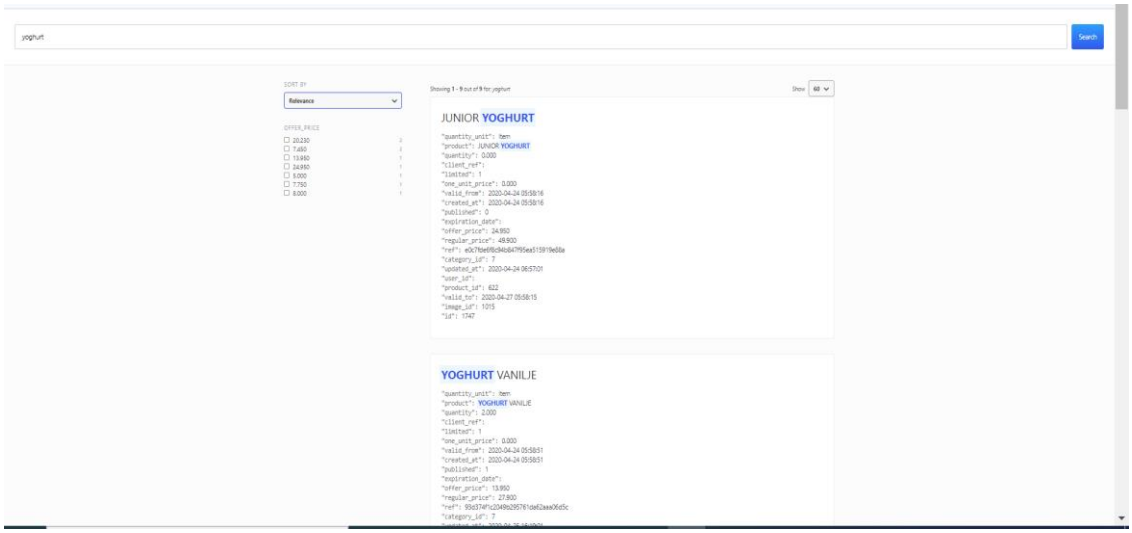


Figure 5 – output 1

Our engine is also able to perform different type of sorts and filters such as products can be sorts by expiry date.

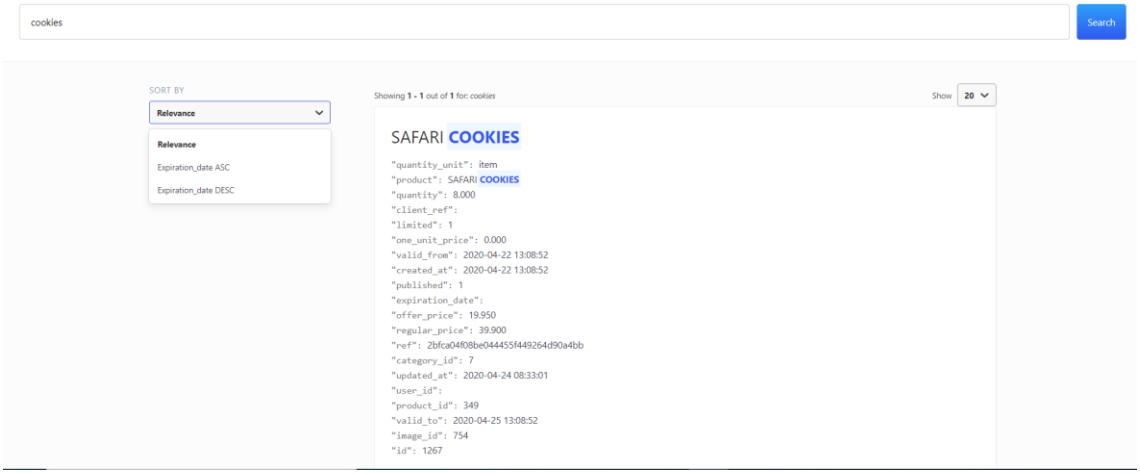


Figure 6 – output 2

5 Discussion and conclusions

5.1 Discussion

Results received from the testing verified and met our expectation from the project. The results also met the objective of the project to produce quick full-text search result over voluminous data. However, we can also customize the way to present our result by using filter, sorting, relevance, tuning, etc as per requirement.

5.2 Conclusions

We are able to develop a search UI(User Interface) using elasticsearch that can search the information by product which was also the main requirement of organization. We used mysql for initially storing our data and imported it to the elasticseach and used app search to create an UI in elastic cloud hosting.

It is also possible to host all ELK components (elasticsearch, logstash, kibana) in our own server and create a search UI there, which we really wanted to do if we had more time.

List of figures

Figure 1- Architecture of elastic stack

Figure 2- Elasticsearch basic workflow

Figure 3- Elastic stack diagram

Figure 4- Screenshot of table

Figure 5- Output 1

Figure 6- Output 2

References/bibliography

Chow, N., & Tetrault, D. (2020). *Webinar on getting started with app search.* Retrieved from: https://www.elastic.co/webinars/getting-started-with-elastic-app-search

Divya, M., & Goyal, S. (2013). An advanced and quick search technique to handle voluminous data.

Han, L., & Zhu, L. (2020). *Design and Implementation of Elasticsearch for Media Data.* Paper presented at the 2020 International Conference on Computer Engineering and Application (ICCEA).

https://logz.io/learn/complete-guide-elk-stack/#elasticsearch

ProgrammingKnowledge. (2019). *How to Install ElasticSearch Logstash and Kibana on Windows 10 (ELK Stack) (Elastic Stack)*. Retrieved from: https://www.youtube.com/watch?v=8iXZTS7f_hY&t=1s