

RANCANG BANGUN APLIKASI REKOMENDASI PEMBELIAN LAPTOP DENGAN METODE FUZZY DATABASE MODEL TAHANI BERBASIS WEB

Hendry Setiawan¹, Seng Hansun²

Program Studi Teknik Informatika, Universitas Multimedia Nusantara, Tangerang, Indonesia

Email: ¹hendrylam93@gmail.com, ²hansun@umn.ac.id

ABSTRACT

A laptop is a computer that looks compact, small, lightweight and easy to carry anywhere. But choosing a laptop is not an easy thing for consumers who want to have a laptop as per the specifications and budget of the consumer finance. Therefore the purpose of this research is to build a web-based application that can give recommendation to purchasers or consumers as needed. The application is built by using PHP programming language and implementing the Fuzzy Database Tahani model which is still using the standard relation of fuzzy logic, and also using MySQL database for the data storage. It is found that the Fuzzy Database Tahani model can be implemented on the application built on to give a correct recommendation for the consumers.

Keyword: *laptop, PHP, MySQL, Tahani Fuzzy Database Models, Fuzzy logic*

1. PENDAHULUAN

Seiring perkembangan zaman, *laptop* merupakan sebuah kebutuhan yang mendasar bagi masyarakat. Toko-toko yang menjual *laptop* sekarang menjual *laptop* hanya menggunakan rekomendasi dari konsumen-konsumen lain untuk penjualan mereka, karena sebagian besar masyarakat belum tahu soal spesifikasi *laptop* dan harga. Bagi mereka itu adalah hal yang sulit karena *laptop* sekarang memiliki harga dan spesifikasi bervariasi mulai dari spesifikasi yang kecil, sedang, dan bagus kemudian untuk harga juga dari murah, sedang, dan mahal. Oleh sebab itu, dibutuhkan sebuah teknologi dan informasi yang bisa menyediakan data *laptop* yang lengkap mengenai *laptop* yang akan dibeli oleh konsumen.

Pada Toko Ricky Computer yang terletak di Jl. Kimjung No.70 Parit III, Jebus-Bangka Barat, merupakan salah satu toko yang menjual *laptop*. Toko tersebut menyadari bahwa mereka membutuhkan sebuah aplikasi yang dapat membantu penjualan *laptop* mereka untuk tetap bisa bersaing terhadap toko-toko lain, yaitu sebuah aplikasi yang dapat membantu merekomendasikan *laptop* pada pembeli, karena mereka hanya

merekomendasikan *laptop* mereka dari konsumen-konsumen lain yang membeli *laptop* di toko mereka. Oleh karena itu mereka sedikit sulit mendapatkan konsumen atau pembeli yang ingin membeli *laptop* di toko mereka sesuai keinginan konsumen dalam hal rekomendasi *laptop* kepada pembeli.

Aplikasi yang akan dibangun serupa dengan Abdul Gani Putra Suratma [1] dengan judul “Penerapan Fuzzy Database untuk Rekomendasi Pembelian Laptop”. Dari penelitian tersebut, penulis membuat beberapa fitur yang utama yang menerapkan metode *Fuzzy Database* antara lain menggunakan kriteria harddisk, harga, berat, lcd, dan *memory* sebagai *variable* himpunan *fuzzy* dan pencarian data menggunakan OR dan AND untuk mencari nilai yang manakah akan direkomendasikan untuk pembeli. Dari akhir penulisan penelitian, penulis tersebut menyarankan untuk membangun aplikasi ini bisa menggunakan bahasa pemrograman web dan penambahan fitur yang masih kurang dari penulis.

Tujuan dari penelitian ini adalah merancang bangun sebuah sistem berbasis *web* dengan menerapkan *Fuzzy Database Model Tahani* untuk mempermudah konsumen dalam pemilihan *laptop* sesuai keinginan dan anggaran keuangan mereka.

2. LANDASAN TEORI

2.1 Konsep Logika Fuzzy

Teori *fuzzy* diperkenalkan pertama kali oleh Lotfi A. Zadeh pada tahun 1965, pada presentasinya mengenai *Fuzzy Sets* [2].

Sebelum munculnya logika *fuzzy*, dikenal sebuah logika tegas (*Crisp Logic*) yang memiliki nilai benar atau salah secara tegas. Sebaliknya Logika *Fuzzy* merupakan sebuah logika yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*) antara benar dan salah. Dalam teori logika *fuzzy* sebuah nilai bisa bernilai benar dan salah secara bersamaan namun berapa besar kebenaran dan kesalahan suatu nilai tergantung kepada bobot keanggotaan yang dimilikinya. Logika *fuzzy* adalah suatu cara yang tepat untuk

memetakan ruang *input* ke dalam suatu ruang *output* [2].

Himpunan tegas (*crisp*) *A* didefinisikan oleh *item-item* yang ada pada himpunan itu. Jika $a \in A$, maka nilai yang berhubungan dengan *a* adalah 1. Namun jika $a \notin A$, maka nilai yang berhubungan dengan *a* adalah 0. Notasi $A = \{x|P(x)\}$ menunjukkan bahwa *A* berisi *item x* dengan *p(x)* benar. Jika χ_A merupakan fungsi karakteristik *A* dan properti *P*, maka dapat dikatakan bahwa $P(x)$ benar, jika dan hanya jika $\chi_A(x)=1$. [3]

Himpunan *Fuzzy* didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan *real* pada interval $[0,1]$. Nilai keanggotaannya menunjukkan bahwa suatu *item* dalam semesta pembicaraan tidak hanya berada pada 0 atau 1, namun juga nilai yang terletak diantaranya. Dengan kata lain, nilai kebenaran suatu *item* tidak hanya benar atau salah. Nilai 0 menunjukkan salah, nilai 1 menunjukkan benar, dan masih ada nilai-nilai yang terletak antara benar dan salah [2].

Perbedaan mendasar logika tegas dengan logika *fuzzy* adalah nilai keluarannya. Logika tegas hanya memiliki dua nilai *output* yaitu 0 atau 1, sedangkan logika *fuzzy* memiliki nilai antara 0 sampai 1. Logika *fuzzy* memiliki banyak nilai keluaran yang dikenal dengan derajat keanggotaannya [3].

Himpunan *fuzzy* memiliki dua atribut, yaitu :

- Linguistik, yaitu sebuah variabel yang memiliki nilai berupa kata-kata dalam bahasa alamiah bukan angka, misalnya sedang, tinggi, rendah.
- Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 25, 30.

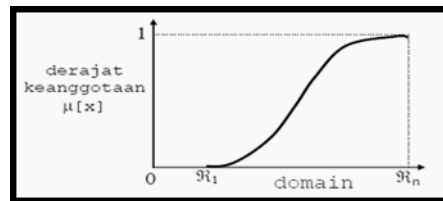
2.2 Fungsi Keanggotaan

Fungsi Keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah melalui pendekatan fungsi. Ada beberapa fungsi yang bisa digunakan yaitu sebagai berikut : [3]

1. Representasi Kurva-S

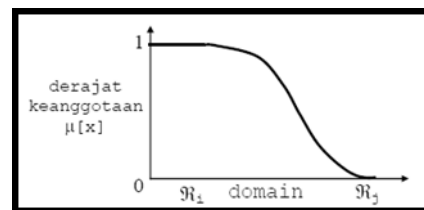
Kurva PERTUMBUHAN dan PENYUSUTAN merupakan kurva-S atau *sigmoid* yang berhubungan dengan kenaikan dan penurunan permukaan secara tak linear.

Kurva-S untuk PERTUMBUHAN akan bergerak dari sisi paling kiri (nilai keanggotaan = 0) ke sisi paling kanan (nilai keanggotaan = 1). Fungsi keanggotaannya akan tertumpu pada 50% nilai keanggotaannya yang sering disebut dengan titik infleksi (Gambar 1).



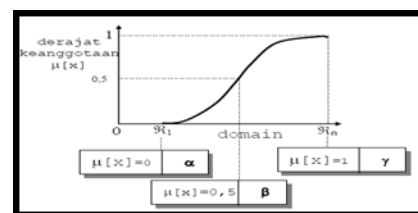
Gambar 1. Himpunan *Fuzzy* dengan Kurva-S: Pertumbuhan [3]

Kurva-S untuk PENYUSUTAN akan bergerak dari sisi paling kanan (nilai keanggotaan = 1) ke sisi paling kiri (nilai keanggotaan = 0) (Gambar 2).



Gambar 2. Himpunan *Fuzzy* dengan Kurva-S: Penyusutan [3]

Kurva-S didefinisikan dengan menggunakan 3 parameter, yaitu: nilai keanggotaan nol (α), nilai keanggotaan lengkap (γ), dan titik infleksi atau *crossover* (β) yaitu titik yang memiliki domain 50% benar. Gambar 3 menunjukkan karakteristik kurva-S dalam bentuk skema.

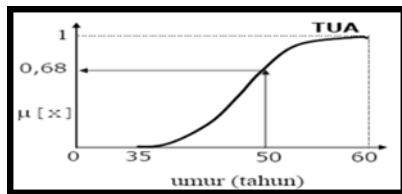


Gambar 3. Karakteristik Fungsi Kurva-S [3]

Fungsi keanggotaan untuk kurva PERTUMBUHAN adalah seperti berikut (Rumus 1):

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 & \rightarrow x \leq \alpha \\ 2((x - \alpha) / (\gamma - \alpha))^2 & \rightarrow \alpha \leq x \leq \beta \\ 1 - 2((\gamma - x) / (\gamma - \alpha))^2 & \rightarrow \beta \leq x \leq \gamma \\ 1 & \rightarrow x \geq \gamma \end{cases}$$

Contoh fungsi keanggotaan untuk himpunan TUA pada variabel umur terlihat seperti pada Gambar 4.



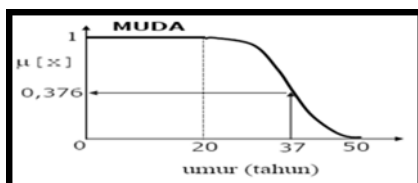
Gambar 4. Himpunan Fuzzy: Tua [3]

$$\begin{aligned} \mu_{TUA}[50] &= 1 - 2((60-50)/(60-35))^2 \\ &= 1 - 2(10/25)^2 \\ &= 0,68 \end{aligned}$$

Sedangkan fungsi keanggotaan pada kurva PENYUSUTAN adalah seperti pada (Rumus 2) berikut :

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 1 \rightarrow x \leq \alpha \\ 1 - 2((x - \alpha)/(\gamma - \alpha))^2 \rightarrow \alpha \leq x \leq \beta \\ 2((\gamma - x)/(\gamma - \alpha))^2 \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

Contoh fungsi keanggotaan untuk himpunan MUDA pada variabel umur terlihat seperti pada Gambar 5.

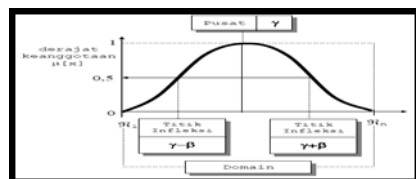


Gambar 5. Himpunan Fuzzy: Muda [3]

$$\begin{aligned} \mu_{MUDA}[50] &= 2((50-37)/(50-20))^2 \\ &= 2(13/30)^2 \\ &= 0,376 \end{aligned}$$

2. Representasi Kurva-BETA

Kurva BETA berbentuk lonceng didefinisikan dengan dua parameter, yaitu nilai pada domain yang menunjukkan pusat kurva (γ), dan setengah lebar kurva (β) (Gambar 6).



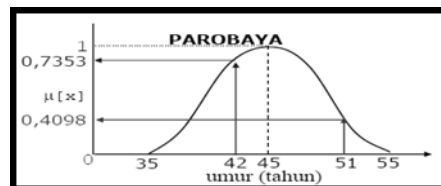
Gambar 6. Karakteristik Fungsi Kurva-Beta [3]

Fungsi keanggotaan pada kurva BETA adalah seperti pada persamaan 3 berikut:

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 \rightarrow x \leq \alpha \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \alpha \leq x \leq \beta \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

Fungsi keanggotaan untuk himpunan PAROBAYA pada *variable* umur seperti terlihat pada Gambar 7.

$$\begin{aligned} \mu_{PAROBAYA}[42] &= 1/(1 + ((42-45)/5)^2) \\ &= 0,7353 \\ \mu_{PAROBAYA}[51] &= 1/(1 + ((51-45)/5)^2) \\ &= 0,4098 \end{aligned}$$



Gambar 7. Himpunan Fuzzy : Parobaya dengan Kurva-Beta [3]

2.3 Operator Himpunan Fuzzy

Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasikan dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi dua himpunan sering dikenal dengan nama *fire strength* atau α - predikat. Ada tiga operator dasar yang diciptakan oleh Zadeh [4], yaitu:

1. Operator AND

Operator ini berhubungan dengan operasi interseksi pada himpunan. α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$$

2. Operator OR

Operator ini berhubungan dengan operasi union pada himpunan. α - predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

3. Operator NOT

Operator ini berhubungan dengan operasi komplemen pada himpunan. α - predikat sebagai hasil operasi dengan operator NOT diperoleh

dengan mengurangi nilai keanggotaan elemen pada himpunan yang bersangkutan dari 1.

$$\mu_A = 1 - \mu_A[x]$$

2.4 Fuzzy Database Model Tahani

Basis data *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya [5]. Model Tahani tersusun atas empat tahapan untuk implementasinya yaitu:

1. Menggambarkan Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki *interval* antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah pendekatan fungsi. Beberapa fungsi yang dapat digunakan yaitu Representasi kurva Linier, Representasi Kurva Segitiga, Representasi Kurva Trapesium, Representasi Kurva Bentuk Bahu, Representasi Kurva-S, Representasi Kurva Bentuk Lonceng (*Bell Curve*). Masing-masing fungsi tersebut, akan menghasilkan nilai antara "0" dan "1" dengan cara yang berbeda, sesuai dengan jenis representasi yang digunakan.

2. Fuzzifikasi

Fuzzifikasi adalah fase pertama dari perhitungan *fuzzy*, yaitu perubahan nilai tegas ke nilai *fuzzy*. Prosesnya adalah sebagai berikut: Suatu besaran analog dimasukkan sebagai masukan (*crisp input*), lalu *input* tersebut dimasukkan pada batas *scope* dari *membership function*. *Membership function* ini biasanya dinamakan *membership function input*. Keluaran dari proses fuzzifikasi ini adalah sebuah nilai *input fuzzy* atau yang biasanya dinamakan *fuzzy input*.

3. Fuzzifikasi Query

Fuzzifikasi *Query* diasumsikan sebuah *query* konvensional (*nonfuzzy*) *Database Management System* yang akan mencoba membuat dan menerapkan sebuah sistem dasar logika *fuzzy query*

4. Operator Dasar Zadeh untuk Operasi Himpunan Fuzzy.

Nilai keanggotaan sebagai dari dua himpunan *fuzzy* dikenal dengan nama *Fire Strength* atau α -predikat. Sangat mungkin digunakan operator dasar dalam proses *query* berupa operator AND dan OR.

α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-

himpunan yang bersangkutan, dinotasikan : $\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$. Sedangkan untuk hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan-himpunan yang bersangkutan, dinotasikan : $\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$.

Alternatif yang direkomendasikan adalah alternatif yang memiliki nilai *Fire Strength* atau tingkat kesesuaian dengan kriteria pilihan di atas angka 0 (nol) sampai dengan angka 1 (satu).

3. PERANCANGAN

Aplikasi yang dirancang mengimplementasikan teknik *fuzzy database* untuk mendapatkan informasinya menggunakan *Fuzzy Database Model Tahani* yaitu sebuah logika *fuzzy* yang dicoba untuk membuat logika *fuzzy* menggunakan *query Database Management (DBMS)*.

Pertama-tama aplikasi akan dirancang dengan beberapa proses yaitu proses *Home*, Tentang Kami, Rekomendasi, Daftar *Laptop*, Admin, dan *Search*. Dari beberapa proses tersebut *Home* dan Tentang Kami hanya menampilkan informasi-informasi tentang *laptop*, toko atau *programmer* dan tidak terkoneksi dengan *database*. Kemudian untuk proses Daftar *Laptop*, menampilkan daftar *laptop* yang ada pada Toko Ricky Computer sesuai *data* yang diberikan oleh pemilik toko tersebut dan proses ini terkoneksi dengan *database* guna untuk menyimpan *data* dan manajemen *laptop*, jika terdapat *data laptop* baru yang akan masuk ke toko tersebut. Kemudian pada proses *Search*, proses ini untuk mencari *laptop* ketika *guest* memasukkan *input*, lalu *input* tersebut akan diproses dan dicari di *database*. Jika *laptop* yang dicari sesuai *data laptop* pada *database* maka akan ditampilkan hasil pencarian sesuai dengan yang di-*input* oleh *guest*.

Proses Rekomendasi akan dimulai ketika *guest* memilih diantara rekomendasi yang ada pada aplikasi, antara lain Rekomendasi *Input* dan Rekomendasi Pilih ketika salah satu dari rekomendasi tersebut dipilih dan *guest* memilih rekomendasi atau meng-*input* rekomendasi maka akan diproses.

Pada Proses rekomendasi menggunakan metode *Fuzzy Database Model Tahani*, lalu mengambil *input* atau pilihan rekomendasi *guest* untuk diproses pada penentuan himpunan *fuzzy*, ketika proses penentuan himpunan *fuzzy* selesai, maka akan dilanjutkan ke proses Penentuan Fungsi Keanggotaan. Pada proses ini *input* atau pilihan dari rekomendasi dihitung dalam proses perhitungan fungsi keanggotaan untuk mendapatkan nilai pada tiap-tiap kurva dan akan dimasukkan ke dalam *database* untuk menyimpan

hasil perhitungan tersebut. Kemudian dilanjutkan ke proses Perhitungan *Fire Strength*. Proses ini untuk mendapatkan hasil dan ditampilkan kepada *guest* hasil rekomendasi *laptop* yang sesuai dengan *input* atau pilihan yang *guest* masukkan.

Kemudian untuk proses Admin dimulai dengan meng-*input form login* dan admin harus mengisi *username* dan *password* guna memproses *login* ke dalam fitur yang ada pada admin. Jika terjadi kesalahan *username* atau *password* maka akan kembali ke *form login*. Ketika berhasil *login* kedalam sistem maka akan diberikan beberapa proses lagi untuk admin, yaitu Manage Master Data dan Manage Himpunan Fuzzy. Kedua proses tersebut untuk berfungsi untuk *me-manage* data yang ada pada *database*.

4. HASIL PENGUJIAN

Uji coba dilakukan untuk melihat hasil analisa dari metode *Fuzzy Database Model Tahani*. Uji coba ini dilakukan dengan beberapa variabel pilihan dan angka dari rekomendasi pilih dan rekomendasi *input*.

Dimulai dari Rekomendasi Pilih dengan variabel pilihan Harddisk Sedang, Harga Murah, dan Processor Rendah cara mendapatkan hasil dari Rekomendasi Pilih ini diambil langsung dari *database* yang telah dibuat, sehingga tidak perlu lagi untuk menghitung kurva-kurva fungsi keanggotaan, karena hasil dari perhitungan kurva tersebut telah disimpan di dalam *database*, oleh karena itu untuk mencari hasil rekomendasi laptopnya tinggal menggunakan Operator Dasar Zadeh yaitu OR.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

Hasil pencariannya nilai terbesar atau disebut dengan nilai *fire strength*-nya dari tiga kriteria yang telah dipilih dapat dilihat pada (Tabel 1).

Tabel 1. Hasil Uji Coba Menghitung Rekomendasi Pilih

Kode Laptop	Harddisk Sedang	Harga Murah	Processor rendah	Nilai
LP001	0.57310 178104 49455	0.4288938 89239530 4	0.31999 999999 999984	0.5731 01781 04494 55
LP002	0.97052 385328 76598	0.9705949 95448779 1	0	0.9705 94995 44877 91

LP003	0.97052 385328 76598	0.4859764 13391114 6	0	0.9705 23853 28765 98
LP004	0.97052 385328 76598	0.9768982 37643384 8	0.03919 999999 999994	0.9768 98237 64338 48
LP005	0.97052 385328 76598	0.8657941 01128828 6	0.68000 000000 00002	0.9705 23853 28765 98
LP006	0	0.5445460 29641033 4	0.68000 000000 00002	0.6800 00000 00000 02
LP007	0.97052 385328 76598	0.9716982 73784715 7	1	1
LP008	0.97052 385328 76598	0.9091658 07795861 2	0.07999 999999 999996	0.9705 23853 28765 98
LP009	0.97052 385328 76598	0.5268429 47814684 1	0.02000 000000 000003 5	0.9705 23853 28765 98
LP010	0.97052 385328 76598	0.3674594 27789595 7	0.31999 999999 999984	0.9705 23853 28765 98
LP011	0.83276 220517 59983	0.0367983 65439352 575	0.02000 000000 000003 5	0.8327 62205 17599 83
LP012	0.41522 909413 27023	0	0	0.4152 29094 13270 23
LP013	0.97052 385328 76598	1	0.5	1
LP014	0.97052 385328 76598	0.8751434 88480399	0.07999 999999 999996	0.9705 23853 28765 98
LP015	0	0.0603630 63216965 28	0	0.0603 63063 21696 528
LP0	0.41522	0	0.07999	0.4152

16	909413 27023		999999 999996	29094 13270 23
LP0 17	0.97052 385328 76598	0.9967030 07742685 5	0.31999 999999 999984	0.9967 03007 74268 55
LP0 18	0.97052 385328 76598	0.7936214 04211477 5	0.18000 000000 000005	0.9705 23853 28765 98
LP0 19	0.97052 385328 76598	0.9850564 89419754 5	0.68000 000000 00002	0.9850 56489 41975 45
LP0 20	0	0.2956080 10586514 7	0.07999 999999 999996	0.2956 08010 58651 47
LP0 21	0.97052 385328 76598	0.8973734 96409157 3	0.07999 999999 999996	0.9705 23853 28765 98
LP0 22	0.97052 385328 76598	0.9557102 30748264 3	0	0.9705 23853 28765 98
LP0 23	0.97052 385328 76598	0.8221841 70721985 9	0.02000 000000 000003 5	0.9705 23853 28765 98
LP0 24	0.97052 385328 76598	0.9557102 30748264 3	0	0.9705 23853 28765 98
LP0 25	0.97052 385328 76598	0.9982718 82193500 4	0.02000 000000 000003 5	0.9982 71882 19350 04
LP0 26	0.54096 772691 28673	0.8608959 50344880 5	0.18000 000000 000005	0.8608 95950 34488 05
LP0 27	0.97052 385328 76598	0.7406743 21662143 6	0.18000 000000 000005	0.9705 23853 28765 98
LP0 28	0	0.1410732 52763690 48	0.31999 999999 999984	0.3199 99999 99999 984

LP0 29	0.25310 161802 7079	0.1800413 38921606 87	0.92	0.92
LP0 30	0.41522 909413 27023	0	0	0.4152 29094 13270 23
LP0 31	0.97052 385328 76598	0.9091658 07795861 2	0	0.9705 23853 28765 98
LP0 32	0.97052 385328 76598	0.9310703 59163112 2	0.5	0.9705 23853 28765 98
LP0 33	0.97052 385328 76598	0.9820613 10237161 1	0	0.9820 61310 23716 11
LP0 34	0.25310 161802 7079	0.5576018 41536085 2	0.68000 000000 000002	0.6800 00000 00000 02
LP0 35	0.97052 385328 76598	0.8357208 01019625 3	0	0.9705 23853 28765 98

Dari hasil yang diperlihatkan pada tabel 1 dapat disimpulkan bahwa perhitungan *fire strength* menggunakan Operator Zadeh OR mendapatkan dua hasil dari sekian banyak rekomendasi *laptop* dan ini terbukti hasilnya sama persis dengan hasil rekomendasi pilih pada sistem aplikasi rekomendasi *laptop* dengan fitur Rekomendasi Pilih.

Kemudian untuk uji coba untuk Rekomendasi *Input* fitur ini sedikit berbeda dengan Rekomendasi Pilih, karena Rekomendasi *Input* yang di-*input* oleh *guest* atau admin berupa angka bukan variabel linguistik, jadi harus dihitung terlebih dahulu untuk mengubah nilai tegas menjadi nilai *fuzzy* menggunakan fungsi keanggotaan. Rekomendasi input akan dicoba dengan inputan pada Harddisk 500 GB , dan LCD 16 Inch. Angka-angka tersebut akan dihitung terlebih dahulu dimulai dari kurva-s penyusutan, lalu kurva-beta loceng, setelah itu kurva-s pertumbuhan, lihat di bawah ini perhitungan kurva-kurva tersebut.

Dimulai dari Harddisk 500 GB untuk kurva-s penyusutan:

$$\alpha = 60, \quad \beta = 301, \quad \gamma = 542, \quad x = 500$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 1 \rightarrow x \leq \alpha \\ 1 - 2((x - \alpha)/(\gamma - \alpha))^2 \rightarrow \alpha \leq x \leq \beta \\ 2((\gamma - x)/(\gamma - \alpha))^2 \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

$$\begin{aligned} \mu_{\text{Harddisk}[500]} &= 2((542-500)/(542-60))^2 \\ &= 2(42/482)^2 \\ &= 0,01518569 \end{aligned}$$

Harddisk 500 GB untuk kurva-beta lonceng:

$$\alpha = 60, \quad \beta = 241, \quad \gamma = 542, \quad x = 500$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 \rightarrow x \leq \alpha \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \alpha \leq x \leq \beta \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

$$\begin{aligned} \mu_{\text{Harddisk}[500]} &= 1/(1 + ((500-542)/241)^2) \\ &= 0,9705 \end{aligned}$$

Harddisk 500 GB untuk kurva-s pertumbuhan :

$$\alpha = 542, \quad \beta = 783, \quad \gamma = 1024, \quad x = 500$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 \rightarrow x \leq \alpha \\ 2((x - \alpha)/(\gamma - \alpha))^2 \rightarrow \alpha \leq x \leq \beta \\ 1 - 2((\gamma - x)/(\gamma - \alpha))^2 \rightarrow \beta \leq x \leq \gamma \\ 1 \rightarrow x \geq \gamma \end{cases}$$

$$\mu_{\text{Harddisk}[500]} = 0$$

Setelah penghitungan kurva telah selesai dari hasil ketiga kurva tersebut di hitung lagi *fire strength*-nya menggunakan Operator Dasar Zadeh yaitu OR.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

$$\text{Hasil} = \max((0,01518569), (0,9705), (0))$$

$$\text{Hasil} = 0,9705$$

Lalu di-*query* untuk mencari nilai batasan harddisk pada tabel hasil himpunan *fuzzy* lihat pada (Tabel 2) hasil *query* dari MySQL.

Tabel 2. Hasil Uji Coba *query* pada tabel hasil himpunan *fuzzy*

Kode Laptop	Harddisk
LP001	0.5731017810449455
LP002	0.9705238532876598
LP003	0.9705238532876598
LP004	0.9705238532876598
LP005	0.9705238532876598
LP007	0.9705238532876598
LP008	0.9705238532876598
LP009	0.9705238532876598
LP001	0.5731017810449455
LP010	0.9705238532876598
LP011	0.8327622051759983
LP012	0.669289440608805
LP013	0.9705238532876598
LP014	0.9705238532876598
LP016	0.669289440608805

LP017	0.9705238532876 598
LP030	0.6692894406088 05
LP031	0.9705238532876 598
LP032	0.9705238532876 598
LP033	0.9705238532876 598
LP034	0.9601935228387 941
LP035	0.9705238532876 598
LP018	0.9705238532876 598
LP019	0.9705238532876 598
LP021	0.9705238532876 598
LP022	0.9705238532876 598
LP023	0.9705238532876 598
LP024	0.9705238532876 598
LP025	0.9705238532876 598
LP026	0.5409677269128 673
LP027	0.9705238532876 598

Setelah menghitung Harddisk sekarang giliran untuk menghitung LCD 16 Inch untuk kurva-s penyusutan:

$$\alpha = 11.6, \quad \beta = 12.7, \quad \gamma = 13.8, \quad x = 16$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 1 - 2((x - \alpha)/(\gamma - \alpha))^2 \rightarrow \alpha \leq x \leq \beta \\ 2((\gamma - x)/(\gamma - \alpha))^2 \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

$$\mu_{LCD[16]} = 0$$

LCD 16 Inch untuk kurva-beta lonceng:

$$\alpha = 11.6, \quad \beta = 1.1, \quad \gamma = 16, \quad x = 16$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 \rightarrow x \leq \alpha \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \alpha \leq x \leq \beta \\ 1/(1 + ((x - \gamma)/\beta)^2) \rightarrow \beta \leq x \leq \gamma \\ 0 \rightarrow x \geq \gamma \end{cases}$$

$$\mu_{LCD[16]} = 0$$

LCD 16 Inch untuk kurva-s pertumbuhan:

$$\alpha = 13.8, \quad \beta = 14.9, \quad \gamma = 16, \quad x = 16$$

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 \rightarrow x \leq \alpha \\ 2((x - \alpha)/(\gamma - \alpha))^2 \rightarrow \alpha \leq x \leq \beta \\ 1 - 2((\gamma - x)/(\gamma - \alpha))^2 \rightarrow \beta \leq x \leq \gamma \\ 1 \rightarrow x \geq \gamma \end{cases}$$

$$\mu_{LCD[16]} = 1$$

Setelah penghitungan kurva telah selesai dari hasil ketiga kurva tersebut di hitung lagi *fire strength*-nya menggunakan Operator Dasar Zadeh yaitu OR.

$$\mu_{A \cup B} = \max(\mu_A[x], \mu_B[y])$$

$$\text{Hasil} = \max((0), (0), (1))$$

$$\text{Hasil} = 1$$

Lalu di-*query* untuk mencari nilai batasan lcd pada tabel hasil himpunan *fuzzy* lihat pada (Tabel 3) hasil query dari MySQL.

Tabel 3. Hasil Uji Coba *query* pada tabel hasil himpunan *fuzzy*

Kode Laptop	LCD
LP001	0.968000053710 9219
LP002	0.968000053710 9219
LP003	0.828766990410 3412

LP004	0.968000053710 9219
LP005	1
LP006	1
LP007	1
LP008	0.968000053710 9219
LP009	0.828766990410 3412
LP010	0.968000053710 9219
LP011	1
LP012	0.851239643627 1378
LP013	1
LP014	0.968000053710 9219
LP015	0.968000053710 9219
LP016	0.828766990410 3412
LP017	0.968000053710 9219
LP018	0.968000053710 9219
LP019	1
LP020	0.968000053710 9219
LP021	0.968000053710 9219
LP022	0.968000053710 9219
LP023	1
LP024	0.968000053710 9219
LP025	0.968000053710 9219
LP026	1
LP027	0.968000053710 9219
LP028	0.968000053710 9219
LP029	1
LP030	0.828766990410

	3412
LP031	0.968000053710 9219
LP032	0.968000053710 9219
LP033	1
LP034	0.828766990410 3412
LP035	0.828766990410 3412

Ketika hasil batasan nilai dari kedua *Input* telah ditemukan pada *query database* maka dari hasil *query* pada tabel uji coba tersebut diiris dan dihitung lagi nilai *fire strength* dengan menggunakan Operator Dasar Zadeh, yaitu OR serta membatasi dengan input Harddisk <= 500 GB dan LCD <= 16 pada *query* hasil rekomendasi *input*. Hasil rekomendasi input dapat dilihat pada (Tabel 4).

Tabel 4. Hasil Uji Coba *query* hasil rekomendasi input

Kode Laptop	Nilai
LP013	1
LP023	1
LP005	1
LP007	1
LP019	1
LP033	1
LP006	1

Dari hasil yang diperlihatkan pada tabel 4, dapat disimpulkan bahwa pada rekomendasi *input* terbukti sama dengan hasil rekomendasi dari aplikasi yang dibangun yaitu fitur rekomendasi *input*.

Kemudian untuk hasil kuesioner yang didapatkan adalah tiga puluh tiga orang yang merespon kuesioner. Kuesioner ini dibuat melalui *google docs* dan disebarakan melalui media sosial *facebook* dan *blackberry messenger*, hasil dari kuesioner tersebut dapat dilihat pada Tabel 5 dan Tabel 6.

Tabel 5. Hasil Kuesioner untuk pilihan Rekomendasi Aplikasi atau Orang

Pilihan	Jumlah Orang	Persentase
Memilih Aplikasi	14	42%
Memilih Orang	12	36%
Jawaban lain atau Tidak menjawab	7	21%

Lalu untuk kriteria-kriteria *laptop* untuk aplikasi rekomendasi yang memilih cukup atau belum dapat dilihat pada (Tabel 6).

Tabel 6. Hasil Kuesioner untuk kriteria-kriteria *laptop* Cukup atau Belum

Pilihan	Jumlah Orang	Persentase
Kriteria <i>Laptop</i> Cukup	11	33%
Kriteria <i>Laptop</i> Belum	7	21%
Jawaban Lain atau Tidak Menjawab	15	45%

Dari hasil kuesioner tersebut masih banyak orang belum mempercayai sebuah aplikasi rekomendasi karena bagi mereka mungkin aplikasi ini belum ada yang mencobanya dalam dunia teknologi informasi, tetapi di sisi lain yang mengerti soal spesifikasi *laptop* tetap ingin adanya aplikasi rekomendasi ini karena aplikasi bisa langsung dicoba oleh pembeli sehingga, toko atau penjual itu sendiri tidak perlu lagi memberitahukan manakah *laptop* yang sesuai atau cocok untuk keperluan pembeli karena aplikasi rekomendasi ini sudah menyediakan data-data *laptop* yang lengkap sesuai dengan *laptop* yang ada di toko komputer tersebut.

Kemudian untuk kriteria-kriteria *laptop* dari hasil kuesioner yang mencakup pada aplikasi yang telah dibuat hasilnya adalah bahwa untuk merekomendasi *laptop* kepada pembeli menggunakan kriteria-kriteria processor, harddisk, memory, lcd dan harga masih kurang dari apa yang diharapkan, oleh karena itu mungkin bisa ditambahkan kriteria-kriteria *laptop* yang masih kurang dalam aplikasi yang telah dibangun.

5. SIMPULAN

Setelah melakukan penelitian dan melakukan uji coba serta menyebarkan kuesioner, bahwa aplikasi rekomendasi *laptop* untuk merekomendasikan *laptop* kepada pembeli ini dapat dibangun dan diimplementasikan dengan baik dalam menyelesaikan masalah yang ada pada Toko Ricky Computer. Aplikasi yang dibangun dapat diimplementasikan dengan baik menggunakan metode *Fuzzy Database Model Tahani* untuk menentukan rekomendasi *laptop* yang sesuai untuk pembeli pada Toko Ricky Computer.

DAFTAR PUSTAKA

- [1] Suratma, Abdul G.P., 2012, "Penerapan Fuzzy Database untuk Rekomendasi Pembelian Laptop", <http://id.scribd.com/doc/99772642/Fuzzy-Database-Tahani>. Diakses pada tanggal 4 Maret 2014.
- [2] Kusumadewi, S., 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu, Yogyakarta.
- [3] Suratma, Abdul G.P., 2012, "Penerapan Fuzzy Database Untuk Rekomendasi Pembelian Laptop", <http://id.scribd.com/doc/99772642/Fuzzy-Database-Tahani>. Diakses pada tanggal 4 Maret 2014.
- [4] Kusumadewi, S., 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu, Yogyakarta.
- [5] Kusumadewi, S., and Purnomo, H., 2010, *Aplikasi Logika Fuzzy Sistem Pendukung Keputusan*, Graha Ilmu, Yogyakarta.