BAB II
DASAR TEORI

2.1 Elastic Stack

Elastic Stack merupakan aplikasi *open source* yang berguna untuk melakukan pengindeksan data dan memvisualisasikannya menjadi sebuah grafik, *metric*, tabel, ataupun gambar. Elastic Stack terdiri dari 3 aplikasi *open source*, yaitu: Elasticsearch, Logstash, dan Kibana. Ketiga aplikasi ini memiliki fungsi yang berbeda-beda juga.

Gambar 2.1 Alur pengolahan data pada Elastic Stack

Pada Gambar 2.1 dapat dilihat alur data yang terjadi bermula dari sebuah *log* dan basis data, lalu diolah oleh Logstash. Hasil data yang telah diolah dilanjutkan ke Elasticsearch untuk dilakukan pengindeksan data. Setelah dilakukan pengindeksan, data dapat diolah untuk divisualisasikan dan dianalisis menggunakan Kibana.

2.1.1 Elasticsearch

Elasticsearch merupakan sebuah mesin pencari dan analisis untuk semua jenis data (Elastic, 2019). Analisis data terdiri dari dua proses, yaitu: mengatur urutan data, dan mengorganisasikanya ke dalam suatu pola, kategori, serta satuan uraian dasar (Moleong, 2007). Tahapan analisis data ini sama seperti yang dilakukan oleh Elasticsearch. Pada Elasticsearch terjadi suatu proses yang dinamakan *data ingestion*. Pada proses tersebut data mentah diolah melalui tahapan penguraian, normalisasi, dan diperkaya dengan data tambahan sebelum

dilakukan proses pengindeksan. Pengindeksan dilakukan untuk mempercepat aplikasi dalam mengatur urutan dan mengorganisasikan data.

Data yang telah terindeks dapat dilihat dalam format JSON. Contoh pengambilan data dengan format JSON dapat dilihat pada Gambar 2.2. Gambar di bawah adalah tampilan awal dari Elasticsearch yang menandakan bahwa Elasticsearch telah ter-*install* pada sistem.


```
{
  "name" : "ppsdm-node",
  "cluster_name" : "ppsdm-elasticsearch",
  "cluster_uuid" : "goqVop6QRXS2gxBe9aqRjQ",
  "version" : {
    "number" : "7.3.1",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "4749ba6",
    "build_date" : "2019-08-19T20:19:25.651794Z",
    "build_snapshot" : false,
    "lucene_version" : "8.1.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```


Gambar 2.2 Tampilan awal Elasticsearch

Data yang telah terindeks juga dapat diolah dengan melakukan penyaringan data melalui kata kunci ataupun *id* dari data. Hal ini dilakukan untuk mengelompokkan data dan menyaring data yang tidak diperlukan. Penyaringan data dengan data yang telah terindeks akan memakan waktu lebih singkat dibandingkan dengan penggunaan *query* pada basis data secara langsung. Contoh hasil dari penyaringan data dapat dilihat pada Gambar 2.3.


```
{
  "took" : 26,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 39,
      "relation" : "eq"
    },
    "max_score" : 17.638153,
    "hits" : [
      {
        "_index" : "activity_grades",
        "_type" : "_doc",
        "_id" : "8846",
        "_score" : 17.638153,
        "_source" : {
          "@version" : "1",
          "item_name" : "Tes Materi 9",
          "id" : 8846,
          "first_name" : "Mochamad Noer Isnin ",
```


Gambar 2.3 Hasil penyaringan data dengan nama “Mochamad Noer Isnin”

Elasticsearch juga menyediakan fitur *query* terhadap data yang telah terindeks. Format keluaran dari penggunaan SQL ini dapat berupa *human readable* ataupun *binary format*. *Human readable* terdiri dari format CSV, TSV, TXT, JSON, dan YAML. Contoh penggunaan

dari format *human readable* dapat dilihat pada Gambar 2.4, sedangkan *binary format* terdiri dari format CBOR dan Smile yang dapat dilihat pada Gambar 2.5.

```
                                                        nama_pelatihan                        |     nilai
--------------------------------------------------------------------------------------------+---------------
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|100
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|176
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|53
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|81
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|80
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|95
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|80
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|80
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|100
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|null
2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar|96
```

Gambar 2.4 Penggunaan SQL dengan keluaran format TXT

```
:)
·��columns���nameMnama_pelatihan�typeCtext��ADnilaiBClong���rows��y2/7_12/
-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel
Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatiha
PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat
Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12
/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG
Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel
Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatiha
PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat
Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2
/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG
Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel
Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatiha
PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat
Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_1
/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2/7_12/7_LPKN-BIG
Hotel Sahira_Pelatihan PBJ Tingkat Dasar���y2/7_12/7_LPKN-BIG Hotel
Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan
PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat
Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2
/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG
Hotel Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel
Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan
PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat
Dasar$·���y2/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar$·���y2
/7_12/7_LPKN-BIG Hotel Sahira_Pelatihan PBJ Tingkat Dasar!��y2/7_12/7_LPKN-BIG
```

Gambar 2.5 Penggunaan SQL dengan keluaran format Smile

Perintah SQL yang dapat dilakukan juga terbatas. Perintah SQL yang dapat digunakan dalam Elasticsearch, yaitu:

a. *DESCRIBE TABLE*

Perintah *describe table* berfungsi untuk mendeskripsikan sebuah indeks.

b. *SELECT*

Perintah *select* berfungsi untuk menampilkan data yang tersimpan pada suatu indeks.

c. *SHOW TABLES*

Perintah *show tables* berfungsi untuk menampilkan semua indeks yang ada.

d. *SHOW COLUMNS*

Perintah *show columns* berfungsi untuk menampilkan semua kolom pada suatu indeks.

e. *SHOW FUNCTIONS*

Perintah *show functions* berfungsi untuk menampilkan fungsi-fungsi yang tersedia.

Contoh penerapan Elasticsearch pada bidang edukasi adalah penelitian yang dilakukan oleh (Atmajaa & Yulianto, 2018). Penelitian tersebut membahas mengenai pemanfaatan Elasticsearch dalam mencari tugas akhir dari para mahasiswa Politeknik Negeri Madiun (PNM). Tujuan dari penelitian tersebut untuk membantu mahasiswa lain yang membutuhkan referensi dalam tugas akhir. Penelitian tersebut mengkombinasikan framework Laravel dengan Elasticsearch. Elasticsearch berfungsi sebagai sebuah basis data *NoSQL* yang akan melakukan pengindeksan pada semua data dari aplikasi yang telah dibuat dengan Laravel. Penelitian tersebut juga membuktikan bahwa Elasticsearch dapat diintegrasikan dengan berbagai macam sistem, salah satunya sistem dengan struktur MVC (*Model View Controller*) dan *framework* Laravel.

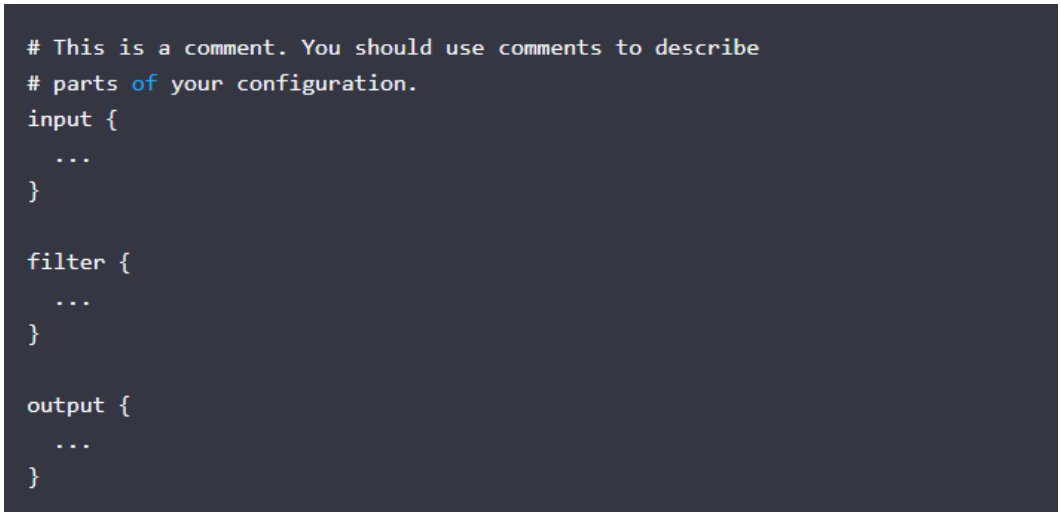
Contoh penelitian lain yaitu optimalisasi *query* dengan menerapkan Elasticsearch untuk membantu pengguna melakukan pencarian obat-obatan berdasarkan kata kunci yang dicari. Penelitian tersebut memanfaatkan API (*Application Programming Interface*) yang telah tersedia di dalam Elasticsearch untuk mengolah data obat yang ada, sehingga dapat mengurangi waktu eksekusi *query* (Bhadane, Mody, Shah, & Sheth, 2014).

2.1.2 Logstash

Logstash merupakan salah satu produk utama dari Elastic Stack, Logstash digunakan untuk mengumpulkan dan mengolah data sebelum diteruskan ke Elasticsearch (Elastic, 2019). Data yang diterima Logstash dapat diperoleh melalui basis data, Apache Kafka, Amazon S3, dan Beats.

Untuk memperoleh input data yang tepat, Logstash memerlukan sebuah dokumen konfigurasi. Dokumen konfigurasi Logstash terdiri dari 3 bagian, yaitu: *input*, *filter,* dan *output*.

Kerangka dokumen konfigurasi Logstash dapat dilihat pada Gambar 2.6. Tiap bagian konfigurasi memiliki *plugin*-nya masing-masing yang telah disediakan. *Plugin* ini berguna untuk membantu Logstash mengambil data sesuai kebutuhan, seperti menerapkan pengambilan terjadwal, mengeluarkan keluaran *log* yang disimpan dalam sebuah dokumen, dan sebagainya.

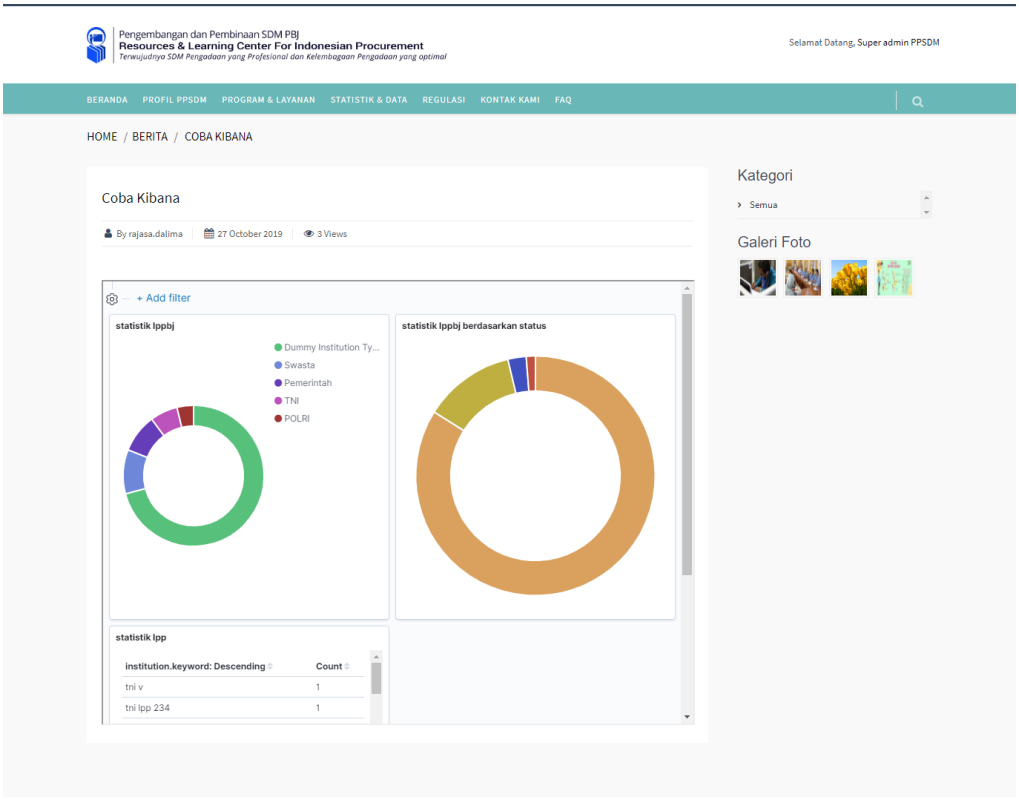


Gambar 2.6 Kerangka dokumen konfigurasi Logstash

Contoh penelitian mengenai pemanfaatan Logstash adalah penelitian yang dilakukan oleh (Arifin, Sugiartowo, & Susilowati, 2018). Penelitian tersebut menggunakan Logstash untuk menarik data *log* yang ada pada server. Data yang telah ditarik dari *log* server kemudian diuraikan dan diteruskan ke Elasticsearch. Setelah itu data ini digunakan untuk membuat *dashboard* admin yang menampilkan data seperti berapa banyak pengguna yang gagal melakukan *log in*, berapa pengguna yang dapat melakukan *log in*, dan berapa persentase keseluruhannya. Penarikan *log* server yang dilakukan oleh Logstash ini mampu membuat kegiatan pemantauan *log* server menjadi lebih efesien.

2.1.3 Kibana

Kibana merupakan sebuah alat visualisasi dan manajemen data untuk Elasticsearch (Elastic, 2019). Data yang divisualisasikan dapat berupa grafik, *metric*, tabel, ataupun gambar. Kibana juga menyediakan fitur *dashboard*, fitur ini berfungsi untuk mengumpulkan data yang telah divisualisasikan ke dalam satu halaman *dashboard*. *Dashboard* yang telah dibuat dapat ditampilkan pada halaman *website* dengan menggunakan *tag iframe*. Contoh *dashboard* yang ditampilkan dapat dilihat pada Gambar 2.7.



Gambar 2.7 Statistik LPPBJ dengan Kibana

Contoh penelitian mengenai pemanfaatan Kibana adalah penelitian yang dilakukan oleh (Shah, Willick, & Mago, 2018). Penelitian tersebut memanfaatkan kemampuan Kibana untuk membuat visualisasi dari data yang telah terindeks oleh Elasticsearch. Pada penelitian tersebut, aplikasi Elasticsearch, Logstash, dan Kibana dikonfigurasi untuk mengambil data secara *realtime*. Hal ini membuat statistik yang dibuat Kibana menjadi lebih efesien dan akurat.

2.2 *Big Data*

Big data merupakan sebuah tren dalam pengelolaan informasi yang terjadi dalam kurun waktu lima tahun terakhir. *Big data* memungkinkan sebuah organisasi dapat menyimpan, mengelola, dan memanipulasi sejumlah besar data yang berbeda dengan kecepatan dan waktu yang tepat (Hurwitz, Halper, Nugent, & Kaufman, 2013).

Dalam pengelolaan *big data* terdapat tiga sumber perolehan data, yaitu: media sosial, lintas data, dan mesin temu balik informasi. Adapun hal yang perlu diperhatikan dalam pengelolaan *big data*, yaitu (Toba, 2015):

a. *Hacking Skills*

   Hacking skills adalah pemahaman dalam menyamakan persepsi dari berbagai format data yang ada. Sebagai contoh, bagaimana cara menyamakan data yang diambil dari Rumah Sakit A dengan data yang diambil dari Rumah Sakit B dengan penamaan kolom yang berbeda, sehingga analis mendapat data yang akurat dan jelas dari kedua rumah sakit tersebut.

b. *Substantive Expertise*

   Pada *substantive expertise* dibutuhkan pandangan dari seorang ahli (*expert*) untuk memahami sebuah data yang didapat agar menjadi sebuah informasi.

c. *Math and Statistics Knowledge*

   Pada *math and statistics knowledge* menjelaskan bahwa untuk mendapat sebuah informasi dari sekumpulan data, diperlukan juga pemahaman tentang pemodelan matematis. Hal ini sangat mempengaruhi keakuratan dan ketepatan informasi yang diperoleh.

Implementasi pengelolaan *big data* dapat dilakukan menggunakan Elastic Stack. Logstash akan digunakan sebagai sebuah mesin penyamaan persepsi dari berbagai format yang ada. Elasticsearch akan digunakan sebagai mesin temu balik informasi, sedangkan Kibana akan digunakan sebagai mesin visualisasi dari data yang ada.

2.3 *High Availability*

High availability atau *fault tolerant* adalah sebuah desain infrastruktur yang digunakan untuk meminimalisir kegagalan pada sebuah sistem. Cara kerja dari infrastruktur ini adalah dengan menyediakan sistem cadangan yang akan berperan untuk menggantikan sistem utama jika terjadi kegagalan.

Adapun percobaan penggunaan *high availability* pada *Network File System* (NFS) menghasilkan kesimpulan sebagai berikut (Umam, Handoko, & Rizqi, 2018):

a. Semakin besar data yang diubah, semakin lama sinkronisasi terjadi.
b. Hasil pengujian pada sistem dengan *high availability* memiliki akurasi 99%.
c. Sistem dengan *high availability* terdiri dari *node master* dan *node slave*.