

Prediksi Understandability Pada CK Metrics Object Oriented Menggunakan Fuzzy Inference System

¹⁾ Ahmad Fauzi

Universitas Bina Sarana Informatika, Jl. Kramat Raya no. 98, Jakarta Pusat, DKI Jakarta, Indonesia
E-Mail: ahmad.aau@bsi.ac.id

²⁾ Hylenearti Hertiana

Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri Jakarta (STMIK Nusa Mandiri);
Jl. Jatiwaringin Raya No.2, Jakarta Timur, DKI Jakarta, Indonesia
E-Mail: hylenearti.hha@nusamandiri.ac.id

ABSTRACT

The ability of the software to easily understand it is very important, to evaluate the performance of a system or process a certain mechanism is needed. So that the achievement level of software development is obtained, the assessment of code quality and design in the metrics have different contributions to the quality properties of software design. The assessment of software characteristics is described in a model based on the Chidamber Kemerer (CK) Entropy Metrics design property indicator. Measure specific properties of the software by mapping them into numbers according to objective and well-defined measurement rules. To be able to help perform the software quality calculation process based on CK metrics on the eclipse equinox data by using the Mamdani fuzzy inference system, pointing the classification into low which can be interpreted as having good quality. And the value affirmation of the defuzzification method in this case is done by using the COA (center of area) method, to determine the crisp x value, obtained from the membership function formed from the composition process of all outputs.

Keyword : *fuzzy inference system Mamdani, software quality, chidamber kemerer metrics.*

PENDAHULUAN

Mengidentifikasi kualitas komponen perangkat lunak dapat menggunakan metrics yang secara tepat memprediksi dari komponen perangkat lunak yang masih dapat digunakan dan dikembangkan dengan baik [1]. Usabilitas metrics ini jika diidentifikasi dalam tahap desain dan dalam tahap penulisan kode program dapat membantu untuk mengurangi dan meningkatkan kualitas penggunaan kembali komponen perangkat lunak [2].

Dataset Chidamber Kemerer (CK) metrics entropy dari eclipse equinox dalam proses pengukuran Understandability terdiri dari tiga properti desain yaitu *Weighted methods per class* (WMC), *depth of inheritance tree* (DIT), dan *response for a class* (RFC), Akan digunakan untuk memperoleh analisis struktur komponen perangkat lunak berbasis *object oriented*. Kualitas perangkat lunak diinterpretasikan dalam parameter-parameter yang sesuai dengan CK. Metrics yang digunakan. Sehingga dapat ditentukan aplikasi yang telah berjalan dapat digunakan kembali dengan syarat mudah dipahami dari sisi desain dan pengkodeannya sehingga proses *development* dengan skala yang besar dengan tingkat kesalahan yang lebih kecil dapat dilakukan. Entropy metrics eclipse equinox yang digunakan

pada penelitian ini mengevaluasi bagian dari sub system yang terdapat pada eclipse.

Penelitian sebelumnya dilakukan oleh [3] mengevaluasi CK metrics untuk mengurangi cacat desain perangkat lunak, dengan menggunakan analisis regresi. Penelitian lain dilakukan oleh [4], penelitian tersebut fokus pada menganalisis setiap rilis perangkat lunak, namun tidak menganalisis subsistem dari perangkat lunak. Berdasarkan penelitian-penelitian sebelumnya maka perlu dilakukan penelitian lanjutan menggunakan metode selain analisa regresi dan tidak hanya menganalisa di proyek utama, menganalisis pada subsistem, serta menggunakan tiga *variable independent* CK metrics. Sehingga pada penelitian ini mengusulkan prediksi Understandability pada entropy eclipse equinox menggunakan pendekatan fuzzy yang menganalisa subsistem proyek dan menggunakan tiga *variable* yang menerapkan pendekatan fuzzy inference system Mamdani dengan memanfaatkan output data unsur pengamatan dari dataset entropy CK metrics yang telah tersedia untuk memprakirakan kualitas perangkat lunak.

BAHAN DAN METODE

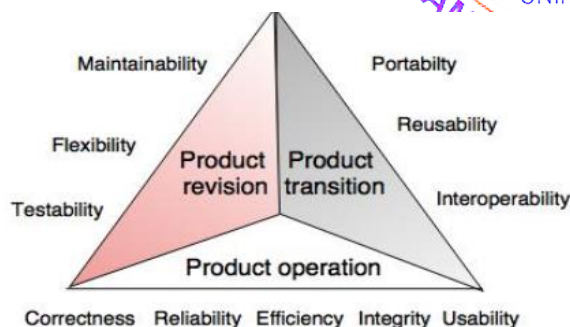
A. Metrics

Mengeksplorasi hubungan antara beberapa metrics desain *object oriented* (termasuk metrics

dari *suite* CK) dan sejauh mana perubahan kode, yang digunakan sebagai ukuran pengganti untuk upaya pemeliharaan. Pengukuran didefinisikan sebagai proses dimana nilai-nilai menjadi atribut dari entitas sehingga menggambarkan perangkat lunak sesuai dengan aturan yang jelas. Kualitas desain sebuah perangkat lunak sangat penting untuk dapat selalu diukur, dengan demikian setiap pengembang akan dapat mengetahui sejauh mana kualitas perangkat lunak dapat berjalan dengan baik. Dan menjelaskan mengenai seberapa tinggi tingkat kemudahan dalam memahami sebuah modul perangkat lunak yang dikembangkan [5], sehingga apabila ditemukan sebuah permasalahan akan cepat ditangani. Indikator penilaian dari pengukuran kualitas sebuah perangkat lunak dinamakan *metrics* digunakan untuk mengukur kualitas desain sebuah perangkat lunak (Glossary, 1990).

B. Properti Kualitas Desain Perangkat Lunak

Beberapa penelitian telah menghasilkan model kualitas perangkat lunak dan perspektif yang berbeda-beda. Salah satunya penelitian yang dilakukan oleh McCall dengan melihat dari tiga perspektif yaitu *product revision*, *product transition* dan *product operations* (Suresh, Pati, dan Rath 2012).[8]. Mengembangkan dari masing-masing perspektif menjadi beberapa faktor kualitas perangkat lunak yaitu *maintainability*, *flexibility*, *testability*, *portability*, *reusability*, *interoperability*, *correctness*, *reliability*, *efficiency*, *integrity* and *usability*. Sehingga total memiliki 11 faktor kualitas perangkat lunak.



Gambar 1. Desain Kualitas Perangkat Lunak

C. Properti Kualitas Desain Perangkat Lunak

Tujuh faktor kualitas *portability*, *reliability*, *efficiency*, *usability*, *testability*, *understandability* dan *flexibility*. ISO juga mengeluarkan model kualitas perangkat lunak yang dituangkan dalam ISO1926-1 dan memiliki enam karakteristik kualitas perangkat lunak *functionality*, *reliability*, *usability*, *efficiency*, *maintainability*, *portability* yang masing-masing dipecah dalam beberapa sub karakteristik. Kualitas yang dapat dievaluasi untuk mengukur kualitas kode dan desain yaitu: *efficiency*, *complexity*, *understandability*, *reusability* dan *maintainability* dan *testability*. Masing-masing parameter dalam *metrics* memiliki

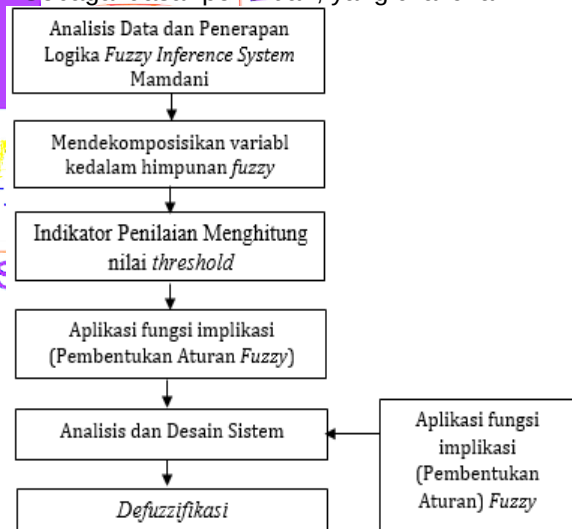
kontribusi yang berbeda pada properti kualitas desain perangkat lunak yang dapat dilihat pada Tabel 1.

Tabel 1. Properti kualitas perangkat lunak dan parameter metrics

Properti Kualitas Perangkat lunak	Metrics
<i>Efficiency</i>	LOCM, COB, DIT, NOC
<i>Complexity</i>	CC (Traditional Metrics)
<i>Understandability</i>	WMC, RFC, DIT
<i>Reusability</i>	WMC, LCOM, COB, DIT, NOC
<i>Maintainability</i>	WMC, RFC, DIT, NOC
<i>Testability</i>	

D. Metode Penelitian

Untuk mendapatkan informasi mengenai kualitas perangkat lunak penulis melakukan pengujian data metrics dari kode program berorientasi objek dengan melakukan pendekatan *fuzzy inference system* Mamdani, sehingga dapat diketahui tingkat kualitas dari sebuah perangkat lunak, adapun dataset yang digunakan adalah *entropy eclipse equinox* sebagai sumber aplikasi *open source*. kerangka penelitian digambarkan dalam Gambar 2. Sebagai dasar penelitian, yang dilakukan.



Gambar 1. Kerangka Penelitian Penelitian

1. Teknik Analisis Data

Teknik analisis data merupakan Cara menganalisa data penelitian untuk menjawab permasalahan yang ada dan menguji hipotesa. Martin Fowler (2004:28). Pada penelitian ini proses penilaian kualitas rekayasa perangkat lunak dari data *entropy eclipse equinox*, digunakan tiga parameter pengukur properti desain yaitu: WMC (*Weight Methods per Class*), RFC (*Response sets For a Class*), dan DIT (*Dept of Inheritance*). Sebagai proses analisis kualitas *metrics*, pengujian dilakukan dengan menggunakan pendekatan *fuzzy inference system* Mamdani.

2. Instrumen Pengujian Nilai *Threshold*

Instrumen pengujian berupa dataset yang akan digunakan untuk melakukan pengukuran dengan tujuan menghasilkan data kuantitatif yang akurat. Ada beberapa referensi yang digunakan pada penelitian dalam memilih nilai *threshold* CK *metrics*. Dapat dilihat pada Tabel 2. Menunjukkan nilai ambang batas yang digunakan oleh (Goyal et al., 2013). Sementara nilai ambang batas penulis gunakan Dapat dilihat pada Tabel 2. Diambil dari dataset yang ada.

Table 2. Nilai Threshold

CK Metrics	Threshold
WMC	0 – 15
RFC	0 - 35
DIT	0 - 6

3. Logika fuzzy inference system

Dalam teknik pendekatan *fuzzy inference system* Mamdani meliputi beberapa tahapan, dua tahapan pertama merupakan tahap pengumpulan data dan dua tahapan selanjutnya merupakan tahap analisis data [9]. Adapun tahap pengumpulan data terdiri dari model *understandability* masukkan dan fungsi keanggotaan serta tahapan untuk analisis data yang terdiri dari aplikasi fungsi implikasi, komposisi aturan dan *defuzzyfikasi*. Pada proses menganalisis data, peneliti menggunakan bantuan perangkat lunak MATLAB 2018b, sebagai sistem pengukuran *metrics*, tahapan ini menarik perhatian pada analisis data yang menghasilkan tiga tingkatan *low*, *midel* dan *high*. Sehingga akan diketahui tingkat kualitas dan *understandability* perangkat lunak.

a) Model Understandability

Fuzzy Inference System adalah pembuatan formulasi proses pemetaan *input* ke bagian *output* dengan menggunakan logika fuzzy. Tahapan utamanya meliputi pendefinisian fungsi keanggotaan *input* dan *output* serta pembuatan basis pengetahuan dengan menggunakan aturan (*rules*) *fuzzy if-then-else*. Berbagai kriteria inilah yang disebut sebagai variabel masukkan *fuzzy* [10]. Variabel masukkan dalam penentuan *understandability* dan kualitas perangkat lunak.

b) Pendefinisian Fungsi Keanggotaan Input Output

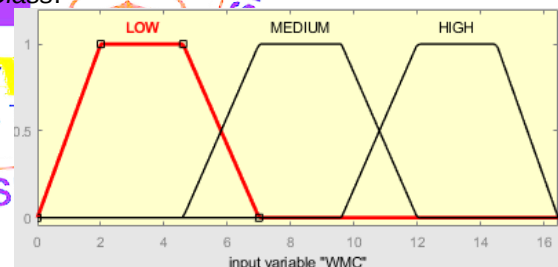
Variabel *fuzzy* dimodelkan adalah paramater *metrics* sebagai tiga variabel *input* sistem dan paramater *understandability* dan variabel *output* sistem. Setiap paramater *metrics* memiliki tiga himpunan *fuzzy*, yaitu *low*, *medium* dan *high* [11]. Begitu pula dengan paramater *output* memiliki tiga himpunan yang sama. Masing-masing himpunan dipetakan terhadap derajat keanggotaan dari 0 sampai dengan 1. Tipe dari fungsi keanggotaan untuk tiga variabel *input metrics*

adalah fungsi trapesium dengan sintaks seperti pada persamaan 1[12].

Kurva trapesium [12]. Tersebut adalah fungsi dari sebuah vektor, x yang tergantung pada empat paramater sakalar a , b , c dan d , sebagaimana diperlihatkan oleh persamaan (2), (3), (4) dan (5). Nilai a , b , c dan d disesuaikan dengan batasan himpunan keanggotaan yang dimiliki oleh setiap paramater *input metrics*. Kurva trapesium pada dasarnya menyerupai bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan satu. Representasi fungsi keanggotaan untuk kurva trapesium adalah sebagai parameter $\{a, b, c, d\}$ (dengan $a < b < c < d$) menentukan koordinat x pada empat sudut dari fungsi keanggotaan trapesium terdapat pada rumus 1.

$$\text{Trapesium} \begin{cases} a; & x \geq a \\ (x - a) / (b - a) & ; a \leq x \leq b \\ 1 & ; b \leq x \leq c \\ (d - x) / (d - c) & ; c \leq x \leq d \\ 0 & ; x \geq d \end{cases} \quad (1)$$

Untuk melengkapi fungsi trapesium tersebut, maka ditentukan nilai paramater a , b , c dan d untuk setiap variabel *input metrics*, Sementara itu bentuk kurva keanggotaan untuk setiap variabel *input CK metrics* diperlihatkan Gambar 3. sebagai contoh *input metrics* dari *Weight Methods per Class*.



Gambar 2. Weight Methods per Class

4. Aplikasi Fungsi Implikasi

Setelah pembentukan himpunan *fuzzy*, maka dilakukan pembentukan aturan *fuzzy* atau *rule*. Aturan-aturan dibentuk untuk menyatakan relasi antara *input* dan *output*. Tiap aturan merupakan suatu implikasi. Operator yang digunakan untuk menghubungkan antara beberapa *input* adalah operator *AND* dan yang memetakan antara *input-output* adalah *IF-THEN*. Aturan *fuzzy* dalam bentuk *if then* menggunakan model Mamdani dapat ditulis sebagai berikut:

$$\text{IF } (x_1 \text{ is } A_1) \text{ AND } (x_2 \text{ is } A_2) \dots \text{AND } (x_n \text{ is } A_n) \text{ THEN } y \quad (2)$$

5. Komparasi Aturan

Pada saat menghitung komposisi aturan fungsi *implikasi* menggunakan fungsi *MAX* yaitu dengan cara mengambil nilai maksimum dari *output* aturan kemudian menggabungkan daerah *fuzzy* masing-

masing aturan dengan operator *AND*. Secara umum dapat dituliskan:

$$\mu_{sf}[xi] \leftarrow \max(\mu_{sf}[xi], \mu_{kf}[xi]) \quad (3)$$

Paramater *input* CK *metrics* memiliki kecenderungan semakin memiliki nilai rendah (*low*) maka semakin baik kualitas. Semakin tinggi nilai *input* maka semakin buruk kualitas. Sehingga dalam aturan ini diberikan pembobotan tertinggi adalah saat kondisi *low* dan terendah saat kondisi *high*. Urutan pembobotan adalah *low* simpulan yang bersesuaian. Terdapat tiga *input* *metrics*, di mana setiap *input* memiliki tiga kemungkinan kondisi yaitu *Low*, *Medium*, *High*. Contoh aturan pertama adalah. *If (WMC is Low) and (DIT is Low) and (RFC is Low) then (Quality is Low)* berikut contoh terdapat pada gambar 3.

6. Proses Defuzzyfikasi

Proses *defuzzyfikasi* adalah mengubah fuzzy *output* menjadi nilai tegas berdasarkan fungsi keanggotaan yang telah ditentukan. simulasi dan analisa melalui perangkat lunak Matlab, maka aturan yang telah ditetapkan, selanjutnya melewati tahapan *defuzzyfikasi* dengan teknik *centroid*. Hasil *defuzzyfikasi* menampilkan kombinasi *input* *output* dalam kondisi awal. Nilai-nilai *input* yang ditampilkan mengambil nilai tengahnya masing-masing dan nilai *output* nya menunjukkan kualitas dengan kategori yang sesuai aturan dalam menghitung *defuzzyfikasi* digunakan metode *centroid method* dengan rumus sebagai berikut:

$$j^* = \frac{\sum_{j=1}^n Z_j \mu(Z_j)}{\sum_{j=1}^n \mu(Z_j)} \quad (4)$$

1). Metode Centre of Area (COA)

Metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah fuzzy, secara umum dirumuskan pada persamaan 2 untuk variabel kontinu dan persamaan 3 untuk variabel diskrit. Secara khusus misalkan *y*-p adalah pusat dari himpunan fuzzy ke-*p* dan *h*-p adalah tingginya, *center average defuzzier* menentukan *y** sebagai.

$$y^* = \frac{\sum_{p=1}^M y-p \cdot h_p}{\sum_{p=1}^M h_p} \quad (5)$$

HASIL DAN PEMBAHASAN

A. Analisis Data dan Penerapan Logika Fuzzy Inference System Mamdani

Tahap analisa data menggunakan logika fuzzy *inference system* model mamdani secara detail.

dataset yang dijadikan bahan penelitian yang pada *repository* <http://bug.inf.usi.ch> dari proyek *eclipse equinox* bagian dari *sub system* terapat 324 *record* *cllas* data.

B. Authors and Affiliations

Kriteria dan indikator yang akan dianalisis dan dijadikan variabel dalam melakukan proses penentuan kualitas perangkat lunak yang dinilai dari tiga *metrics* *object oriented*. Berdasarkan sampel penilaian kualitas perangkat lunak dari tiga properti desain (WMC, DIT dan RFC) akan menitik beratkan pada penentuan kualitas dan *understandability* pada dataset *entropy eclipse equinox*. Instrumen pengujian berupa dataset yang akan digunakan untuk melakukan pengukuran dengan tujuan menghasilkan data kuantitatif yang akurat. Dalam penelitian ini pengujian akan dilakukan terhadap *Entropy* CK *metrics* *object oriented*, sesuai dengan *threshold* CK *metrics*.

Table 3. Nilai Threshold

CK Metrics	Threshold
WMC	0 – 16.45
RFC	0 - 18.07
DIT	0 - 0.43

C. Figures and Tables

Setelah pembentukan himpunan fuzzy, maka dilakukan pembentukan aturan-aturan fuzzy. Tiap aturan merupakan suatu *implikasi*. Pada penelitian ini terdapat beberapa *rule*, Semua *rule* yang terbentuk. Khusus untuk penentuan kualitas perangkat lunak dan *understandability*.

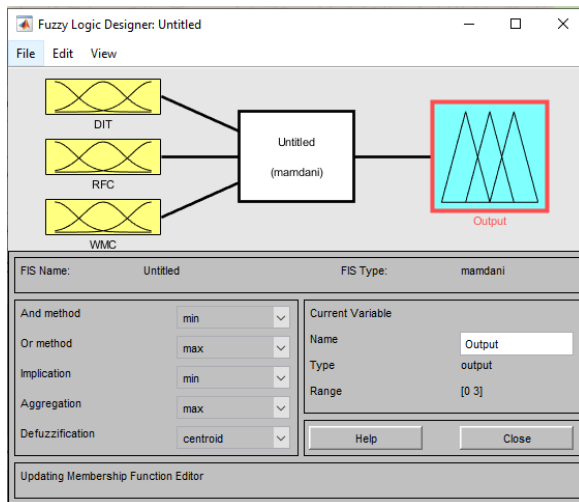
Tabel 3. Aturan Fuzzy

No	DIT	RFC	WMC	Hasil
1	Low	Low	Low	Low
2	Low	Low	Medium	Medium
3	Low	Low	High	Medium
4	Low	High	High	High
5	Medium	Low	Low	Medium
6	Medium	Low	Medium	Medium
8	Medium	Low	High	Medium
9	Medium	Medium	Medium	Medium
10	High	Medium	Low	Medium
11	High	Medium	Medium	High

Sumber: (Hasil Penelitian, 2020)

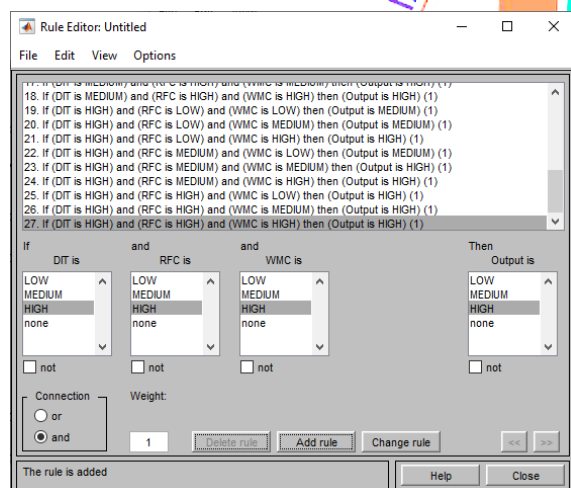
D. Analisis Dan Desain Sistem

- 1) Kriteria yang akan dianalisis dijadikan variabel fuzzy dari enam *metrics* dengan bantuan *Toolbox* MATLAB R2011b pada Gambar 4, dengan memiliki 6 *input* dan 1 *output*.



Gambar 3. Fuzzy Inference System Mamdani Editor. Sumber: (Hasil Penelitian, 2020)

- 2) *Rule Integritas* menjadi penentuan tingkat kualitas dan *reusability* akan diklasifikasikan kedalam empat tingkatan *bug* dari enam *metrics* yang diujikan.



Gambar 4. Rule Editor. Sumber: (Hasil Penelitian, 2020)

E. Proses Defuzzyfykasi

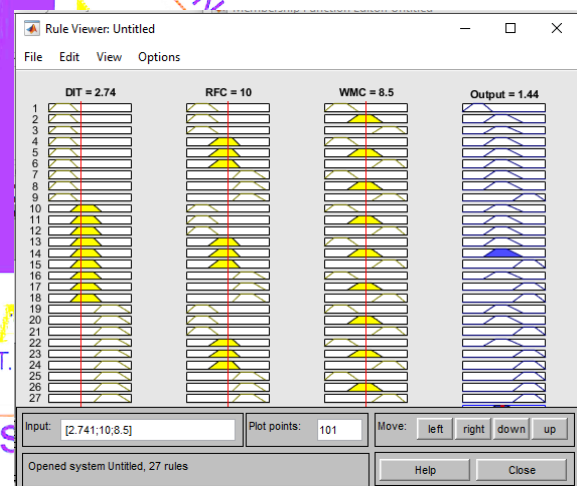
Metode *defuzzyfykasi* pada kasus ini dilakukan dengan menggunakan metode COA (*centre of area*), Bisektor, Mom (*Mean Of Maximum*), Lom (*Largest Of Maximum*), dan Som (*Smallest Of Maximum*), untuk menentukan nilai *crisp x*, didapat dari fungsi keanggotaan yang terbentuk dari proses komposisi semua *output*. Berikut akan dijabarkan hasil dari metode *defuzzyfikasi* tersebut

$$COA = \frac{(4,6 \times 0) + (7 \times 1) + (10 \times 1) + (12 \times 0)}{0 + 1 + 1 + 0}$$

$$\frac{0 + 7 + 10 + 0}{2} = 8,5$$

F. Pengujian

Pada tahap ini dilakukan pengujian untuk memperlihatkan perbandingan hasil *defuzzyfikasi* sesuai dengan beberapa metode yaitu metode COA (*center of area*), berikut pengujian metode *defuzzyfykasi* dengan MATLAB 2018b. Pada Gambar 4. yang berwarna kuning menandakan komposisi variabel masukkan yang nilainya bisa berubah secara manual dengan memasukkan nilai pada *field* masukkan yang berada di pojok kiri bawah. Hasil untuk keluaran dapat langsung ditampilkan berada di kotak kanan atas. Hasil *defuzzyfykasi* diatas akan diterapkan pada data pengujian. Berdasarkan sampel penilaian kualitas perangkat lunak dari enam properti desain (WMC, DIT dan RFC) akan menitik beratkan pada penentuan kualitas *understandability* mengenai seberapa tinggi tingkat kemudahan dalam memahami sebuah modul perangkat lunak yang dikembangkan berdasarkan dataset *eclipse equinox* data yang dijadikan bahan penelitian.



Gambar 5. FIS Model Mamdani Editor *Integrita*. Sumber: (Hasil Penelitian, 2020)

KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah penulis lakukan maka dapat diperoleh kesimpulan sebagai berikut:

1. *Fuzzy Inference System Mamdani*. sangat cocok digunakan untuk mengukur sebuah kelas dari tiga parameter desain perangkat lunak.
2. Kesalahan desain bisa diantisipasi dalam proses perancangan perangkat lunak mulai dari proses *development* hingga *implementasi*, *fuzzy inference system*.

Mamdani dapat digunakan untuk mengukur tingkat kualitas *understandability* perangkat lunak. tingkat kualitas yang baik memiliki angka yang relatif kecil, jika hasil perhitungan dari tiga *metrics* tergolong tinggi maka tingkat *understandability* sangat rendah, dan hasil dari pengujian yang dilakukan menggunakan MATLAB 2018b

menunjukkan bahwa nilai yang didapat berada pada titik medium dengan nilai output 1.44, dari 27 rols yang dibuat.

DAFTAR PUSTAKA

- [1] A. Shri, P. S. Sandhu, V. Gupta, and S. Anand, "Prediction of Reusability of Object Oriented Software Systems using Clustering Approach," vol. 4, no. 7, pp. 1174–1177, 2010.
- [2] A. Halim, A. Xandra, and A. Sim, "Mengukur Tingkat Reusability dan Efficiency dari Kode Program dengan Pendekatan Fuzzy Logic," vol. 4, pp. 109–114, 2015.
- [3] M. R. J. Qureshi, "Evaluation of the Design Metric to Reduce the Number of Defects in Software Development," no. April 2012, 2015, doi: 10.5815/ijitcs.2012.04.02.
- [4] I. Turnu, M. Marchesi, and R. Tonelli, "Entropy of the degree distribution and Object-Oriented Software Quality," no. June, 2012, doi: 10.1109/WETSoM.2012.6226997.
- [5] M. Nazir, R. A. Khan, and K. Mustafa, "A Metrics Based Model for Understandability Quantification," no. April, 2010.
- [6] A. September, "IEEE Standard Glossary of Software Engineering Terminology," vol. 121990, 1990.
- [7] Y. Suresh, J. Pati, and S. K. Rath, "Effectiveness of Software Metrics for Object-oriented System Effectiveness of software metrics for object-oriented system," no. December, 2012, doi: 10.1016/j.protcy.2012.10.050.
- [8] V. Axelius, K. Siren, N. Y. Setiawan, and R. I. Rokhmawati, "Evaluasi Kualitas Perangkat Lunak Menggunakan ISO / IEC 9126-4 Quality In Use (Studi Kasus : FILKOM Apps)," vol. 3, no. 2, pp. 1625–1632, 2019.
- [9] U. S. Utara and U. S. Utara, *Analisis Kinerja Fuzzy AHP dalam Perangkingan*. 2018.
- [10] P. K. Bhatia and R. Mann, "An Approach to Measure Software Reusability of OO Design An Approach to Measure Software Reusability of OO Design," no. January 2008, 2015.
- [11] M. Borys, "Metrics of Object Oriented Software," no. May, 2014.
- [12] J. Mago, "Analysis of Quality of the Design of the Object Oriented Software using Fuzzy Logic," pp. 21–25.