

Terbit online pada laman web jurnal : <http://teknosi.fti.unand.ac.id/>

Jurnal Nasional Teknologi dan Sistem Informasi

| ISSN (Print) 2460-3465 | ISSN (Online) 2476-8812 |



Studi Kasus

Pemanfaatan Elasticsearch untuk Temu Kembali Informasi Tugas Akhir

Ardian Prima Atmaja^a, Susilo Veri Yulianto^b^{ab}Jurusan Teknik Politeknik Negeri Madiun, Jl. Serayu 84, Kota Madiun 63133, Indonesia

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 12 Oktober 2018

Revisi Akhir: 11 Januari 2018

Diterbitkan Online: 12 Januari 2018

KATA KUNCI

Information retrieval,

Elasticsearch,

Laravel,

Big Data

KORESPONDENSI

Telepon: +62 82132133275

E-mail: atmaja@pnm.ac.id

A B S T R A C T

Teknologi Informasi (TI) telah menjadi pendukung utama dalam menjalankan operasional di berbagai kebutuhan organisasi, termasuk perguruan tinggi. Bentuk dari produk teknologi informasi yang paling sering digunakan adalah sistem informasi yang dapat diakses dari perangkat-perangkat TI seperti laptop atau *smartphone*. Di perguruan tinggi, data-data digital yang terkumpul di pusat data dapat dimanfaatkan kembali sehingga dapat bermanfaat kembali bagi siapa saja yang membutuhkan. Salah satu data yang dapat dimanfaatkan kembali adalah Tugas Akhir mahasiswa dari tahun ke tahun. Hal ini membuka peluang untuk dilakukan *Big Data Analytics* di perguruan tinggi. Politeknik Negeri Madiun (PNM) merupakan perguruan tinggi negeri baru yang belum memiliki pusat data Tugas Akhir yang terintegrasi. Makalah ini dibangun sebuah pusat data Tugas Akhir yang dapat dilakukan pencarian kembali (*information retrieval*) oleh pengguna yang membutuhkannya sebagai referensi ilmiah, sekaligus sebagai inisiasi Big Data yang nantinya dapat dilakukan pengembangan analisis lebih lanjut. Sistem temu kembali informasi atau mesin pencari dokumen Tugas Akhir yang dikembangkan berbasis Elasticsearch dan *framework* PHP Laravel. Sistem ini juga akan diintegrasikan dengan *Single Sign On* (SSO) dan *Student Portal* PNM yang sudah ada, sehingga untuk mengakses informasi yang lebih lengkap seperti makalah publikasi ilmiah atau video Tugas Akhir (bila ada) hanya menggunakan otentifikasi satu akun yang dimiliki oleh civitas akademika PNM saja. Penelitian ini juga bertujuan sebagai inisiasi pengelolaan Big Data di lingkungan PNM yang dipastikan setiap tahun semakin bertambah banyak.

1. PENDAHULUAN

Di perguruan tinggi pada umumnya, teknologi informasi telah menjadi pendukung utama dalam menjalankan operasionalnya. Hari ini, berbagai aktifitas di perguruan tinggi hampir selalu melibatkan teknologi informasi, terutama yang berbentuk sistem informasi yang dapat diakses menggunakan PC, laptop atau *smartphone*. Data yang dapat diakses oleh akademisi antara lain data tentang kegiatan pengajaran, penelitian dan pengabdian dosen maupun mahasiswa yakni berbentuk Tugas Akhir, koleksi perpustakaan, dan lain sebagainya. Semua aktifitas tersebut dari waktu ke waktu menyebabkan digitalisasi data yang selanjutnya menyebabkan ledakan data di Perguruan Tinggi sehingga membuka peluang untuk dilakukan *Big Data Analytics* di perguruan tinggi. Data-data digital dalam jumlah banyak meninggalkan tentang apa yang mahasiswa dan akademisi lihat, apa yang mereka baca, keterlibatan dan perilaku mereka, penilaian, maupun tentang kepentingan dan preferensi mereka

sehingga menyediakan sejumlah besar data yang dapat ditambang untuk pengalaman pembelajaran [1].

Politeknik Negeri Madiun (PNM) sebagai perguruan tinggi politeknik negeri yang baru dinegerikan pada akhir tahun 2012 lalu belum memiliki sebuah database Tugas Akhir terintegrasi yang dapat diakses atau dilakukan temu kembali informasi (*information retrieval*) oleh pengguna yang membutuhkannya sebagai referensi ilmiah. Sehingga aksesibilitas mahasiswa menjadi terbatas untuk mengembangkan Tugas Akhir yang telah diselesaikan kakak tingkatnya. Namun setelah tahun 2016, ketika infrastruktur server telah terpasang dengan baik di PNM, wacana pengembangan mesin pencari dokumen Tugas Akhir ini mulai direncanakan. Hal ini seiring dengan semakin bertambahnya data dokumen Tugas Akhir dari tahun ke tahun serta perkembangan sistem-sistem informasi pengelolaan tugas Akhir dan akademik yang telah berjalan dengan baik pula.

Memperhatikan hal tersebut di atas, dimana di PNM telah tersedia infrastruktur teknologi informasi yang berupa perangkat server dan jaringan yang memadai namun di sisi lain pengelolaan pemanfaatan temu kembali informasi pada Big Data yang sedang berkembang, terutama dokumen Tugas Akhir yang bersifat digital yang belum terkelola dengan baik, maka diusulkan untuk dikembangkan sebuah sistem temu kembali informasi atau mesin pencari informasi Tugas Akhir yang dapat diakses oleh siapa saja yang membutuhkan serta telah terintegrasi dengan data akademik dan lainnya di PNM. Usulan ini juga bertujuan sebagai inisiasi pengelolaan Big Data di lingkungan PNM yang dipastikan setiap tahun semakin bertambah banyak.

Berdasarkan pada latar belakang yang telah disampaikan sebelumnya, maka dirumuskan untuk mengembangkan sebuah sistem temu kembali informasi atau mesin pencari informasi Tugas Akhir mahasiswa PNM berbasis web yang terintegrasi dengan pusat data yang sudah ada sehingga dapat diakses oleh pihak-pihak yang memerlukan menggunakan perangkat yang terhubung dengan internet. Sistem yang dibangun menjadi inisiasi pengelolaan Big Data yang akan dimanfaatkan oleh masyarakat civitas akademika PNM dan masyarakat umum.

Untuk menghindari meluasnya masalah yang diteliti, permasalahan yang akan diteliti dibatasi sebagai berikut:

1. Mesin pencari yang digunakan adalah Elasticsearch 5.6.3 berbasis Windows.
2. Sistem dibangun menggunakan framework PHP Laravel 5.3 yang berbasis bahasa pemrograman HTML5, CSS3, PHP 5.6.33, jQuery 1.11.2 dan database Microsoft SQL Server 2012 yang berjalan pada sistem operasi Microsoft Windows Server 2012. Fitur yang digunakan pada Laravel adalah Scout dengan kustomisasi konfigurasi *driver*.
3. Sistem temu kembali informasi Tugas Akhir nantinya dapat diakses menggunakan komputer, tablet maupun *smartphone* yang terhubung dengan internet dengan mengalami penyesuaian tampilan sesuai layar (*responsive*).
4. Data Tugas Akhir alumni yang digunakan untuk implementasi merupakan data riil dan terintegrasi yang bersumber dari database yang dikelola oleh masing-masing Program Studi Tahun Akademik 2017/2018.

2. TINJAUAN PUSTAKA

2.1. Big Data

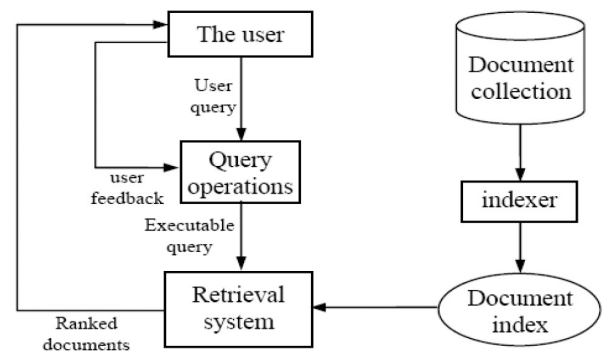
Big Data menjadi salah satu tren teknologi paling penting yang berpotensi mengubah cara organisasi menggunakan informasi untuk meningkatkan pengalaman pelanggan bahkan mengubah model bisnis mereka. Big Data adalah kombinasi teknologi manajemen data yang telah berkembang dari waktu ke waktu yang memungkinkan organisasi untuk menyimpan, mengelola, dan memanipulasi sejumlah besar data dengan kecepatan yang tepat dan pada waktu yang tepat untuk mendapatkan wawasan yang tepat. Cara untuk memahami data yang besar adalah bahwa data harus dikelola dengan baik sehingga dapat memberikan solusi terhadap kebutuhan bisnis yang dijalankan. Sebagian besar perusahaan berada pada tahap awal dengan perjalanan data besar mereka [2].

Asniar (2015) dalam makalahnya mengusulkan penggunaan Big Data Analytic di Perguruan Tinggi, yakni meliputi *Learning Analytic*, *Academic Analytic*, dan *Process Mining* [1]. *Learning Analytic* adalah melakukan analisis data pembelajaran secara real time sehingga dapat digunakan untuk memprediksi mahasiswa sukses dan mahasiswa yang beresiko akademik. Dengan *academic analytic* dapat dilakukan analisis secara *real time* terhadap data-data yang merupakan variabel pengukuran kinerja akademisi sehingga dapat diketahui staf-staf akademisi yang berprestasi maupun staf-staf akademisi yang kinerjanya sangat kurang dibandingkan dengan staf-staf akademisi lainnya. Sedangkan untuk *Process Mining*, data-data yang digunakan dapat diperoleh dari data log atau data aktifitas dari mahasiswa, dosen, dan unit-unit terkait dengan proses-proses dan aktivitas yang terjadi di Perguruan Tinggi untuk kemudian dilakukan *process analytic* dengan menggunakan *process mining* untuk menemukan model proses bisnis baru. Namun proses *analytic* tidak terbatas pada proses penemuan proses bisnis tapi juga memungkinkan untuk memeriksa kesesuaian, mendeteksi penyimpangan, memprediksi penundaan, mendukung pengambilan keputusan, dan merekomendasikan desain ulang proses. Tahapan yang dilalui dalam penelitian, pembangunan konsep, atau penyelesaian kasus, dituliskan pada bagian metodologi.

Pada usulan Asniar (2015) tersebut memerlukan repositori data yang bersumber dari banyak data di lingkungan akademik. Sistem yang dibangun ini mengumpulkan data-data tersebut melalui proses pengindeksan yang dilakukan oleh Elasticsearch. Pengindeksan juga akan mengkonversi data dari SQL menjadi NoSQL menggunakan platform tunggal dan terpadu. Hal ini akan menjadi inisiasi sebuah ekosistem Big Data pada PNM yang dapat dianalisis lebih lanjut di kemudian hari (*Big Data Analytic*) dengan menggabungkan data akademik lainnya.

2.2. Information Retrieval

Gerard Salton, pelopor dalam pencarian informasi dan salah satu tokoh terkemuka dari tahun 1960-an hingga 1990-an, mengusulkan sebuah definisi *Information Retrieval* dalam buku klasiknya 1968 [3], yakni bidang yang berkaitan dengan struktur, analisis, organisasi, penyimpanan, pencarian, dan pengambilan informasi. Meskipun dunia mengalami kemajuan besar dalam pemahaman dan teknologi pencarian dalam 40 tahun terakhir ini, definisi tersebut masih tepat dan akurat. Arsitektur *Information Retrieval* dapat digambarkan pada Gambar 1 di bawah ini.



Gambar 1. Arsitektur *Information Retrieval* [4]

2.3. Elasticsearch

Elasticsearch adalah mesin pencarian teks dan analitik yang bersifat *opensource* yang sangat *scalable* serta memungkinkan penggunaannya untuk menyimpan, mencari, dan menganalisis data dengan volume yang besar dengan cepat dan mendekati *real time*. Elasticsearch umumnya digunakan sebagai mesin/teknologi yang mampu mengoptimasi sebuah aplikasi yang memiliki fitur pencarian.

Penelitian Bhadane, et. Al (2014) mengoptimalkan perangkat lunak yang mereka bangun dengan menambahkan Elasticsearch sehingga mampu memudahkan pengguna dalam mengambil daftar obat-obatan berdasarkan kata kunci tertentu. Selain itu, Elasticsearch digunakan untuk meningkatkan *query* pada database konvensional sehingga membuatnya lebih cepat sebelum mengirimnya ke server. Hal ini mampu mengurangi latensi waktu yang digunakan ketika melakukan pencarian [5]. Sementara itu Kanade, et.al (2017) membangun analisis tekstual yang terlibat dalam pengembangan mesin pencari portal *ebook* untuk pencarian pencarian yang efisien. Mereka melibatkan analisis dalam mengembangkan mesin pencarian portal *ebook* untuk dapat menghasilkan hasil pencarian buku yang paling relevan [6].

Bedmar, et.al (2016) dalam makalahnya menjelaskan partisipasi tim LABDA dalam Tugas BioASQ 2016 4a pada pengindeksan biomedis semantik berskala besar. Pendekatan yang mereka ambil berdasarkan pada penggunaan mesin pencari *opensource* Elasticsearch. Hasil dari eksperimen menunjukkan bahwa pendekatan yang diambil mencapai *recall* yang tinggi dengan tetap mempertahankan waktu pemrosesan yang rendah. Mereka menyimpulkan bahwa Elasticsearch adalah sistem yang kompetitif dan *scalable* untuk mengindeks literatur biomedis [7]. Sedangkan Vidhya dan Vadivu (2016) mengimplementasikan server Elasticsearch untuk membangun mesin pencari yang membantu dalam mencari, mengambil dan mengunduh makalah penelitian yang disimpan di dalam database menggunakan *framework* Django yang mana proses *indexing* dilakukan oleh Elasticsearch [8]. Elasticsearch merupakan server pencarian berdasarkan Lucene yang memiliki skalabilitas dan pencarian mendekati real-time dapat digunakan untuk mencari semua jenis dokumen. Sementara itu, dalam hal implementasi Elasticsearch pada aplikasi web, Vadavalli (2014) mengembangkan sebuah aplikasi web menggunakan Ruby on Rails, yakni sebuah *framework* *opensource* berbasis pola Model-View-Controller (MVC). Pada aplikasi ini, program-program aplikasi dikelola dengan mengintegrasikan plugin JDBC River untuk melakukan *indexing* terhadap database MySQL sehingga menghasilkan model indeks *schema-less* dari Elasticsearch. Aplikasi yang dibangun juga mengimplementasikan fitur-fitur *full text search*, *faceting* hasil pencarian, dan *autocomplete* menggunakan sebuah *plugin* dari Ruby on Rails bernama *Tire* and *Searchkick Gems* [9].

Berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya, dapat disimpulkan bahwa Elasticsearch merupakan mesin pencari yang sangat handal dan dapat menyajikan hasil pencarian yang relevan. Elasticsearch juga terbukti mampu mengelola data pencarian yang sangat besar dengan tetap mempertahankan waktu pemrosesan yang rendah. Selain itu, Elasticsearch juga dapat berkolaborasi dengan *framework*

berbasis pola MVC, sehingga peneliti dapat menggunakan Laravel sebagai *framework* pemrograman antarmuka web pada sistem yang dibangun. Pemilihan Laravel sebagai *framework* pemrograman dikarenakan sistem yang dibangun ini merupakan bagian dari sistem lainnya yang sudah ada dan terintegrasi, sehingga dapat menjaga konsistensi pemilihan *framework* pada arsitektur sistem informasi yang ada di PNM.

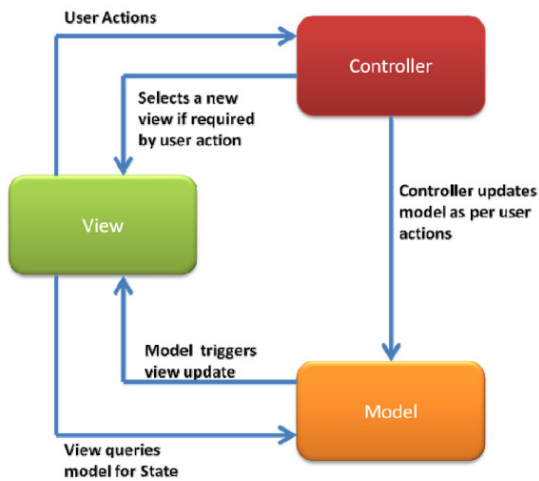
2.4. Laravel

Laravel adalah sebuah *framework* PHP yang bersifat *open source* yang dirilis dibawah lisensi MIT yang ditujukan untuk pengembangan aplikasi web mengikuti pola arsitektur Model-View-Controller (MVC). MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti manipulasi data, *controller*, dan *user interface* [10].

- 1) Model. Model mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
- 2) View. View adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web.
- 3) Controller. Controller merupakan bagian yang menjembatani model dan view.

Cara kerja konsep MVC ditunjukkan pada Gambar 2. Pada gambar tersebut terlihat adanya keterkaitan antara Controller, View, dan Model. Beberapa fitur yang terdapat di Laravel:

- 1) *Bundles*, yaitu sebuah fitur dengan sistem pengemasan modular dan tersedia beragam di aplikasi.
- 2) *Eloquent ORM*, merupakan penerapan PHP lanjutan menyediakan metode internal dari pola "*active record*" yang mengatasi masalah pada hubungan objek database.
- 3) *Application Logic*, merupakan bagian dari aplikasi, menggunakan Controller atau bagian Route.
- 4) *Reverse Routing*, mendefinisikan relasi atau hubungan antara Link dan Route.
- 5) *Restful Controllers*, memisahkan logika dalam melayani HTTP GET and POST.
- 6) *Class Auto Loading*, menyediakan loading otomatis untuk class PHP.
- 7) *View Composer*, adalah kode unit logikal yang dapat dieksekusi ketika view sedang loading.
- 8) *IoC Container*, memungkinkan obyek baru dihasilkan dengan pembalikan controller.
- 9) *Migration*, menyediakan sistem kontrol untuk skema database.
- 10) *Unit Testing*, tes untuk mendeteksi dan mencegah regresi.
- 11) *Automatic Pagination*, menyederhanakan tugas dari penerapan halaman.

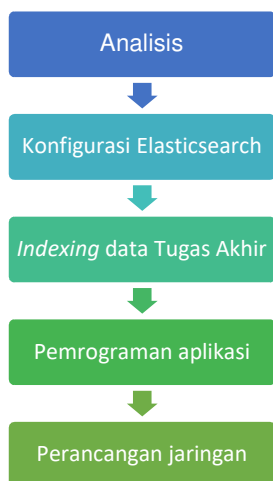


Gambar 2. Cara Kerja Arsitektur MVC [11]

Pada pengembangan mesin pencari Tugas Akhir ini, dimanfaatkan fitur Scout yang ada pada Laravel 5.3. Laravel Scout menyediakan solusi berbasis driver yang sederhana untuk menambahkan pencarian teks lengkap ke model Eloquent. Dengan memanfaatkan Model, Scout akan secara otomatis mengelola indeks pencarian yang akan sinkron dengan database. Pada Laravel 5.3, Scout menggunakan *search engine* Algolia secara default [12]. Selain Algolia, Laravel Scout juga dapat bekerja menggunakan beberapa *search engine* lainnya antara lain TNTSearch dan Elasticsearch. Untuk mengkolaborasikannya, pengguna harus melakukan konfigurasi tambahan pada tiap-tiap driver tersebut agar sistem pencarian dapat berjalan sebagaimana mestinya. Dan pada penelitian ini, Scout akan digabungkan dengan Elasticsearch dengan cara melakukan melakukan instalasi Elasticsearch terlebih dahulu serta melakukan kustomisasi pada *driver* Elasticsearch di Laravel.

3. METODOLOGI

Metode penelitian yang digunakan secara berurutan ditunjukkan pada Gambar 3:



Gambar 3. Alur Metodologi Penelitian

3.1. Analisis

Sebelum dilakukan analisis, terlebih dahulu dilakukan studi terhadap sumber-sumber pustaka yang mendukung penelitian yakni berupa buku pedoman akademik PNM tahun 2017-2018,

buku-buku teks penunjang teknologi mesin pencari atau temu kembali informasi (*information retrieval*), pemrograman web berbasis *framework* Laravel dan hasil penelitian yang telah atau sedang dikembangkan oleh para peneliti lainnya. Wawancara juga dilakukan untuk menggali informasi yang dimiliki oleh *stake holder*, dalam hal ini Unit Pelaksana Teknis (UPT) Perpustakaan dan Program Studi di PNM serta beberapa mahasiswa terutama yang sedang mengerjakan Tugas Akhir. Wawancara juga akan menyimpulkan bagaimana rencana skenario implementasi sistem yang telah dikembangkan ini dalam jangka waktu menengah dan panjang.

Pada tahap analisis akan dilakukan analisis permasalahan awal yang ditemukan pada domain permasalahan yaitu pada UPT Perpustakaan dan Program Studi. Tahapan ini akan memahami aturan bisnis dan *Student Portal* yang telah ada, yakni dalam hal penginputan dokumen Tugas Akhir yang dilakukan secara mandiri oleh mahasiswa yang telah dinyatakan lulus sidang serta menyelesaikan semua revisi pada laporan Tugas Akhirnya. Tahap analisis juga menentukan batasan masalah dan metode rekayasa perangkat lunak yang dipilih untuk mengembangkan sistem yang telah ada tanpa menggantikan fungsi dari sistem yang sudah ada dan digunakan. Tahap ini juga akan dilakukan analisis terhadap penelitian-penelitian yang sudah dilakukan sebelumnya serta analisis terhadap buku-buku teks yang mendukung penelitian ini.

Dari tahap ini dihasilkan metode rekayasa perangkat lunak yang digunakan yakni *Rapid Application Development (RAD)*, yang merupakan sebuah model proses pengembangan perangkat lunak yang menekankan pada siklus pengembangan yang sangat pendek dan menghasilkan komponen yang bersifat *reuse* sehingga sangat mengakomodasi pengembangan perangkat lunak secara fleksibel. Sumber data yang digunakan dalam pengembangan sistem ini berasal dari data tugas akhir tahun 2017 dan 2018 yang merupakan data riil yang diinputkan oleh prodi dan diperbarui oleh mahasiswa melalui aplikasi *Student Portal*.

3.2. Konfigurasi Elasticsearch

Elasticsearch yang telah diunduh secara gratis melalui website resminya, terlebih dahulu dilakukan instalasi di server yang akan menyediakan sistem yang sedang dibangun. Dalam melakukan konfigurasi, ditentukan beberapa batasan dalam hal pengelolaan server Elasticsearch. Pada penelitian ini, Elasticsearch versi 5.6.3 diinstall pada Windows Server menggunakan port 9200. Untuk proses indexing data, Elasticsearch berkolaborasi dengan Laravel. Data yang diindeks berasal dari database yang telah terintegrasi dengan data-data lainnya pada *Data Center* PNM.

3.3. Indexing Data Tugas Akhir

Database dengan tabel-tabel penyimpanan data Tugas Akhir yang telah ada pada *Data Center* PNM dilakukan *indexing* atau penyimpanan data dengan tujuan merekam daftar yang mengandung istilah atau frasa (menyatakan penulis, judul, isi abstrak, dan sebagainya). Data pada database yang diindeks antara lain; NPM, nama mahasiswa, judul Tugas Akhir, para pembimbing, para penguji, abstrak, *keywords*, dan kode rak perpustakaan. Data hasil *indexing* ini nantinya akan digunakan sebagai indeks dalam proses pencarian oleh pengguna. Entitas-entitas data (tabel) yang telah ada pada *data warehouse* PNM dianalisis dan disesuaikan untuk proses *indexing*. *Data*

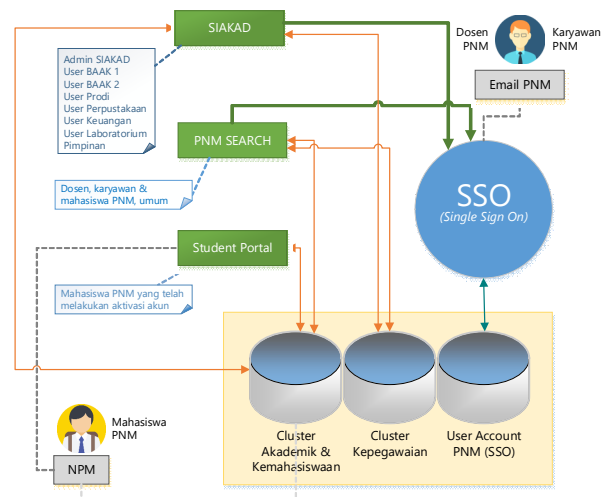
warehouse di PNM yang digunakan berbasis Microsoft SQL Server. Database ini akan digunakan sebagai penyimpan dan penghubung data-data pada sistem yang dibangun.

3.4. Pemrograman Aplikasi

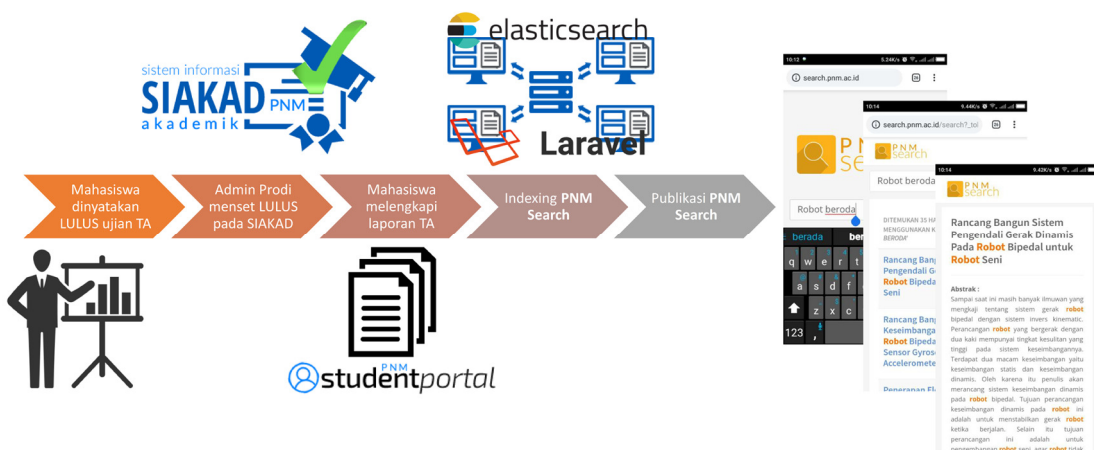
Tahap ini dilakukan pendesainan tampilan antarmuka halaman-halaman web menggunakan *framework* Laravel yang berbasis bahasa pemrograman web HTML5 dan CSS3. Sedangkan untuk pemrograman dinamis digunakan bahasa pemrograman PHP 5. Tahap pemrograman dilakukan berdasarkan pada tahap analisis dan penyesuaian dan pengintegrasian Elasticsearch dengan database yang telah dilakukan sebelumnya. Hasil dari pemrograman akan dikoneksikan dengan database Microsoft SQL Server untuk melakukan pengolahan data di sisi mahasiswa. Untuk proses otentifikasi dosen dan staf PNM, dilakukan integrasi dengan database *Single Sign On* (SSO) PNM. Sehingga baik mahasiswa dan dosen atau staf PNM dapat melakukan login untuk mendapatkan akses lebih lanjut yakni menampilkan artikel ilmiah dan video (bila tersedia) dari Tugas Akhir tersebut.

Antarmuka dibuat secara responsif dengan memanfaatkan Bootstrap. Bootstrap merupakan sebuah *open source toolkit* yang dikembangkan dengan HTML, CSS, dan JS. Bootstrap memungkinkan membuat halaman web dapat menyesuaikan tampilannya ketika diakses menggunakan perangkat dengan resolusi layar yang berbeda, baik berupa *browser* pada komputer, *smartphone* maupun tablet [13]. Dalam mengimplementasikan Bootstrap pada halaman web, pengembang hanya tinggal memanggil nama class untuk CSS dan nama library untuk Javascript. Dalam hal ini, digunakan nama library *bootstrap.min.js* untuk Javascript dan *style.css* sebagai CSS yang di dalamnya sudah ditambahkan kode responsif Bootstrap. Pada bagian akhir dari alur pemrosesan Tugas Akhir yang ada di PNM pada Gambar 5 menunjukkan hasil halaman web yang telah menyesuaikan tampilannya (responsif).

komputer server yang sudah terinstall sistem operasi Windows Server 2012. Pada komputer server, dilakukan instalasi sistem temu kembali informasi tugas akhir tersebut (PNM Search) agar dapat diakses oleh perangkat klien secara *online* melalui sebuah website portal mahasiswa yakni <http://search.pnm.ac.id>. Klien dalam hal ini adalah seluruh civitas akademika PNM serta masyarakat umum. Gambar 4 menunjukkan infrastruktur terintegrasi dari database, sistem informasi dan jaringan dari sistem informasi evaluasi kegiatan mengajar dosen yang telah dibangun. Terlihat bahwa sistem yang dibangun merupakan bagian dari integrasi sistem-sistem yang telah ada sebelumnya, antara lain; Sistem Informasi Akademik (SIKAD) yang merupakan sumber data akademik termasuk tugas akhir, dan *Student Portal* yang merupakan laman privat setiap mahasiswa di PNM yang berfungsi sebagai portal informasi akademik di PNM. Database yang digunakan adalah pada *cluster* akademik dan kemahasiswaan serta *cluster* kepegawaian. Selain itu, database SSO yang digunakan merupakan data akun tunggal yang wajib dimiliki oleh seluruh dosen dan tenaga kependidikan di PNM yang digunakan untuk mengakses aplikasi-aplikasi yang ada di PNM.



Gambar 4. Peta Infrastruktur PNM Search Terintegrasi di PNM



Gambar 5. Alur Pemrosesan Tugas Akhir di PNM

3.5. Perancangan Jaringan

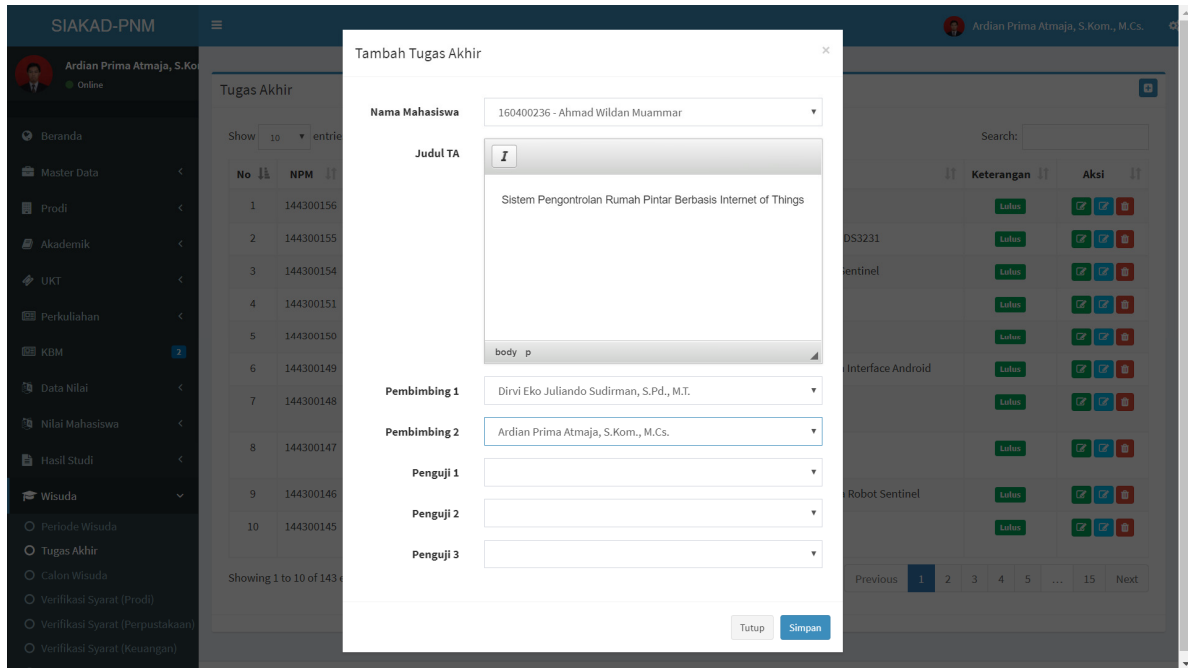
Setelah dilakukan desain terhadap perangkat lunak, maka dilakukan perancangan perangkat keras yang terdiri dari perangkat jaringan yang menghubungkan perangkat klien dengan

4. HASIL DAN PEMBAHASAN

Sistem yang dibangun terdiri dari tiga bagian yakni SIKAD sebagai input data tugas akhir mahasiswa, Student Portal sebagai sistem untuk melakukan updating data terkait data tugas akhir,

dan PNM Search sendiri sebagai laman mesin pencariannya. Ketiga aplikasi tersebut saling berkaitan satu sama lain. Gambar 5 menunjukkan alur pemrosesan data tugas akhir di PNM.

pada Gambar 9. Pada laman ini, pengguna dapat melakukan pencarian dokumen tugas akhir menggunakan kata kunci judul, kata kunci, abstrak, NPM, dan nama mahasiswa. Sedangkan



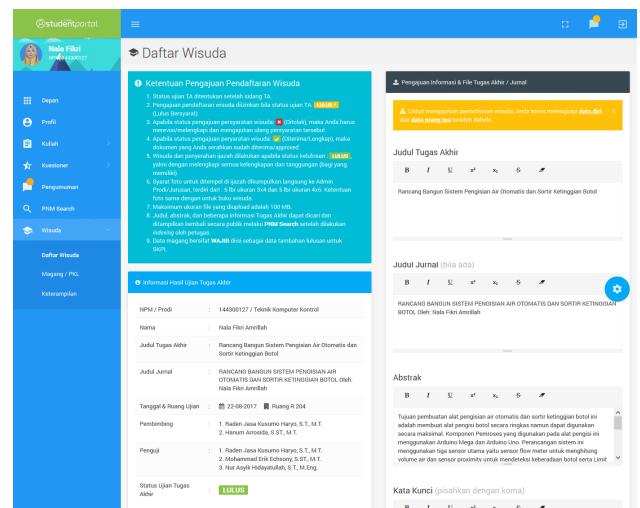
Gambar 6. Laman Input Data Tugas Akhir di SIKAD

Judul tugas akhir diinputkan oleh para petugas administrasi program studi melalui SIKAD yang dapat diakses melalui alamat <http://akademik.pnm.ac.id> pada saat awal semester ketika mahasiswa yang bersangkutan mengambil matakuliah Tugas Akhir. Judul Tugas Akhir tersebut merupakan data awal yang disimpan di server. Pada perjalanan prosesnya, judul tugas akhir tersebut berkembang menjadi sebuah dokumen tugas akhir yang tak hanya berisi judul melainkan juga berisi abstrak, kata kunci, pembimbing, penguji, tanggal ujian, serta status ujian. Data-data ini didapatkan setelah mahasiswa tersebut melewati fase ujian Tugas Akhir. Gambar 6 merupakan antarmuka laman input data Tugas Akhir melalui aplikasi SIKAD yang sekaligus menandakan bahwa mahasiswa tersebut sedang mengambil matakuliah Tugas Akhir.

Setelah status Tugas Akhir mahasiswa dinyatakan lulus ujian Tugas Akhir baik dengan perbaikan maupun tanpa perbaikan, maka dokumen Tugas Akhir dapat dilengkapi atau disesuaikan melalui laman Student Portal masing-masing mahasiswa di alamat <http://student.pnm.ac.id> sebagai syarat mengikuti wisuda. Persyaratan kelengkapan wisuda sesuai dengan kebijakan masing-masing program studi. Laman untuk melengkapi persyaratan wisuda, termasuk di dalamnya terdapat informasi dan dokumen Tugas Akhir ditunjukkan pada Gambar 7.

Setelah semua dokumen Tugas Akhir diperbarui atau disesuaikan dan telah lolos verifikasi oleh para petugas, maka data Tugas Akhir telah siap untuk dilakukan pengindeksan untuk selanjutnya dapat ditampilkan di laman PNM Search yang dapat diakses melalui alamat <http://search.pnm.ac.id>. Hasil *report* proses pengindeksan ditunjukkan pada Gambar 8. Setelah semua data berhasil dilakukan pengindeksan, laman sistem temu kembali informasi yang telah dibangun dapat dijalankan seperti tampak

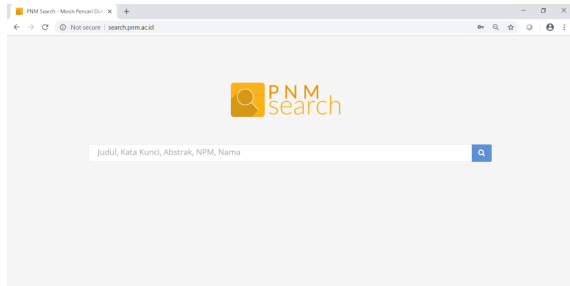
Gambar 10 merupakan laman hasil pencarian. Detail hasil pencarian Tugas Akhir ditampilkan pada Gambar 11. Pada laman detail hasil pencarian ini, beberapa hasil Tugas Akhir yang berupa produk/alat dapat ditampilkan videonya. Informasi lainnya yang ditampilkan pada laman detail hasil pencarian antara lain dosen pembimbing, dosen penguji, serta informasi kapan Tugas Akhir tersebut diselenggarakan. Para pengunjung juga dapat mencari dokumen tersebut di perpustakaan dengan mengacu pada nomor rak yang tertera di laman tersebut.



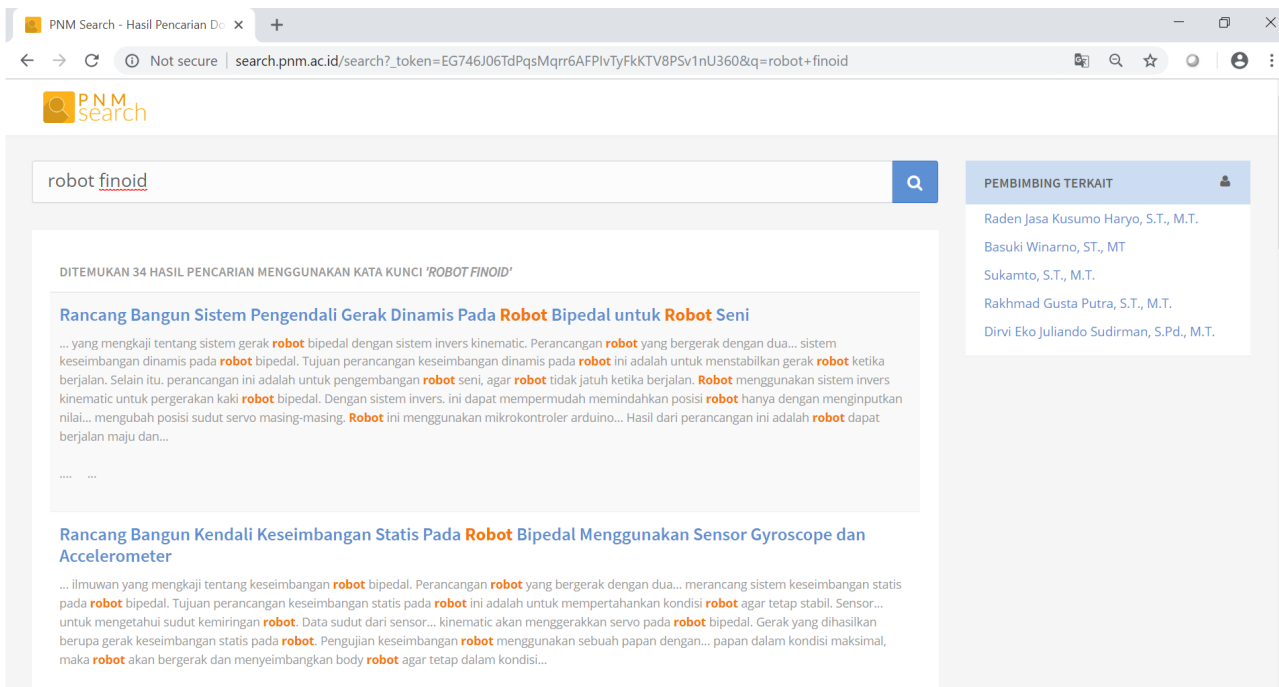
Gambar 7. Student Portal untuk Melengkapi Informasi Tugas Akhir

```
C:\xampp\htdocs\search2>php artisan scout:import "App\MsTA"
Mengimpor model [App\MsTA] ke dalam ID: 114
Mengimpor model [App\MsTA] ke dalam ID: 214
Mengimpor model [App\MsTA] ke dalam ID: 314
Mengimpor model [App\MsTA] ke dalam ID: 415
Mengimpor model [App\MsTA] ke dalam ID: 515
Mengimpor model [App\MsTA] ke dalam ID: 617
Mengimpor model [App\MsTA] ke dalam ID: 717
Mengimpor model [App\MsTA] ke dalam ID: 817
Mengimpor model [App\MsTA] ke dalam ID: 917
Mengimpor model [App\MsTA] ke dalam ID: 942
Semua [App\MsTA] data telah berhasil diimpor.
C:\xampp\htdocs\search2>
```

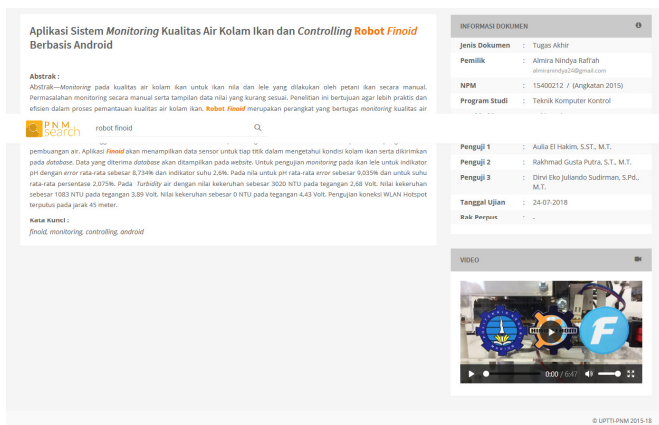
Gambar 8. Indexing Report Data Tugas Akhir



Gambar 9. Laman Awal PNM Search



Gambar 10. Laman Hasil Pencarian PNM Search



Gambar 11. Laman Detail Hasil Pencarian PNM Search

5. KESIMPULAN DAN SARAN

Sistem temu kembali informasi (mesin pencari) Tugas Akhir PNM telah berhasil dikembangkan serta digunakan dengan baik melalui URL <http://search.pnm.ac.id> dan telah terintegrasi dengan pusat data PNM. Sistem ini berbasis Laravel Scout & Elasticsearch yang mudah diimplementasikan dengan tampilan yang menyesuaikan layar (*responsive*) sehingga dapat diakses oleh pengguna menggunakan berbagai perangkat yang terhubung dengan internet tanpa harus menggunakan otentifikasi login terlebih dahulu. Hasil pencarian dapat ditampilkan dengan cepat sesuai dengan kata kunci yang diberikan serta dari hasil pencarian tersebut dapat ditampilkan kembali detailnya. Sistem ini bermanfaat bagi mahasiswa pada khususnya untuk melakukan pencarian referensi penelitian atau mendukung Tugas Akhir mereka, dan bagi masyarakat pada umumnya.

UCAPAN TERIMA KASIH

Makalah ini merupakan bagian dari skema Penelitian Kompetitif Internal PNM tahun 2018 yang didanai oleh DIPA PNM No. SP DIPA-042.01.2.400873/2018 sesuai dengan Surat Perjanjian Pelaksanaan Penelitian Kompetitif No. 186/PL33.110/LT/2018. Terima kasih pula kepada Direktur dan Kepala Pusat Penelitian dan Pengabdian Masyarakat (P3M) PNM yang telah mendukung penelitian hingga terbitnya makalah ini.

DAFTAR PUSTAKA

[1] Asniar, "Penggunaan Big Data Analytic di Perguruan Tinggi," in *Konferensi Nasional Sistem Informasi (KNSI)*, 2015.

- [2] J. Hurwitz, A. Nugent, F. Halper, and M. Kaufman, *Big Data for Dummies; A Wiley Brand*. John Wiley & Sons, 2013.
- [3] W. S. Cooper, "Automatic Information Organization and Retrieval. Gerard Salton," *Libr. Q.*, vol. 39, no. 3, pp. 276–277, 2009.
- [4] B. Liu and B. Liu, "Information Retrieval and Web Search," in *Web Data Mining*, 2011, pp. 211–268.
- [5] C. Bhadane, H. A. Mody, D. U. Shah, and P. R. Sheth, "Use of Elastic Search for Intelligent Algorithms to Ease the Healthcare Industry," *Int. J. Soft Comput. Eng.*, vol. 3, no. 6, pp. 222–225, 2014.
- [6] P. Kanade, A. Sadasivan, K. Dhuri, M. Muralidaran, and M. Mohan, "Search Engine For Ebook Portal," vol. 6, no. 5, pp. 77–81, 2017.
- [7] I. Segura-Bedmar, A. Carruana, and P. Martínez, "LABDA at the 2016 BioASQ challenge task 4a: Semantic Indexing by using Elasticsearch," in *Proceedings of the Fourth BioASQ workshop*, 2016, pp. 16–22.
- [8] R. Vidhya and G. Vadivu, "Research document search using Elastic search," *Indian J. Sci. Technol.*, vol. 9, no. 37, 2016.
- [9] S. K. Vadavalli, "Implementing Elasticsearch in a Web Application Using Ruby on Rails," Doctoral dissertation, West Virginia University, 2014.
- [10] T. Otwell, "Laravel-the php framework for web artisans." Internet: <https://laravel.com/>, [March 23, 2018].
- [11] A. Verma, "Mvc Architecture: a Comparative Study Between Ruby on Rails and Laravel," *Indian J. Comput. Sci. Eng.*, vol. 5, no. 5, pp. 196–198, 2014.
- [12] T. Otwell, "Laravel Scout." Internet : <https://laravel.com/docs/5.3/scout>, [March 23, 2018].
- [13] M. Otto. dkk, "Bootstrap." Internet : https://getbootstrap.com, [March 23, 2018].

BIODATA PENULIS



Ardian Prima Atmaja
Dosen Prodi Teknologi Informasi, Jurusan Teknik Politeknik Negeri Madiun, memiliki bidang keahlian *computer science*. Aktif melakukan penelitian di bidang aplikasi terapan berbasis web, *mobile*, dan IoT.



Susilo Veri Yulianto
Dosen Prodi Teknologi Informasi, Jurusan Teknik Politeknik Negeri Madiun, memiliki bidang keahlian *software engineering*. Aktif penelitian di bidang aplikasi terapan, dan sistem informasi manajemen *enterprise*.