**3. ANALISIS DAN PERANCANGAN SISTEM**

**3.1 Analisis**

**3.1.1 Analisis Sistem yang Sedang Berjalan**

Pada sistem mesin pencari yang telah ada saat ini, pengguna mengirimkan *query* pada siste pencarian dimana kemudian diproses oleh sistem. Pengguna juga memperoleh hasil pencarian dari sistem. Pengembang *website* diperlukan untuk memungkinkan *web crawler* menemukan hasil pencarian. Pihak administrator *website* menetapkan aturan-aturan tertentu pada hasil informasi yang diambil oleh *crawler*. Informasi ini termasuk berbagai kendala saat akses data. Berikut ini gambaran sistem mesin pencari yang telah ada saat ini

**Gambar 3.1 *Use Case Diagram* Sistem Mesin Pencari yang Sedang Berjalan**

Fungsi utama dalam suatu mesin pencari yaitu pemrosesan *query*. Pemrosesan *query* merupakan fungsi utama pencarian yang berhubungan dengan pengguna. Objek utama yang terlibat dalam fungsi ini antara lain pengguna, pencari, *presenter*, dan *store server*. Pengguna mengirimkan *query* yang diberikan kepada *presenter* yang kemudian diserahkan kepada pencari. Pencari mencari kata kunci dan mnciptakan daftar dokumen yang memberikan alamat yang berisi seluruh URL yang berhubunan dengan kata kunci yang disebutkan pengguna. *Presenter* kemudian melakukan *request* kepada *store server* untuk mengembalikan dokumen pada setiap URL. Hal ini dapat menciptakan rangkuman dari tiap URL dan menampilkannya kepada pengguna dalam bentuk daftar URL bersama dengan rangkuman penjelasan dari tiap URL tersebut. Pengguna juga dapat memberi umpan baili kepada setiap URL yang dikunjungi pengguna yang kemudian dapat digunakan untuk pemberian peringkat.

**Gambar 3.2 *Sequence Diagram* *Query Processing***

mesin pencari memiliki sebuah program yang terus berjalan yang disebut sebagai *crawler* yang mengindeks hasil pencarian yang berkaitan dengan yang diinginkan pengguna. Kerja utama mesin pencari *crawler* meliputi berbagai objek seperti *server* URL, *crawler, website* dan *store server*. *Crawler* melakukan*request* kepada *server* URL untuk menyediakan URL selanjutnya untuk di-*crawl*. *Crawler* kemudian melakukan *request* kepada file robot.txt pada *website* yang memindai 65 izin akses dai halaman *website*. *Crawler* tersebut kemudian bergantung kepada permintaan izin akses halaman *website* dan menyimpannya di *store server*.

**Gambar 3.3 *Sequence Diagram* pada *Crawler***

*Indexer* merupakan program lain yang terus berjalan yang berfungsi setelah tugas utama *crawler*. Tugas utama *indexer* yaitu menyusun halaman *website* yang telah di-*crawl*. Objek yang terkait dengan *indexer* yaitu *indexer*, *store server,* dan *server URL*.

*Indexer* melakukan request halaman baru dari *store server*. *Indexr* kemudian memindai halaman dan menyiapkan *forward index* intuk tiap halaman. *Forward index* ini kemudian disimpan dalam *store server* dan *link* yang ditemukan pada URL tertentu juga disimpan pada *server URL* untuk *crawling*.

**Gambar 3.4 *Sequence Diagram* pada *Indexer***

*Sorter* adalah program utama lain dari sistem mesin pencari yang berfungsi mengurutkan dokumen menurut signifikansi dokumen kepada kata-kata yang ada pada dokumen dengan membuat indeks yang terbalik. Berbagai objek yang terlibat antara lain *sorter* dan *store server*.

*Sorter* meminta *forward index* dari *store server* dan kemudian menciptakan indeks yang terbalik untuk tiap kata dan menyimpan nomor ID dari data hasil pencarian untuk diurutkan lagi berdasarkan tingkat kepentingannya.

**Gambar 3.5 *Sequence Diagram* pada *Sorter***

**3.1.2 Analisis Permasalahan**

Berdasarkan hasil analisis terhadap sistem mesin pencari yang berjalan pada *website* *e-commerce* saat ini, ditemukan beberapa masalah yang timbul:

1. Melakukan pencarian pada *database* relasional pada *website* memperlambat proses pencarian. Hal iniakan diperparah jika jumlah data dalam *website* sangat besar, dengan jumlah *field* yang banyak, serta relasi yang rumit.
2. Berdasarkan hasil studi lapangan terhadap enam *website e-commerce* penjualan kosmetik terbaik di Indonesia, masih banyak yang belum menerapkan fitur pencarian seperti *autocomplete* yang secara otomatis melengkapi kata kunci yang dimasukan pengguna, fitur *autocorrect* yang memperbaiki kesalahan pengetikan kata kunci yang dilakukan pengguna, serta fitur *autosuggestion* yang memberikan hasil pencarian yang relevan sebagai pengganti hasil pencarian yang tidak ditemukan.

**3.1.3 Analisis Kebutuhan**

Pada tahap analisis kebutuhan dilakukan analisis terhadap kebutuhan dari berbagai macam aspek agar sistem yang dibuat menjadi solusi dalam masalah yang ditimbulkan sistem yang saat ini sedang berjalan, dan agar sistem yang dikembangkan sesuai dengan kebutuhan pengguna. Analisis kebutuhan ini dibagi menjadi dua bagian, yaitu analisis kebutuhan fungsional dan analisis kebutuhan non-fungsional.

**3.1.3.1 Analisis Kebutuhan Fungsional**

Implementasi metode *Natural Language Processing* (NLP) pada Elasticsearch untuk pengembangan mesin pencari di *website e-commerce* digunakan untuk membantu pengguna untuk mendapatkan hasil pencarian yang sesuai harapan pengguna disaat barang yang dicari tidak ditemukan. Berdasarkan analaisis yang telah dijelaskan sebelumnya, maka fasilitas yang menjadi kebutuhan sistem adalah sebagai berikut :

1. Terdapat fitur pencarian *autocomplete* yang memberikan saran pencarian yang muncul secara otomatis saat pengguna mengetikan kata kunci pada mesin pencari.
2. Terdapat fitur pencarian *autocorrect*. Fitur ini akan memperbaiki ejaan kata kunci yang salah ketik yang dimasukan oleh pengguna mesin pencari.
3. Terdapat fitur pencarian *auotosuggestion* yang memberikan saran hasil pencarian yang lain sebagai pengganti hasil pencarian yang tidak dapat ditemukan.
4. Pencarian yang dilakukan dapat menggunakan kata kunci yang lebih dari satu kata.

**3.1.3.2 Analisis Kebutuhan Non-Fungsional**

Selain kebutuhan fungsional, kebutuhan non-fungsional juga harus terpenuhi. Kebutuhan non-fungsional meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan kebutuhan pengguna.

**3.1.3.2.1 Kebutuhan Perangkat Keras**

Spesifikasi perangkat keras yang digunakan dalam pengembangan sistem ini antara lain :

1. *Processor Intel Pentium Inside*
2. *Laptop Asus A43SD*
3. *RAM* 4 GB
4. *Hardisk* 500 GB
5. 2 BG *Graphic* *NVIDIA* *GeForce* 610M

**3.1.3.2.2 Kebutuhan Perangkat Lunak**

Spesifikasi perangkat lunak yang digunakan dalam mengembangkan sistem ini antara lain :

1. *Operating System Ubuntu* 14.04 LTS
2. *Text Editor Geany*
3. *Browser Google Crome*
4. *Ubuntu Terminal*

**3.1.3.2.3 Kebutuhan Pengguna**

Sistem ini tidak mebutuhkan pengguna dengan keterampilan khusus, cukuo dengan kemampuan menggunkan komputer dan internet untuk menggunakan mesin pencari.

**3.1.4 Sistem yang Diusulkan**

Berdasarkan masalah yang telah dijelaskan, maka diperlukan solusi yang tepat. Dari hasil analisis yang didapat, maka diusulkan implementasimetode *Natural Language Processing* (NLP) yang telah ada dalam Elasticsearch pada *website e-commerce* penjualan kosmetik. Berikut ini penjelasan dari sistem yang diusulkan.

**3.1.4.1 Gambaran Sistem yang Diusulkan**

Berdasarkan hasil analisis yang telah dilakukan, baik analisis permasalahan pada sistem yang telah berjalan saat ini dan analisis terhadap berbagai kebutuhan fungsional dan no-fungsional, didapatkan gambaran usulan sistem yang baru. Berikut ini sistem baru yang diusulkan :

**3.2 Perancangan**

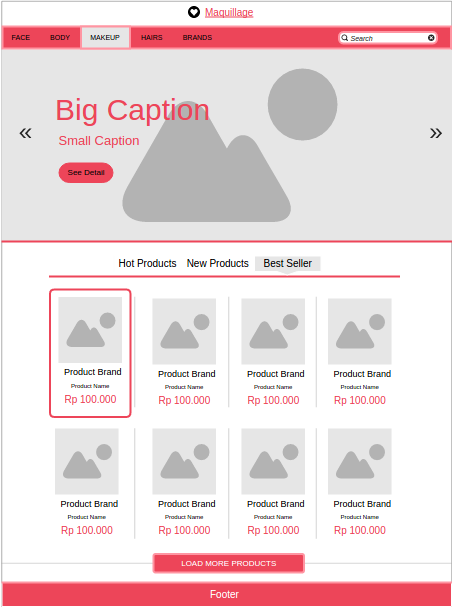
**3.2.1 Struktur Navigasi**

**3.2.2 Activity Diagram**

**3.2.3 Perancangan Antarmuka**

**3.2.3.1 Perancangan Halaman Utama**

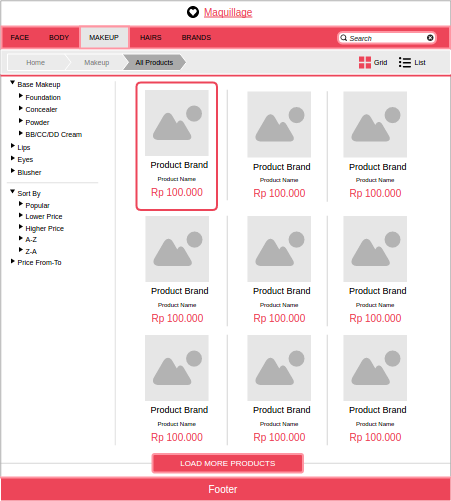
Pada Gambar 3.7 terlihat rancangan antarmuka halaman utama yang merupakan tampilan paling awal saat *website* dikunjungi. Pada halaman ini ditampilkan *header* halaman untuk menampilkan nama *website,* *menu* kategori produk yang dijual yang berdampingan dengan kotak pencarian, *slider* untuk menampilkan *caption* dari website, beberapa daftar produk yang dijual, dan bagian *footer* yang menampilkan tahun penciptaan *website*.



**Gambar 3.7 Rancangan Halaman Utama**

**3.2.3.2 Perancanagn Halaman *Product Grid***

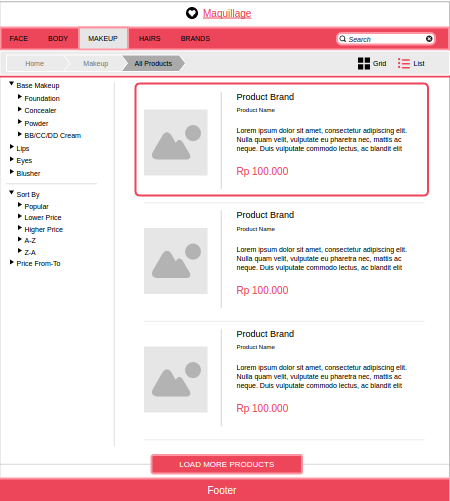
Pada Gambar 3.8 terlihat rancangan antarmuka halaman *product grid* yang merupakan tampilan yang akan muncul saat menu pilihan kategori produk dipilih. Halaman ini juga akan muncul saat menampilkan hasil pencarian. Pada halaman ini ditampilkan *header* halaman untuk menampilkan nama *website,* *menu* kategori produk yang dijual yang berdampingan dengan kotak pencarian, beberapa daftar produk yang dijual yang ditampilkan dalam kotak-kotak kecil, dan bagian *footer* yang menampilkan tahun penciptaan *website*.



**Gambar 3.8 Rancangan Halaman *Product Grid***

**3.2.3.2 Perancangan Halaman *Product List***

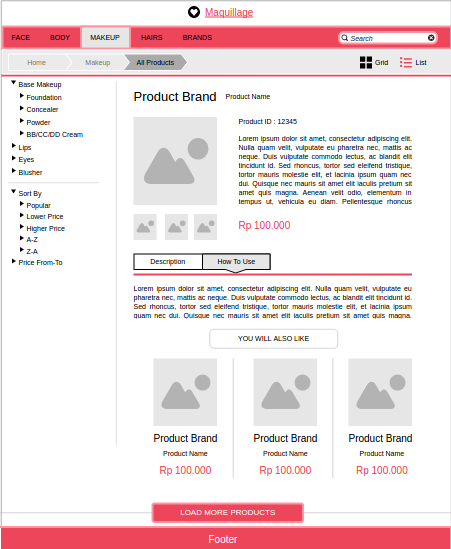
Pada Gambar 3.9 terlihat rancangan antarmuka halaman *product list* yang merupakan tampilan yang akan muncul saat menu pilihan kategori produk dipilih. Halaman ini juga akan muncul saat menampilkan hasil pencarian. Pada halaman ini ditampilkan *header* halaman untuk menampilkan nama *website,* *menu* kategori produk yang dijual yang berdampingan dengan kotak pencarian, beberapa daftar produk yang dijual yang ditampilkan dalam daftar yang disertai deskripsi produk secara singkat, dan bagian *footer* yang menampilkan tahun penciptaan *website*.



**Gambar 3.9 Rancangan Halaman *Product List***

**3.2.3.3 Perancangan Halaman *Product Detail***

Pada Gambar 3.10 terlihat rancangan antarmuka halaman *product detail* yang merupakan tampilan yang akan muncul saat salah satu hasil pencarian produk dipilih. Pada halaman ini ditampilkan *header* halaman untuk menampilkan nama *website,* *menu* kategori produk yang dijual yang berdampingan dengan kotak pencarian, gambar produk, nama merek, nama produk, deskripsiproduk, harga produk, cara penggunaan produk, dan bagian *footer* yang menampilkan tahun penciptaan *website*.



**Gambar 3.10 Rancangan Halaman *Product Detail***