

Rancang Bangun Reusability Metric Tool pada Bahasa PHP

Aditya Wedha Utama¹, Dana Sulistyio Kusumo, Ph.D², Bayu Munajat, ST³

^{1,2,3}*Departemen Informatika, Fakultas Teknik, Universitas Telkom, Bandung, Indonesia*

¹awedha1993@gmail.com, ²02780291@ittelkom.ac.id, ³bayu1887@gmail.com

Abstrak— PHP atau Hypertext Preprocessor merupakan bahasa server-side scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. PHP adalah bahasa pemrograman yang paling banyak digunakan oleh programmer di masa sekarang .

Dalam pembangunan suatu aplikasi web, pengukuran kualitas merupakan hal pokok bagi disiplin rekayasa. Salah satunya adalah pengukuran metric reusability. Dalam pengembangan sebuah web diperlukan reusability dalam pengukuran kualitas sebuah pemrograman web. Reuse of software diharapkan dapat menghemat resource dan waktu pengembangan pemrograman web, serta menghasilkan modul dengan kualitas yang lebih tinggi. Namun, tool untuk menghitung nilai reusability sebuah kode bahasa pemrograman PHP belum tersedia. Karena itu penelitian ini berfungsi untuk memecahkan masalah tersebut.

Sistem ini, dibangun dengan identifikasi prosedur perhitungan metrik reusability terlebih dahulu. Kemudian untuk mencocokkan string teks, seperti karakter tertentu, kata-kata, atau pola karakter menggunakan metode regular expression di PHP[1]. Hasil perancangan sistem ini telah tervalidasi 100% oleh penguji ahli (tenaga pengajar mata kuliah yang bersangkutan atau yang telah mendapatkan sertifikasi OOAD). Selain itu, sistem ini mencapai keakuratan 100% dengan pengujian secara manual.

Keyword : reusability, PHP, OOP, Metric, kualitas software

I. PENDAHULUAN

PHP (PHP: Hypertext Preprocessor) adalah bahasa server-side scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis[2]. Karena PHP merupakan server-side scripting maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya dikirimkan ke browser dalam format HTML. PHP termasuk dalam Open Source Product, sehingga source code PHP dapat dirubah dan didistribusikan secara bebas. PHP dapat berjalan di sistem operasi apapun seperti Windows dan Linux. Selain itu, PHP merupakan bahasa pemrograman yang paling sering digunakan oleh programmer[3]. Karena, 21% perusahaan mencari PHP programmer ketimbang programmer bahasa pemrograman lain[2].

Sebagai programmer PHP, pengukuran kualitas perangkat lunak merupakan suatu hal pokok yang harus diperhatikan dalam pengembangan web[4]. Programmer harus dapat merancang perangkat lunak agar proyek dapat digunakan kembali (reusable) sehingga mudah dalam melakukan sebuah pengembangan. Reusability berkaitan dengan penggunaan kembali sebuah modul dari suatu proyek perangkat lunak ke

dalam pengembangan proyek perangkat lunak yang baru di masa mendatang[5]. Artinya, dalam pengembangan sebuah web diperlukan reusability dalam pengukuran kualitas sebuah pemrograman web. Reuse of software diharapkan dapat menghemat resource dan waktu pengembangan pemrograman web, serta menghasilkan modul dengan kualitas yang lebih tinggi. Kualitas yang lebih tinggi ini diasumsikan diperoleh berdasarkan hasil evaluasi terhadap modul tersebut ketika digunakan pada pemrograman web sebelumnya. Yang mana kesalahan dan kekurangan pada modul tersebut telah terdeteksi dan diperbaiki untuk digunakan pada perangkat lunak yang baru.

Dalam mengevaluasi performa suatu sistem atau proses diperlukan suatu mekanisme dan menentukan tingkat efisiensinya. Untuk setiap pengukuran yang dilakukan dibutuhkan tersedianya suatu ukuran kuantitatif yang disebut dengan metric. Metric Software merupakan tools yang bisa digunakan untuk mengukur kode yang kita buat dengan berbagai macam tipe pengukuran yang berhubungan dengan sistem, proses atau dokumen[6].

Pada tugas akhir ini, penulis merancang sebuah metric software yang menghitung reusability pada sebuah file PHP berbasis web. Penulis memilih bahasa pemrograman PHP karena reusability metric tool sebuah file PHP berdasarkan literature review yang penulis lakukan, belum pernah penulis temukan sampai sekarang. Padahal bahasa pemrograman PHP adalah bahasa pemrograman yang sering digunakan oleh programmer. Harapan yang ingin dicapai adalah programmer PHP bisa mengetahui kualitas web dan programmer PHP khususnya dari aspek reusability dapat meningkatkan kualitas web yang mereka bangun.

II. TINJAUAN PUSTAKA

A. PHP (Hypertext Preprocessor)

PHP (Hypertext Preprocessor) merupakan bahasa procedural, akan tetapi bahasa PHP dapat diperluas menjadi bahasa object-oriented[7]. PHP sudah mendukung object-oriented mulai di PHP versi 5, yang mempunyai bentuk pemodelan object yang baru untuk mengatasi kompleksitas masalah yang tidak bisa dilakukan pada versi sebelumnya.

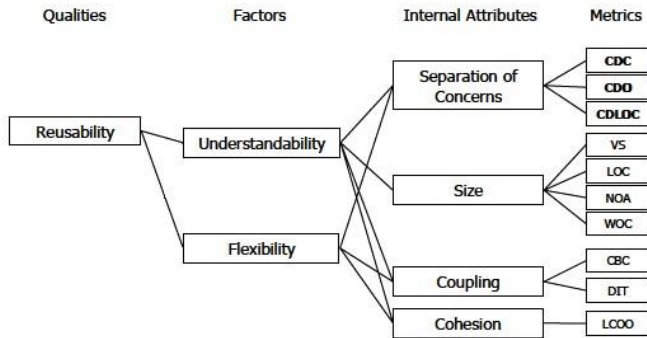
Pengerjaan kode pemrograman PHP dengan menerapkan konsep object-oriented mempunyai 2 keuntungan, yaitu[7]:

a. Code reusability, yaitu dengan menerapkan object-oriented suatu perintah yang kompleks dapat dipecahkan menjadi generic modules yang memudahkan untuk reuse of

code. Dengan file class yang terpisah dari skrip utama, maka kode dapat dengan mudah dikembangkan dalam proyek yang berbeda.

b. Easier maintenance and reliability, dengan menerapkan konsep object-oriented skrip dapat dengan mudah di maintenance. Karena, method pada setiap class berfokus pada satu perintah. Sehingga apabila ada kerusakan atau error pada satu perintah, maka dapat dengan mudah kita cari kesalahan dan memperbaikinya.

B. Reusability



Gambar 1 Reusability Model

Reusability adalah kemampuan dari elemen perangkat lunak untuk menyediakan konstruksi dari elemen yang berbeda di sistem perangkat lunak yang sama atau sistem perangkat lunak yang berbeda. Ada beberapa metrik yang digunakan, yaitu :

TABLE I METRIK REUSABILITY

No.	Reusability Metrics	
1	CDC	Menghitung jumlah <i>class</i> dan aspek yang mempunyai tujuan utama untuk berkontribusi pada implementasi <i>concern</i> dan jumlah dari <i>class</i> dan aspek lain yang mengaksesnya.
2	CDO	Menghitung jumlah <i>constructor</i> , <i>method</i> , dan <i>advice</i> yang mempunyai tujuan utama untuk berkontribusi pada implementasi <i>concern</i> dan jumlah dari <i>method</i> dan <i>advice</i> lain yang mengaksesnya.
3	CDLOC	Menghitung jumlah poin transisi untuk setiap <i>concern</i> melalui LOC.
4	CBC	Menghitung jumlah komponen yang terdeklarasikan di parameter formal, tipe pengembalian, deklarasi pelemparan, <i>class</i> , dan aspek.
5	DIT	Mengukur kedalaman hirarki pada suatu <i>class</i> .
6	LCOO	Mengukur kurang-kohesian pada sebuah <i>class</i> atau aspek dimana jumlah pasangan <i>method</i> dan <i>advice</i> yang tidak mengakses <i>variable instance</i> yang sama
7	VS	Menghitung jumlah komponen system (<i>class</i> , <i>abstract</i> , <i>interface</i> , dan aspek).
8	LOC	Menghitung jumlah <i>new line</i> di setiap <i>file</i> .
9	NOA	Menghitung jumlah atribut dari setiap <i>class</i> atau aspek.

10	WOC	Mengukur kompleksitas dari komponen dari operasi tersebut. Kompleksitas operasi didapat dari penghitungan jumlah parameter dari operasi.
----	-----	--

C. Regular Expression (Regex)

Dalam penelitian ini diperlukan source code dari aplikasi yang akan dihitung. Source code tersebut akan dimanipulasi sedemikian rupa sehingga memperoleh sebuah pattern. Permasalahan ini dapat diselesaikan dengan Metode Regular Expression atau regex. Karena, regex merupakan sebuah teknik yang digunakan untuk mencocokkan string teks, seperti karakter tertentu, kata-kata, atau pola karakter[13]. Regex memiliki 2 fungsi utama yakni mencari dan mengganti, mencari suatu pola tertentu dalam teks lalu menambah, mengurangi atau menggantinya menjadi pola yang lain [1].

Regex digunakan oleh banyak teks editor, utilities, dan bahasa pemrograman untuk pencarian dan memanipulasi teks berdasarkan pola. Misalnya, Perl, Ruby dan Tcl memiliki engine regular expression yang kuat dibangun pada sintaks mereka.

Berikut adalah sintaks regular expression pada Bahasa Pemrograman PHP:

TABLE II Sintaks Regular Expression

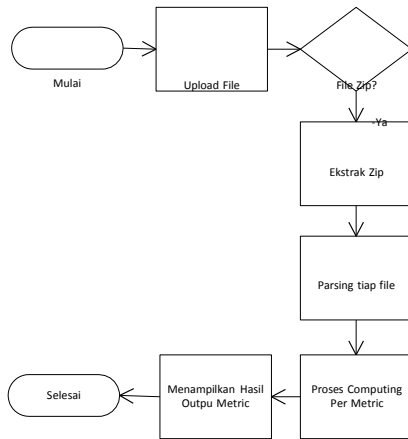
Regular Expression	Keterangan
Foo	Kata 'foo'
^foo	String diawali dengan kata 'foo'
Foo\$	String diakhiri dengan kata 'foo'
^foo\$	String 'foo' berdiri sendiri
[abc]	a,b, atau c
[a-z]	Semua huruf kecil
[^A-Z]	Semua huruf tanpa huruf besar
(gif jpg)	Cocokan gif atau jpg
[a-z]+	Minimal satu huruf kecil
[0-9.-]	Semua nomor, tanda baca titik ataupun tanda minus
^[a-zA-Z0-9_]{1,}\$	Semua kata minimal satu huruf, nomor atau _
([wx])([yz])	Wy,wz,xy atau xz
[^A-Za-z0-9]	Simbol apapun kecuali huruf ataupun nomor
([A-Z]{3}[0-9](4))	Mencocokkan tiga karakter dan empat angk

III. PERANCANGAN DAN PEMBANGUNAN

Aplikasi metric reusability tools merupakan sebuah web tools yang dibangun untuk para developer PHP. Nama aplikasi tersebut adalah Reusy. Web Tools ini membantu developer untuk menghitung nilai reusability dalam proyek website yang mereka bangun.

Web tools ini menerima sebuah file project PHP yang diberikan user, lalu mengkonversikan file tersebut menjadi sebuah nilai metric. Metric tersebut merupakan faktor yang

mempengaruhi sebuah nilai reusability pada sebuah file project. Berikut adalah flowchart reusability metric tool:



Gambar 2 Rancangan Sistem

A. Metric

1) Concern Diffusion over Component (CDC)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai CDC sebagai berikut:

TABLE III
Regex pada metrik CDC

Keterangan	Regex
Concern	/\$(\S+)[\s\n]*=[\s\n]*new (\S+)/
Class	/class[\s\n]+(\S+)[\s\n]*[extends]*[\s\n]*[a-zA-Z]*[\s\n]*{/

2) Concern Diffusion over Operations (CDO)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai CDO sebagai berikut:

TABLE IV
Regex pada Metrik CDO

Keterangan	Regex
Concern	/\$(\S+)[\s\n]*=[\s\n]*new (\S+)/
Class	/class[\s\n]+(\S+)[\s\n]*[extends]*[\s\n]*[a-zA-Z]*[\s\n]*{/
Invoke Variable	/[\$]this[->][w+][\s\n]*{/

3) Concern Diffusion over LOC (CDLOC)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai CDLOC sebagai berikut:

TABLE II
Regex pada Metrik CDLOC

Keterangan	Regex
Concern	/\$(\S+)[\s\n]*=[\s\n]*new (\S+)/
Invoke Variable	/[\$](w+)[\s\n]*{/

4) Coupling Between Componenets (CBC)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai CBC sebagai berikut:

TABLE II
Regex pada Metrik CBC

Keterangan	Regex
Memanggil Class	/\$(\S+)[\s\n]*=[\s\n]*new (\S+)/
Class	/class[\s\n]+(\S+)[\s\n]*[extends]*[\s\n]*[a-zA-Z]*[\s\n]*{/

5) Depth of Inheritance (DIT)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai DIT sebagai berikut:

TABLE II
Regex pada Metrik DIT

Keterangan	Regex
Class	/class[\s\n]+(\S+)[\s\n]*[extends]*[\s\n]*[a-zA-Z]*[\s\n]*{/

6) Lack of Cohesion in Operations (LCOO)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai LCOO sebagai berikut:

TABLE II
Regex pada Metrik CBC

Keterangan	Regex
Function	/(public private)[\s\n]+function[\s\n]+(\S+)[\s\n]*{/
Invoke Variable	[\$]this[->](w+)[\s\n]*; ;
Attribute	/(private public protected)[\s\n]+[\$](\S+)[\s\n]*;/

7) Vocabulary size (VS)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai VS sebagai berikut:

TABLE II
Regex pada Metrik VS

Keterangan	Regex
Class	/class[\s\n]+(\S+)[\s\n]*[extends]*[\s\n]*[a-zA-Z]*[\s\n]*{/

8) Lines of Code (LOC)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai LOC sebagai berikut:

TABLE II
Regex pada Metrik LOC

Keterangan	Regex
Comment	(?://)[^\r\n]*/*[\s\S]*?/*
WhiteSpace	/(^\r\n)*[\r\n+][\s\t]*[\r\n]+/

9) Number of Attributes (NOA)

Pola regex pada PHP yang digunakan untuk mendapatkan nilai NOA sebagai berikut:

TABLE II
Regex pada Metrik NOA

Keterangan	Regex
Attribute	/(private public protected)[\s\n]+[\$](\S+)[\s\n]*;/

10) *Weight of Class (WOC)*

Pola regex pada PHP yang digunakan untuk mendapatkan nilai WOC sebagai berikut:

TABLE II
Regex pada Metrik WOC

Keterangan	Regex
method	/(public private protected)* function[\s\n]+(\S+)[\s\n]*((\S+)*)/

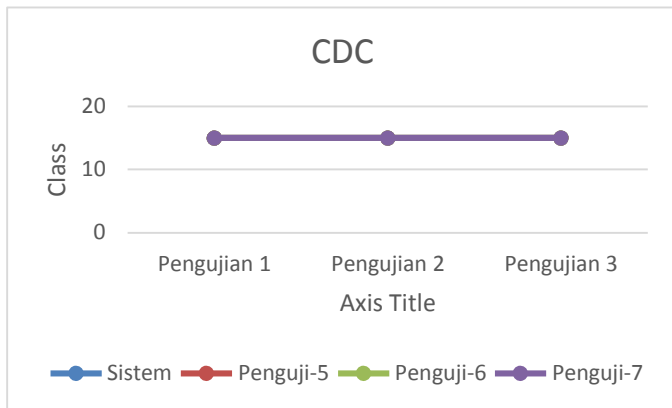
IV. HASIL DAN ANALISIS

Pada Gambar 1, dapat diketahui faktor dan metrik yang mempengaruhi reusability. Untuk memvalidasi bahwa sistem

yang dibangun, maka dilakukan suvey validitas terhadap penguji ahli dimana penguji ahli tersebut merupakan tenaga pengajar pada bidang yang terkait dan yang memiliki sertifikasi OOAD. Hasil total validitas tiap penguji mempunyai nilai yang sama dengan jumlah metrik didapatkan dalam penelitian ini. Sehingga, dapat dikatakan presentase validitas sistem yang dibangun adalah 100%.

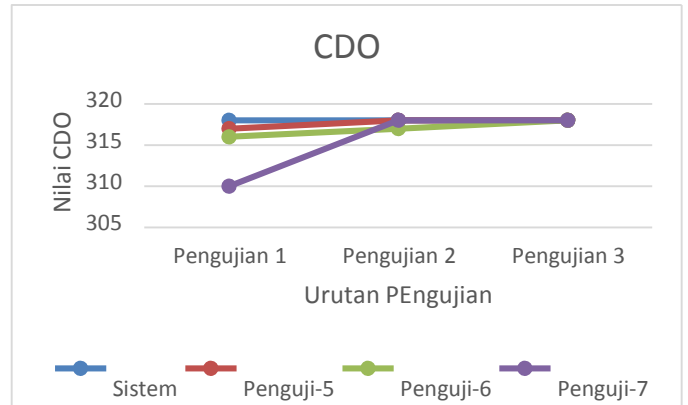
Sedangkan untuk menguji nilai yang dihasilkan pada metrik adalah valid maka dilakukan validasi oleh 3 orang penguji. Dimana 3 penguji tersebut merupakan penguji yang melakukan perhitungan secara manual terhadap dataset untuk tiap metrik. Pengujian ini dilakukan dengan cara membandingkan hasil nilai metrik yang sistem perhitungan reusability dengan hasil perhitungan secara manual dimana dataset merupakan sebuah proyek PHP.

A. CDC



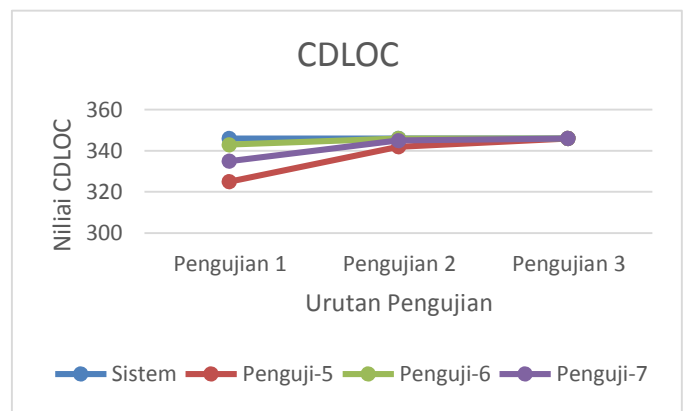
Gambar 3 Grafik Perhitungan pada metrik CDC

B. CDO



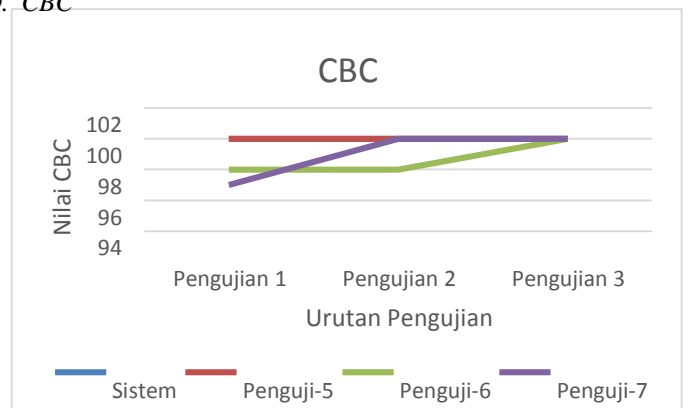
Gambar 4 Grafik Perhitungan pada metrik CDO

C. CDLOC



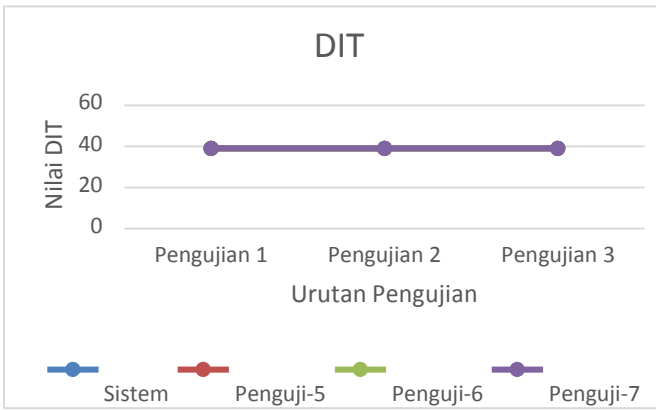
Gambar 5 Grafik Perhitungan pada metrik CDLOC

D. CBC



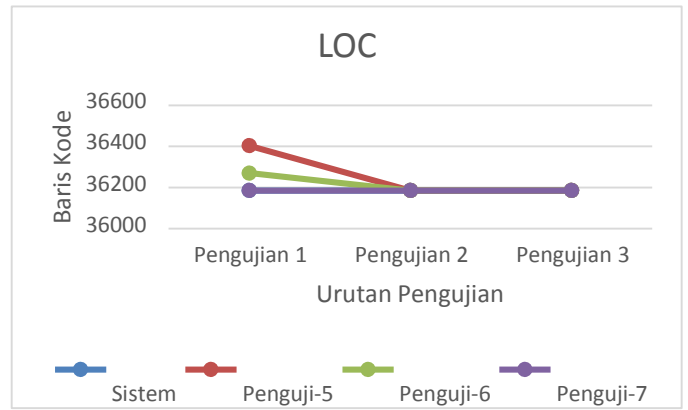
Gambar 6 Grafik Perhitungan pada metrik CBC

E. DIT



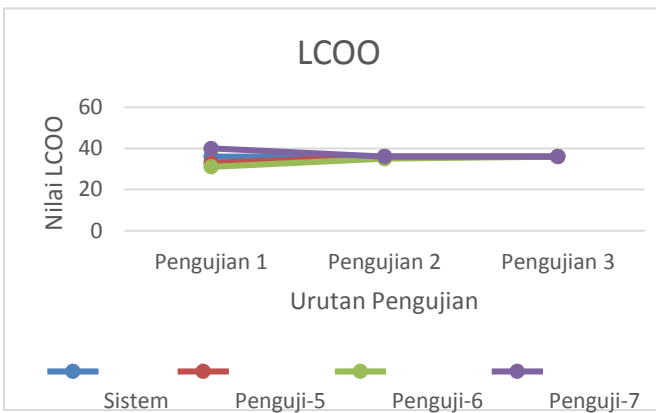
Gambar 7 Grafik Perhitungan pada metrik DIT

H. LOC



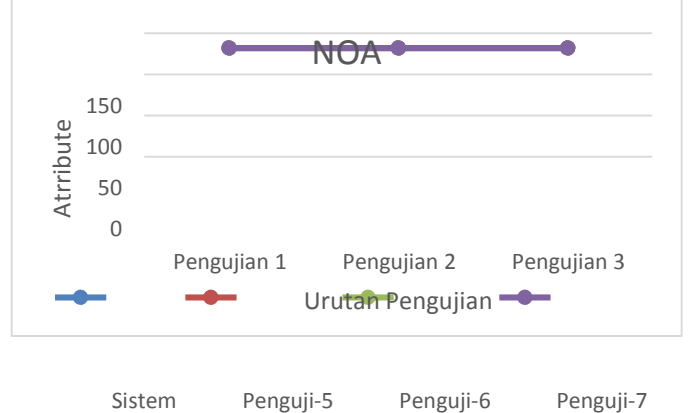
Gambar 10 Grafik Perhitungan pada metrik LOC

F. LCOO



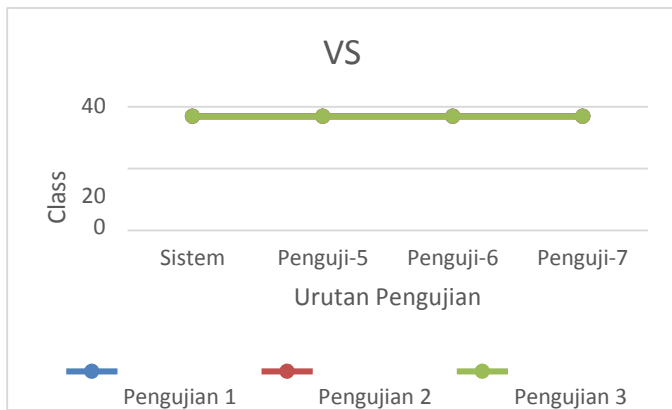
Gambar 8 Grafik Perhitungan pada metrik LCOO

I. NOA



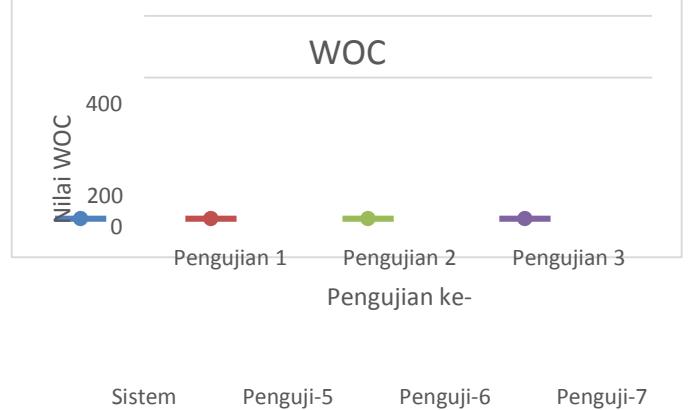
Gambar 11 Grafik Perhitungan pada metrik NOA

G. VS



Gambar 9 Grafik Perhitungan pada metrik VS

J. WOC



Gambar 12 Grafik Perhitungan pada metrik VS

Dari analisa pada pengujian yang dilakukan dengan survey (100%) dan pengujian perhitungan manual yang mendapatkan nilai sempurna (100%) pada pengujian ke tiga sehingga dapat disimpulkan bahwa nilai yang dihasilkan sistem adalah valid.

V. KESIMPULAN

Berdasarkan hasil review literature dan hasil validasi pada dapat disimpulkan faktor-faktor yang mempengaruhi tiap metrik, sebagai berikut:

- a) NOA
Metrik ini dipengaruhi oleh variable global yang berada di tiap class file.
- b) LOC
Metrik ini dipengaruhi oleh jumlah baris code, comment dan baris kosong.
- c) VS
Metrik ini dipengaruhi oleh jumlah class pada project.
- d) WOC
Metrik ini dipengaruhi oleh jumlah method dan parameter local method tersebut.
- e) CDC
Metrik ini dipengaruhi oleh inisiasi tiap concern pada tiap class.
- f) CDO
Metrik ini dipengaruhi oleh concern dan pemanggilan method pada concern terhadap tiap class.
- g) CDLOC
Metrik ini dipengaruhi oleh concern, pemanggilan variable inisiasi concern dan letak pergantian pemanggilan concern di tiap class.
- h) CBC
Metrik ini dipengaruhi jumlah instansiasi object serta extend didalam tiap class.
- i) DIT
Metrik ini dipengaruhi oleh class dan class extend terhadap class tersebut.
- j) LCOO
Metrik ini dipengaruhi oleh variable global dan method yang memiliki irisan terhadap method lain yang mengakses variable instance yang sama pada tiap class.

Metode regex dapat mengidentifikasi pattern yang ada pada tiap metrik. Langkah pertama dalam menggunakan metode regex ini adalah membaca file. Kemudian, mengekstraks sintaks sebagai string dan dicocokkan dengan pattern pada tiap metrik. Lalu dapat dihitung nilai metrik tersebut.

Berdasarkan hasil pengujian yang dilakukan, pada pengujian pertama dan kedua terdapat beberapa kasus yang menghasilkan nilai keakuratan yang berbeda. Hal ini dikarenakan, penguji masih berupaya beradaptasi dalam perhitungan awal. Pengujian ini memerlukan tingkat ketelitian pada pengujian. Kemudian pada pengujian ketiga keseluruhan pengujian menghasilkan nilai yang sama dengan nilai yang dikeluarkan oleh sistem yang dibangun. Yaitu, memiliki keakuratan 100%. Sehingga dapat disimpulkan, sistem yang dibangun sudah sesuai dengan

requirement dan prosedur penghitungan nilai metrik reusability.

REFERENSI

- [1] J. E. F. Friedl, *Mastering Regular Expressions*. O'Reilly Media, Inc., 2002.
- [2] R. Lerdorf, K. Tatro, and P. MacIntyre, *Programming PHP*. O'Reilly Media, 2006.
- [3] S. Turner, "Security vulnerabilities of the top ten programming languages: C, Java, C++, Objective-C, C#, PHP, Visual Basic, Python, Perl, and Ruby."
- [4] I. Sommerville, *Software Engineering*. Pearson/Addison-Wesley, 2011.
- [5] A. claudio Nogueira, "On the Reuse and Maintenance of Aspect-Oriented Software: An Assessment Framework," Agustus 2003.
- [6] A. Abran, *Software Metrics and Software Metrology*. John Wiley & Sons, 2010.
- [7] D. Powers, *PHP Object-Oriented Solutions*. Apress, 2008.
- [8] B. Henderson, "Object-Oriented Metrics Measures of Complexity," 1996.
- [9] H. Iqbal, "Software Engineering: A Practitioner Approach by Roger S. Pressman." [Online]. Available: http://www.academia.edu/2492406/Software_Engineering_A_Practitioner_Approach_by_Roger_S_Pressman. [Accessed: 30-Jun-2014].
- [10] H. Washizaki, H. Yamamoto, and Y. Fukazawa, "A metrics suite for measuring reusability of software components," in *Software Metrics Symposium, 2003. Proceedings. Ninth International, 2003*, pp. 211–223.
- [11] A. Abran, A. Sellami, and W. Suryan, "Metrology, measurement and metrics in software engineering," in *Software Metrics Symposium, 2003. Proceedings. Ninth International, 2003*, pp. 2–11.
- [12] M. Xenos, D. Stavrinoudis, K. Zikouli, and D. Christodoulakis, *Object-Oriented Metrics - A Survey*. 2000.
- [13] K. Thompson, "Programming Techniques: Regular expression search algorithm," *Commun ACM*, vol. 11, no. 6, pp. 419–422, Jun. 1968.