**PENENTUAN SOLUSI PRODUK CACAT MENGGUNAKAN *CASE BASED REASONING* DI PT YAMATOGOMU INDONESIA**

**Dewi Ratnasari1, Tacbir Hendro Pudjiantoro2, Asep Id Hadiana3**

1,2,3)Universitas Jenderal Achmad Yani, Jl Terusan Jenderal Sudirman-Cimahi

Email : dewiratnasariprofil@gmail.com; tacbir23501027@yahoo.com; ahadiana@gmail.com

**Abstrak**

Pada perusahaan-perusahaan manufaktur proses produksi merupakan bagian penting, didalam proses produksi pasti mengalami kerusakan atau terdapat produk cacat akibat kondisi mesin,metode kerja dan lain-lain. Setiap perusahaan punya cara tersendiri untuk mengatasi produk cacat, biasanya apabila produk cacat parah maka produk tersebut dibuang dan apabila produk cacat tersebut masih bisa diperbaiki, maka akan diperbaiki dengan cara tertentu sesuai dengan kerusakan yang terjadi. Cara memperbaiki produk cacat tersebut didapat dari permasalahan produk cacat terdahulu serta pengetahuan dari *Quality Assurance* yang sudah berpengalaman dalam proses produksi.Tetapi permasalahannya adalah banyaknya staff *Quality Assurance* yang keluar sehingga menyebabkan pengetahuan dari perbaikan produk cacat tersebut sedikit hilang dan mengakibatkan proses penanganan yang banyak membutuhkan waktu. Maka dari itu dibutuhkan sebuah sistem berbasis penalaran menggunakan *Case Based Reasoning* dan penalaran kasus yang mirip menggunakan *vector space model* sebagai metodenya. Keluaran dari sistem tersebut berupa informasi solusi perbaikan produk cacat sesuai dengan masalah yang dimasukkan.

**Kata Kunci :** produk cacat, *quality assurance*, supervisor, PT Yamatogomu Indonesia, *vector space model,* *case based reasoning*.

**Abstract**

*In the manufacturing company process production is the important thing, in the production process curtain have flaw or there some damage product because of machine condition, work method, etc. every company have own tradition for repairing demage product, usually if the product is totally flaw then that product have to throw away and if the damage product is still can be repair then that product will be repaired with certain way dependent the kind of damage that happened. The way to repairing that damage product be found from previous problem of damage product and knowledge from Quality Assurance that have experienced in production prosess. But the problem is so many worker in Quality Assurance that resign so cause the ability from repairing that damage product is little lost and cause handling process that much need time. Because of that so needed a system have as a base reasoning use Case Based Reasoning and reasoning case that similar use Vector Space Model as a method. Result from that system is a information for solution repairing damage product dependent with the problem that inputed.*

**Keywords :***damage product, quality assurance, supervision, PT. Yamatogomu Indonesia, vector space model, case based reasoning.*

**PENDAHULUAN**

 Pengetahuan dari seorang ahli atau pakar akan sangat berguna bagi dunia pendidikan terutama pada bidangnya masing-masing maupun bagi seseorang pada bidang tersebut yang belum mempunyai banyak pengalaman dan pengetahuan. pada era globalisasi saat ini, sebuah industri dituntut untuk memberikan produk yang tidak cacat dan sesuai dengan spesifikasi (Windarti, 2014). Usaha peningkatan kualitas produk dilakukan dengan cara mengatasi penyebab cacat pada suatu proses produksi. Peningkatan dan pengendalian kualitas produksi memerlukan komitmen untuk perbaikan yang melibatkan antara faktor manusia (motivasi) dan faktor mesin (teknologi). Pengendalian Mutu Terpadu (*Total Quality Control*) sebagai pendekatan manajemen modern, dalam menjalankan suatu usaha untuk memaksimumkan daya saing perusahaan melalui perbaikan secara terus-menerus (*continous improvement*) atas produk atau bahan baku. Salah satunya adalah mengenai penangan masalah produk cacat, setiap produk cacat yang terjadi mempunyai solusi yang berbeda-beda. Namun, solusi tersebut didapat dari pengetahuan *Quality Assurance* dan kasus yang pernah terjadi sebelumnya. Pada objek kasus yang diteliti, dalam proses penentuan solusi atas kasus produk cacat baru dapat dilihat dari kasus-kasus sebelumnya,dikarenakan kasus sebelumnya berjumlah sangat banyak, sehingga proses pencarian pun membutuhkan waktu yang lama. Maka penelitian ini dibuat untuk membantu pihak *Quality Assurance* dalam mencari solusi atas kasus produk cacat yang sama. Contohnya pada diagnosa kerusakan sepeda motor matic. Sepeda motor matic sangatlah banyak jenisnya dan spesifikasi yang berbeda. Terkadang jumlah mekanik yang terbatas dan minimnya pengetahuan pengguna menyebabkan berbagai kesulitan dalam perawatannya khususnya dalam mengatasi kerusakan mesin (Putra, 2015).

 Adapun manfaat yang dirasakan oleh pihak PT Yamatogomu Indonesia selaku objek penelitian, terutama pada bagian *Quality Assurance* yang menggunakan sistem secara langsung, membuat pekerjaan menjadi lebih mudah dalam menentukan solusi dari perbaikan produk cacat dengan mencari kasus sebelumnya yang serupa. Selain itu memudahkan pihak *Quality Assurance* dalam mengelola kasus produk cacat yang baru atau kasus tersebut belum pernah terjadi sebelumnya. Seperti halnya dalam objek wisata di daerah Yogyakarta, Yogyakarta adalah tujuan wisata kedua setelah Bali dan merupakan surga wisata di pulau Jawa. Setiap tahunnya wisatawan yang berkunjung di Yogyakarta meningkat 7%-15%. Khususnya dimasa libur panjang ada banyak wisatawan yang berkunjung. Informasi tentang tujuan wisata dapat diperoleh melalui browsur atau katalog wisata, dari internet maupun cerita teman. Namun informasi saja tidaklah cukup jika tidak diramu dengan cepat untuk mendapat alternatif terbaik dalam menentukan tujuan wisata. Atas dasar kebutuhan tersebut maka dibuatlah perancangan aplikasi Case Based Reasoning untuk menentukan tujuan wisata pribadi di Yogyakarta berbasis mobile web (Dewi *et al*, 2012). Dalam proses Retrieve biasanya banyak digunakan metode Information Retrieval seperti Vector Space Model, TF-idf dll. *Retrieve* berfungsi untuk mencari kesamaan dokumen dengan dokumen terdahulu, metode similaritas selain digunakan untuk mencari kesamaan dokumen, dapat juga digunakan untuk mendeteksi plagiarisme dalam penelitian, dapat kita ketahui plagiarisme terkadang tidak diketahui karena jumlah penelitian yang banyak serta penelitian yang sudah lama dilakukan (Isa dan Abidin, 2013).

**TINJAUAN PUSTAKA**

***Produk Cacat***

 Dengan adanya pengalokasian produk rusak dan produk cacat secara tepat akan mempengaruhi ketepatan perhitungan harga pokok produksi yang selanjutnya juga mempengaruhi penetapan harga jual produk yang dihasilkan perusahaan. Faktor dari luar yang mempengaruhi penentuan harga jual antara lain yaitu pesaing, luas pasar dan sifat produk. Faktor dari dalam perusahaan yang juga mempengaruhi penetapan harga jual yaitu seperti biaya produksi dan biaya-biaya lain yang relevan (Wahyuni, 2015).

***Case Based Reasoning***

 Gambaran konseptual bagaimana kecerdasan buatan dapat dimanfaatkan dalam proses pembelajaran. *Case-based Reasoning* (CBR) merupakan bagian dari kecerdasan buatan menyediakan model pembelajaran pemecahan masalah. Pemecahan masalah dalam *Case-based Reasoning* dilakukan dengan cara menggunakan kembali pemecahan masalah sebelumnya yang memiliki kemiripan (similarity). Perkembangan *Case-based Reasoning* sangat dipengaruhi oleh ilmu kognitif, banyak penelitian telah membuktikan tentang keberhasilan *Case-based Reasoning* dalam pembelajaran. *Case-based Reasoning* mampu menjadi solusi alternatif dalam mengembangkan kemampuan pemecahan masalah pada siswa dalam pembelajaran (Gerhana *et al*, 2013). Siklus CBR yang dikemukakan oleh Aamodt dan Plaza diantaranya adalah:

1. *Retrieve*

 Mendapatkan/memperoleh kembali kasus yang paling menyerupai/relevan (*similar*) dengan kasus yang baru. Tahap *retrieval* ini dimulai dengan menggambarkan/menguraikan sebagian masalah, dan diakhiri jika ditemukannya kecocokan terhadap masalah sebelumnya yang tingkat kecocokannya paling tinggi. Bagian ini mengacu pada segi identifikasi,kecocokan awal, pencarian dan pemilihan serta eksekusi.

1. *Reuse*

Memodelkan/ menggunakan kembali pengetahuan dan informasi kasus lama berdasarkan bobot kemiripan yang paling relevan ke dalam kasus yang baru, sehingga menghasilkan usulan solusi dimana mungkin diperlukan suatu adaptasi dengan masalah yang baru tersebut.

1. *Revise*

Meninjau kembali solusi yang diusulkan kemudian mengujinya pada kasus nyata (simulasi) dan jika diperlukan memperbaiki solusi tersebut agar cocok dengan kasus yang baru.

1. *Retain*

Mengintegrasikan/menyimpan kasus baru yang telah berhasil mendapatkan solusi agar dapat digunakan oleh kasus-kasus selanjutnya yang mirip dengan kasus tersebut. Tetapi Jika solusi baru tersebut gagal, maka menjelaskan kegagalannya, memperbaiki solusi yang digunakan, dan mengujinya lagi.

***Vector Space model***

Pencarian informasi saat ini dilakukan dengan menggunakan mesin pencari yang ada pada situs layanan di dunia maya. Mesin pencari yang sudah ada dan banyak digunakan saat ini memberikan hasil informasi yang sangat banyak, sehingga diperlukan waktu untuk memilah-milah informasi yang dibutuhkan. Informasi yang didapatkan bila terlalu banyak akan menyulitkan user karena user juga mendapatkan informasi yang tidak berguna . Banyaknya informasi ini menjadikan waktu yang ada akan terbuang atau tidak efisien karena terlalu banyaknya informasi menyebabkan pekerjaan-pekerjaan yang tidak efektif dilakukan yaitu : mencari dan memilih.

*Vector Space Model (VSM)* adalah metode untuk melihat tingkat kedekatan atau kesamaan *(smilarity)* term dengan cara pembobotan term. Dokumen dipandang sebagi sebuah vektor yang memiliki *magnitude* (jarak) dan *direction* (arah). Pada *Vector Space Model,* sebuah istilah direpresentasikan dengan sebuah dimensi dari ruang vektor. Relevansi sebuah dokumen ke sebuah *query* didasarkan pada similaritas diantara vektor dokumen dan vektor *query* (Amin, 2011).

1. *Input*

Masukan dari sistem ini adalah data latih dan data uji. Data latih berupa data produk cacat terdahulu yang disimpan kedalam *knowledge-base*. Kemudian data uji adalah masalah produk cacat baru.

1. *Proses*

Sebuah kasus baru yang memiliki fakta kecacatan akan dicocokan dengan kasus terdahulu yang telah disimpan ke dalam basis kasus. fakta kecatatan pada kasus baru akan dimasukkan ke dalam proses *case base reasoning*. Pada konsep *case base reasoning*.

*Case Based Reasoning* (CBR) adalah proses dari penyelesaian kasus yang baru dengan solusi yang diambil dari kasus yang sama sebelumnya.

1. *Retrieve*

Pada tahap ini diambil dari kasus produk cacat baru kemudian dihitung tingkat kemiripan dengan kasus terdahulu yang ada dalam basis kasus terdahulu dengan menggunakan *Vector Space Model*, kemudian setelah mendapat nilai similaritas tertinggi, masuk ke tahap *Reuse*. Kemudian lanju ke tahap berikutnya yaitu Pra proses, diantara adalah *Tokenizing,* proses merubah dokumen menjadi kata-kata. *Filtering,* Proses ini berfungsi untuk membuang atau menghapus kata-kata yang tidak penting.*Stemming,* Proses ini berfungsi untukmerubah kata sambung menjadi kata dasar atau membuang imbuhan pada setiap kata.

Selanjutnya masuk ke tahap pembobotan menggunakan metode *Term Frequency Inverse Document Frequency.*Perhitungan bobot dengan *Term Frequency – Inverse Document Frequency* (TF-IDF) menggunakan kombinasi dari dua nilai, yaitu frekuensi kata dan *inverse* frekuensi dokumen yang didapat dari membagi jumlah dokumen secara keseluruan dengan jumlah dokumen dimana kata tersebut muncul. Berikut adalah rumus untuk mencari bobot dengan TF-IDF :

 (1)

dengan:

*tf d,t* = frekuensi banyak-nya kata ke-t dari kata kunci pada dokumen ke-d

 *dft* = jumlah dokumen yang mengandung kata ke-t dari kata kunci

 D = jumlah semua dokumen yang ada di dalam database

 *W d,t* = bobot dokumen ke-d terhadap kata kunci ke-t

Selanjutnya masuk ke tahap perhitungan Simlaritas menggunakan *Vector Space Model. Vector Space Model* adalah Model dalam IR yang berbasis token untuk memungkinkan partial maching dan pemeringkatan dokumen (pengindexan). Dengan prinsip dasar dokumen menjadi sebuah token yang kemudian dikumpulkan menjadi (n) token-token, kemudian Query menjadi vector token yang berfungsi untuk mencari token-token yang berhubungan dengan melihat kesamaan vektor dokumen dan query berdasarkan jarak vektor.

 (2)

Dengan |q| adalah Jarak query, dan Wiq adalah bobot query dokumen ke-i, maka Jarak query (|q|) dihitung untuk didapatkan jarak query dari bobot query dokumen (Wiq) yang terambil oleh sistem. Jarak query bisa dihitung dengan persamaan akar jumlah kuadrat dari query. Dengan |dj | adalah jarak dokumen, dan Wij adalah bobot dokumen ke-i, maka Jarak dokumen (|dj |) dihitung untuk didapatkan jarak dokumen dari bobot dokumen dokumen (Wij) yang terambil oleh sistem. Jarak dokumen bisa dihitung dengan persamaan akar jumlah kuadrat dari dokumen.

Perhitungan pengukuran Similaritas *query document (inner product)*, menggunakan persamaan

 (3)

Dengan Wij adalah bobot term dalam dokumen, Wiq adalah bobot query, dan Sim (q, dj) adalah Similaritas antara query dan dokumen. Similaritas antara query dan dokumen atau inner product/Sim (q, dj) digunakan untuk mendapatkan bobot dengan didasarkan pada bobot term dalam dokumen (Wij) dan bobot query (Wiq) atau dengan cara menjumlah bobot q dikalikan dengan bobot dokumen.

Pengukuran Cosine Similarity (menghitung nilai kosinus sudut antara dua vector) menggunakan persamaan

 (4)

Similaritas antara query dan dokumen atau Sim(q,dj) berbanding lurus terhadap jumlah bobot query (q) dikali bobot dokumen (dj) dan berbanding terbalik terhadap akar jumlah kuadrat q (|q|) dikali akar jumlah kuadrat dokumen (|dj| ). Perhitungan similaritas menghasilkan bobot dokumen yang mendekati nilai 1 atau menghasilkan bobot dokumen yang lebih besar dibandingkan dengan nilai yang dihasilkan dari perhitungan *inner product (*Wisnu dan Hetami, 2015).

1. *Reuse*

Pada tahapan ini, informasi yang terdapat dari hasi *case retrieve* akan digunakan kembali apabila terdapat kasus yang sama atau mendekati.

1. *Revise*

Pada ini solusi belum sesuai akan dilakukan revisi secara otomatis oleh sistem. Setelah solusi diperbaiki, maka solusi tersebut akan masuk ke dalam tahapan retain.

1. *Retain*

Pada tahapan ini adalah tahap menyimpan kasus baru sebagai pengetahuan baru.

1. *Output*

Output dalam sistem ini adalah dokumen yang mirip dengan nilai *similarity* tertinggi.

**METODE PENELITIAN**

Dalam penelitian ini algoritma/metode yang digunakan adalah metode vector space model dengan konsep *case based reasoning*. Adapun pada penelitian ini terdapat empat tahap utama,yaitu *retrieve*, mencari similaritas kasus masalah produk cacat dengan kasus sebelumnya. Kedua adalah *reuse*, *reuse* adalah menggunakan kembali solusi dari kasus sebelumnya untuk kasus yang sedang dicari solusinya. Yang ketiga adalah *revise, revise* adalah proses mengubah solusi atau membuat solusi baru apabila kasus tersebut belum pernah terjadi sebelumnya. Kemudian yang keempat adalah *retain,retain* adalah proses penyimpanan pada basis kasus atas perubahan solusi maupun menyimpan kasus yang baru. Untuk lebih jelasnya dapat dilihat pada Gambar 1.



**Gambar 1.** Gambaran Umum Sitem Penentuan Solusi Produk Cacat

**HASIL DAN PEMBAHASAN**

Model perancangan yang digunakan dalam penelitian ini adalah Unified Modeling Language. UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain system perangkat lunak, khususnya system yang dibangun menggunakan pemrograman berorientasi objek.

Perancangan Sistem adalah salah satu tahap dari Metodologi pengembangan sistem, dan merupakan salah satu bagian yang sangat penting di dalam itu.Tahap ini dilakukan oleh sistem designer yang melakukan interaksi dengan pengguna sistem, hasil desain dievaluasi oleh user dari sudut pandang kepentingan pemakai untuk kemudian diimplementasikan kembali oleh sistem designer (Rismawan dan Hartati, 2012).

*a. Business Actor*

*Business Actor* merupakan pemodelan yang menggambarkan aktor-aktor yang berperan dalam penggunaan sistem. Berdasarkan analisis yang telah dilakukan maka dalam sistem ini aktornya adalah admin yang merupakan pengguna utama. *Business actor* untuk sistem ini dapat dilihat pada Gambar 2.



**Gambar 2.** *Business Actor*

**Tabel 1.** Business Actor

|  |  |  |
| --- | --- | --- |
| No. | Aktor | Deskripsi |
| 1. | Quality Assurance | *Quality Assurance* adalah aktor yang menggunakan sistem. Memiliki hak akses dalam mengelola data produk cacat dan mengelola hasil penentuan produk cacat. |
| 2. | Supervisor | *Supervisor* Adalah Aktor Sekaligus Admin Yang Menggunakan Sistem. Memiliki Hak Akses Dalam Mengelola Solusi Baru Dan Persetujuan Atas Produk Cacat Yang Baru. |

1. *Use Case Diagram*

*Use Case Diagram* menggambarkan bagaimana aktor berinteraksi dengan sebuah sistem. *Usecase* dibuat berdasarkan analisa sistem yang sedang berjalan, *business actor* dan *business use case*. *Business actor* menggambarkan pengguna yang terdapat di dalam sistem, dan *business use case* menggambarkan fungsi-fungsi apa saja yang dapat dilakukan oleh setiap aktor di dalam sistem tersebut. Pada sistem ini terdapat 3 *use case* utama yaitu, kelola Produk cacat, kelola Solusi, kelola *Stopword*, kelola Token, dan kelola *user*.



**Gambar 3.** *Use Class Diagram*

1. *Class Diagram Conceptual*

*Class diagram conceptual* ini dibuat dengan memperhatikan *use case* dan skenario dari masing-masing *use case* pada diagram sebelumnya, maka terlihat beberapa struktur relasi antar informasi. Untuk memodelkan struktur informasi ini maka dapat digambarkan *diagram class conceptual.*



**Gambar 4.** Class Diagram Conceptual

1. *Class Diagram*

Diagram kelas merupakan kumpulan kelas-kelas objek yang memiliki atribut dan perilaku yang sama dimana setiap objek memiliki entintas masing-masing yang unik. Setiap kelas memiliki relasi tertentu dengan kelas lain. Pada sistem yang akan dibangun *class diagram* dibuat berdasarkan *class conceptual* yang telah didefinisikan, kemudian setiap operasi dibuat sesuai dengan *sequence diagram* pada masing-masing kelas. Adapun *class diagram* yang dirancang dapat dilihat pada Gambar 5.



**Gambar 5.** *Class Diagram*

Pada penelitian ini pembobot dokumen dilakukan dengan menggunakan metode *Term Frequency – Inverse Document Frequency (TF-IDF).* Sebelum melakukan proses pembobotan setiap dokumen harus melalui tahap text preprocessing yang terdiri dari *tokenizing, filtering dan stemming*. Adapun proses text preprocessing dapat dilihat pada Gambar 2, dan perhitungan TF-IDF dapat dilihat pada tabel 1. pada tabel, dokumen 1 sapai 5 adalah dokumen produk cacat terdahulu dan ditambah dengan 1 dokumen uji



**Gambar 6.** *Preprocessing*

1. *Hasil Perhitungan Vector Space Model*

Setelah melalui pra-proses dan pembobotan Term Frequency-inverse document frequency, dokumen tersebut akan dihitung similaritasnya menggunakan vector space model. Perhitungan dan hasil perhitungan dapat dilihat pada Tabel 2.

**Tabel 2.** Perhitungan vector space model

|  |  |  |  |
| --- | --- | --- | --- |
| kata | Kk | Tf | VSM |
| D1 | D2 | D3 | D4 | D5 | D1 | D2 | D3 | D4 | D5 |
| Robek | 0.488 | 0.488 | 0 | 0 | 0 | 0 | 0.238 | 0 | 0 | 0 | 0 |
| Lubang | 0.158 | 0.158 | 0 | 0 | 0 | 0.158 | 0.025 | 0 | 0 | 0 | 0.025 |
| Besar | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| compound | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Sesuai | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Cepat | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| matang | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Dapat | 0 | 0 | 0.158 | 0.049 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bari | 0.049 | 0 | 0.049 | 0.049 | 0 | 0.049 | 0 | 0.002 | 0.002 | 0 | 0.002 |
| Tinggal | 0 | 0 | 0.049 | 0.158 | 0 | 0.049 | 0 | 0 | 0 | 0 | 0 |
| Mesin | 0 | 0 | 0.158 | 0.158 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Press | 0 | 0 | 0.158 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Lalu | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| komulatif | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Cuci | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| abnormal | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Tebal | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| lengket | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Zigzag | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Cavity | 0.488 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0.238 | 0 | 0 |
| Sebab | 0 | 0 | 0 | 0.158 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Muka | 0.158 | 0 | 0 | 0.488 | 0 | 0.158 | 0 | 0 | 0.077 | 0 | 0.025 |
| Seal | 0.488 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0.238 | 0 | 0 |
| Grafik | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| rheometer | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| Baik | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| Hasil | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| Uji | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| Kurang | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| Akurat | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 | 0 |
| serabut | 0 | 0 | 0 | 0 | 0 | 0.488 | 0 | 0 | 0 | 0 | 0 |
| Celah | 0 | 0 | 0 | 0 | 0 | 0.486 | 0 | 0 | 0 | 0 | 0 |
| Jumlah | 3.089 | 3.504 | 3.016 | 3.419 | 1.392 | 0.263 | 0.002 | 0.557 | 0 | 0.052 |
| SQRT | 1.353 | 1.757 | 1.8721 | 1.736 | 1.8493 | 1.1799 | 0.513 | 0.0492 | 0.746 | 0 | 0.2292 |

Selanjutnya menghitung nilai *cosinus* sudut antara *vector* kata kunci dengan tiap dokumen pasal dengan menggunakan rumus dibawah ini:

* + - 1. Data Latih 1

= SUM(KK\*D1)/(sqrt(KK)\*sqrt(D1)

= 0,1108

* + - 1. Data Latih 2

= SUM(KK\*D2)/(sqrt(KK)\*sqrt(D2)

= 0,00096

* + - 1. Data Latih 3

= SUM(KK\*D3)/(sqrt(KK)\*sqrt(D3)

= 0,43705

* + - 1. Data Latih 4

= SUM(KK\*D4)/(sqrt(KK)\*sqrt(D4)

= 0

* + - 1. Data Latih 5

= SUM(KK\*D5)/(sqrt(KK)\*sqrt(D5)

= 0,03292

Sesuai perhitungan diatas maka nilai cosinus setiap dokumen telah didapat, seperti Tabel 3.

**Tabel 3.** Perhitungan Cosinus

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cosinus | D10,11088 | D20,00096 | D30,23705 | D40 | D50,03292 |
| Nilai Similarity | 0,11088 x 100% = 11,08% | 0,00096 x 100% = 0,09% | 0,43705 x 100% = 43,7% | 0 x 100% = 0% | 0,03292 x 100% = 3,29% |
|  | Rank 2 | Rank 4 | Rank 1 | Rank 5 | Rank 3 |

Dapat diketahui bahwa tingkat kemiripan ada pada dokumen 3 sebanyak 43,3%.

1. *Rancang Basis Data*

Sistem yang dibangun menggunakan basis data relasional sebagai media penyimpanan informasi dan data transaksi yang dikelola. Karena setiap data dibungkus dengan objek, maka dilakukan pemetaan dari class ke dalam tabel basis. Berdasarkan *Class Diagram* yang dibuat terdapat tabel relasional yang mewakili kelas objek pada sistem yang akan dibangun. Tabel relasional memungkinkan dibuat identik dengan kelas yang dirancang, namun setiap kelas tidak harus selalu dipetakan ke dalam satu buah tabel, karena tabel relasional yang dibangun disesuaikan dengan kebutuhan data yang akan disimpan. Adapun hasil pemetaan *class diagram* ke tabel relasional dapat dilihat pada tabel-tabel relasional dibawah ini. Berdasarkan perancangan class diagram, terdapat 4 kelas yang akan menjadi tabel karena memiliki atribut dan relasi dengan tabel lainnya, yaitu tabel Produk Cacat, token dan *stopword* dan *user*.

Tabel *User* merupakan table hasil pemetaan dari *class user*. Berisikan data *user* yang ditambahkan pada proses tambah *user*. Adapun atribut pada table *user* dapat dilihat pada Tabel 4.

Tabel 4. Tabel User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  Nama table | Atribut  | Type  | Size  | Keterangan |
| User | Username | Int | 25 | Primary key |
| Password | String | 8 |  |
| Level | String | 10 |  |

**KESIMPULAN**

 Kesimpulan yang dapat diambil bahwa dalam penentuan solusi produk cacat, proses pencarian secara otomatis dan pengelolaan solusi baru. Saran untuk aplikasi penentuan solusi produk cacat diharapkan dapat dikembangkan kembali dan dapat dijadikan bahan evaluasi untuk penelitian selanjutnya dengan menggunakan metode pembobotan dan similaritas yang lain.

**DAFTAR PUSTAKA**

Amin, F. 2011. Implementasi Search Engine (Mesin Pencari) Menggunakan Metode Vector Space Model. *Jurnal Ilmiah Dinamika Teknik Vol.5 No.1*, pp.45-58.

Dewi, E. K., Suyoto, dan Anindito, K. 2012. Analisis dan Perancangan Aplikasi Case Based Reasoning Untuk Menentukan Tujuan Wisata. *Seminar Nasional Informatika 2012*. pp.C-33-C-40.

Fowler, M. *UML Distilled Edisi 3*. Melrose: Penerbit Andi.

Gerhana, Y. A., Sudanyana, H. R., dan Budiman, T. 2013. Case-Based Reasoning (CBR) dan Pengembangan Kemampuan Penyelesaian Masalah*. Jurnal ISTEK Vol.7, No.1*. pp.210-224.

Isa, T. M. dan Abidin, T. F. 2013. Mengukur Tingkat Kesamaan Paragraf Menggunakan Vector Space Model untuk Mendeteksi Plagiarisme. *Seminar Nasional dan ExpoTeknik Elektro 2013*, pp. 229-234.

Karmayasa, I. B. M. Oka. 2012. *Implementasi vector space model dan beberapa notasi metode term frequency inverse document frequency* (tf-idf).

Putra, I. P. W. 2015. Sistem Pakar untuk Mendeteksi Kerusakan Sepeda Motor. *Konferensi Nasional Sistem & Informatika STMIK STIKOM 2015*, pp.478-483.

Rismawan, T. dan Hartati, S. 2012. Case-Based Reasoning untuk Diagnosa Penyakit THT (Telinga Hidung dan Tenggorokan). *Indonesian Journal of Computing and Cybernetics Systems, Vol. 6, No.2, p. 67-78*.

Wahyuni, F. I. 2015*. Analisis Perlakuan Produk Rusak dan Produk Cacat dalam Perhitungan Biaya Produksi untuk Menentukan Harga Jual Produk pada UD. Susana Baru (Skripsi).* Fakultas Keguruan dan Ilmu Pendidikan Universitas Nusantara Persatuan Guru Republik Indonesia*.*

Windarti, T. 2014*.* Pengendalian Kualitas untuk Meminimasi Produk Cacat pada Proses Produksi Besi Beton. *J@TI Undip Vol. 9, No.3*, pp.173-180.

Wisnu D. dan Hetami, A. 2015. Perancangan Information Retrieval (IR) Untuk Pencarian Ide Pokok Teks Artikel Berbahasa Inggris dengan Pembobotan Vector Space Model. *Jurnal Ilmiah Teknologi dan Informasi ASIA. Vol. 9, No. 1*, p. 53-59.