

# RANCANG BANGUN KAKAS BANTU PENGUKURAN KUALITAS PERANGKAT LUNAK KARAKTERISTIK *MAINTAINABILITY, PORTABILITY, DAN COMPATIBILITY* BERDASARKAN ISO 25010

Itsna Dzakiatul Huriroh, Sarwosri, dan Umi Laili Yuhana  
Departemen Informatika Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember (ITS)  
Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia  
e-mail: itsnadzakia@gmail.com, sarwosri@if.its.ac.id, yuhana@if.its.ac.id

**Abstrak**— *Perkembangan teknologi informasi yang sangat pesat mengakibatkan semakin banyak perangkat lunak yang dikembangkan untuk memenuhi kebutuhan manusia dalam segala jenis bidang. Untuk mencapai tujuan maksimal dalam penggunaan perangkat lunak, dibutuhkan suatu standar yang dapat menilai kualitas perangkat lunak tersebut. Sistem Informasi Akademik ITS (SIKAD ITS) merupakan salah satu situs penting dalam penyediaan informasi dan layanan akademik bagi mahasiswa Institut Teknologi Sepuluh Nopember. Mengingat pentingnya Sistem Informasi Akademik ITS, maka diperlukan adanya pengukuran untuk mengetahui kualitas SIKAD ITS. Standar yang digunakan untuk mengukur SIKAD ITS adalah ISO 25010 yang merupakan standar ISO terbaru dan hasil perbaikan dari versi ISO 9126. Karakteristik kualitas yang akan dibahas pada penelitian ini adalah Maintainability, Portability, dan Compatibility serta pengukuran otomatis nilai Cohesion subkarakteristik Modularity. Dalam Tugas Akhir ini, metode yang digunakan dalam penelitian adalah Analytical Hierarchy Process (AHP). Pertama, AHP digunakan untuk menentukan bobot karakteristik dan subkarakteristik yang akan digunakan dalam pengukuran. Bobot ini didapat dari kuesioner pakar yang membandingkan antar karakteristik dan karakteristik lain, antar subkarakteristik dan subkarakteristik lain. Setelah mendapat bobot pengukuran, dilakukan perhitungan untuk menentukan jumlah responden yang akan disurvei. Dari hasil perhitungan kemudian dilakukan penyebaran kuesioner kepada dua jenis responden yaitu 119 mahasiswa dan 16 developer. Hasil kuesioner digunakan sebagai input pada kakas bantu. Pengukuran otomatis nilai Cohesion subkarakteristik Modularity akan dihitung menggunakan library lcom. Data uji yang digunakan pada pengukuran otomatis adalah filr source code SIMPEG. Dari file yang diunggah akan dibaca oleh kakas bantu dan diambil isinya untuk menudian diukur nilai cohesionnya. Hasil dari penelitian ini adalah kakas bantu pengukuran kualitas perangkat lunak karakteristik Maintainability, Portability, dan Compatibility serta pengukuran otomatis nilai Cohesion subkarakteristik Modularity. Kakas bantu ini dapat digunakan untuk mengukur kualitas perangkat lunak yang hasilnya dapat digunakan sebagai acuan dan pertimbangan dalam pengembangan perangkat lunak kedepannya.*

**Kata Kunci**— *AHP, Cohesion, Compatibility, ISO 25010, LCOM, Maintainability, Modularity, Portability.*

## I. PENDAHULUAN

Perkembangan teknologi informasi yang sangat pesat mengakibatkan semakin banyak perangkat lunak yang dikembangkan untuk memenuhi kebutuhan manusia dalam segala jenis bidang. Untuk mencapai tujuan maksimal dalam

penggunaan perangkat lunak, dibutuhkan suatu standar yang dapat menilai kualitas perangkat lunak tersebut. Banyaknya standar kualitas yang ada saat ini menunjukkan bahwa begitu pentingnya pencapaian sebuah kualitas [1].

ISO 25010 merupakan bagian dari *Software product Quality Requirements and Evaluation* (SquaRE) yang merupakan standar untuk kualitas produk perangkat lunak, kebutuhan, dan evaluasi yang termasuk pada divisi model kualitas (ISO 2501n).

Berdasarkan penelitian sebelumnya oleh Sugiyanto (2015) pernah dilakukan pengukuran kualitas perangkat lunak dengan model pengukuran kualitas yang telah dimodifikasi berdasarkan ISO 9126 dan metode *Analytical Hierarchy Process* (AHP) [1]. Studi oleh Nida Regita Fauzianti (2020) telah membuat kakas bantu pengukuran kualitas menggunakan standar pengukuran ISO 25010 untuk mengukur kualitas produk perangkat lunak dari DPTSI dan menggunakan metode *Extended Goal Question Metric* (GQM) untuk menentukan pertanyaan yang akan digunakan dalam survey penilaian kualitas perangkat lunak [2].

Sistem Informasi Akademik (SIKAD) Integra ITS merupakan salah satu situs penting dalam penyediaan informasi akademik bagi mahasiswa Institut Teknologi Sepuluh Nopember. Dengan adanya SIKAD Integra ITS mahasiswa dapat mengakses data, informasi, serta layanan yang diperlukan untuk selama perkuliahan. Oleh karena itu, dibutuhkan suatu kakas bantu untuk mengukur kualitas SIKAD Integra ITS. Dari studi yang dilakukan oleh Nida Regita Fauzianti (2020) metode yang digunakan dalam pengukuran kualitas tidak memungkinkan untuk memberi bobot kepentingan antar karakteristik dan subkarakteristik sesuai dengan kebutuhan perangkat lunak. Oleh karena itu, pada penelitian ini akan dibuat kakas bantu menggunakan metode AHP untuk memberi bobot kepentingan yang disesuaikan dengan kebutuhan SIKAD Integra ITS untuk karakteristik *Maintainability, Portability, dan Compatibility*, serta terdapat fitur pengukuran otomatis untuk subkarakteristik *Modularity* karakteristik *Maintainability* yang mengukur nilai *Cohesion*.

## II. TINJAUAN PUSTAKA

### A. Sistem Informasi Akademik ITS

Sistem Informasi Akademik ITS merupakan sebuah sistem informasi akademik menyediakan banyak fitur yang akan

mempermudah baik mahasiswa maupun dosen ITS dalam memperoleh data, informasi, dan layanan yang berhubungan dengan akademik maupun non akademik. Dalam Sistem Informasi Akademik ITS terdapat sebanyak 14 fitur yang dapat diakses oleh pengguna. Dari semua fitur yang ada, terdapat fitur SI Akademik (SIKAD), fitur ini berisi data, informasi, dan layanan akademik yang digunakan oleh dosen dan mahasiswa [3].

### B. ISO 25010

ISO 25010 merupakan bagian lain dari seri SquARE (ISO/IEC 25000 – ISO/IEC 25099), standar untuk kualitas produk perangkat lunak, kebutuhan, dan evaluasi yang termasuk pada divisi model kualitas (ISO/IEC 2501n) [4]. Pendekatan berbasis metrik digunakan untuk menilai kualitas yang diberikan sistem dan model berkembang dari keandalan sistem digunakan untuk memprediksi kualitas [5]. Dalam ISO 25010 terdapat total 8 karakteristik dan 31 subkarakteristik. Pada penelitian ini akan diukur 3 karakteristik yaitu *Maintainability*, *Portability*, dan *Compatibility* serta pengukuran otomatis nilai *Cohesion* subkarakteristik *Modularity* karakteristik *Maintainability*.

*Maintainability* adalah tingkat efektivitas dan efisiensi suatu produk atau sistem dapat dimodifikasi oleh pengelola. Karakteristik ini terdiri dari sub-karakteristik berikut:

1. *Modularity* adalah tingkat dimana sistem atau program komputer terdiri dari komponen-komponen tersendiri sehingga perubahan pada satu komponen memiliki dampak minimal pada komponen lainnya.
2. *Reusability* adalah sejauh mana suatu aset dapat digunakan di lebih dari satu sistem, atau dalam membangun aset lainnya.
3. *Analizability* adalah tingkat efektivitas dan efisiensi yang memungkinkan untuk menilai dampak pada suatu produk atau sistem dari perubahan yang dimaksudkan untuk satu atau lebih dari bagian-bagiannya, atau untuk mendiagnosis suatu produk untuk kekurangan atau penyebab kegagalan, atau untuk mengidentifikasi bagian-bagian untuk dimodifikasi.
4. *Modifiability* adalah sejauh mana suatu produk atau sistem dapat dimodifikasi secara efektif dan efisien tanpa memperkenalkan cacat atau menurunkan kualitas produk yang ada.
5. *Testability* adalah tingkat efektivitas dan efisiensi yang dengannya kriteria pengujian dapat ditetapkan untuk suatu sistem, produk atau komponen dan pengujian dapat dilakukan untuk menentukan apakah kriteria tersebut telah dipenuhi.

*Portability* adalah tingkat efektivitas dan efisiensi suatu sistem, produk atau komponen dapat ditransfer dari satu perangkat keras, perangkat lunak, lingkungan operasional atau penggunaan lainnya ke yang lain. Karakteristik ini terdiri dari sub-karakteristik berikut:

1. *Adaptability* adalah sejauh mana suatu produk atau sistem dapat secara efektif dan efisien diadaptasi untuk perangkat keras, perangkat lunak atau lingkungan operasional atau penggunaan lainnya yang berbeda.
2. *Installability* adalah tingkat efektivitas dan efisiensi yang dapat digunakan produk atau sistem untuk berhasil diinstal dan / atau dihapus dalam lingkungan tertentu.

*Compatibility* adalah sejauh mana suatu produk, sistem atau komponen dapat bertukar informasi dengan produk, sistem atau komponen lain, dan / atau melakukan fungsi yang disyaratkan saat berbagi lingkungan perangkat keras atau perangkat lunak yang sama. Karakteristik ini terdiri dari sub-karakteristik berikut:

1. *Coexistence* adalah tingkat di mana suatu produk dapat melakukan fungsi yang diperlukan secara efisien sambil berbagi lingkungan dan sumber daya yang sama dengan produk lain, tanpa dampak yang merugikan pada produk lain.
2. *Interoperability* adalah sejauh mana dua atau lebih sistem, produk atau komponen dapat bertukar informasi dan menggunakan informasi yang telah dipertukarkan [4].

### C. Analytical Hierarchy Process(AHP)

*Analytical Hierarchy Process* adalah metode pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Model pendukung keputusan ini akan menguraikan permasalahan dengan banyak faktor maupun kriteria yang kompleks menjadi suatu hierarki. Menurut Saaty (1993), hierarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multilevel dimana level pertama adalah tujuan, yang diikuti level factor, kriteria, sub kriteria, dan seterusnya hingga level terakhir dari alternatif [1].

Menurut Kadarsyah dan Ali (1998), langkah-langkah yang dilakukan dalam metode AHP adalah sebagai berikut:

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan.
2. Membuat struktur hierarki yang diawali dengan tujuan utama
3. Membuat matrik perbandingan berpasangan yang menggambarkan kontribusi relative atau pengaruh setiap elemen terhadap tujuan atau kriteria yang setingkat di atasnya.
4. Mendefinisikan perbandingan berpasangan sehingga diperoleh jumlah penilai seluruhnya sebanyak  $n \times [(n-1)/2]$  buah, dengan  $n$  adalah banyaknya elemen yang dibandingkan.

Tabel 1.  
Skala Penilaian Perbandingan Berpasangan

Intensitas Kepentingan	Definisi
1	Kedua elemen sama penting
3	Elemen yang satu sedikit lebih penting dari elemen lainnya
5	Elemen yang satu lebih penting dari elemen lainnya
7	Elemen yang satu sangat penting dari elemen lainnya
9	Elemen yang satu mutlak sangat penting dari elemen lainnya
2, 4, 6, 8	Nilai diantara nilai di atasnya

5. Menghitung nilai eigen.
6. Menghitung vector eigen dari setiap matriks perbandingan berpasangan yang merupakan bobot setiap elemen untuk penentuan prioritas elemen-elemen pada tingkat hierarki terendah sampai mencapai tujuan. Perhitungan dilakukan dengan menjumlahkan nilai setiap

kolom yang bersangkutan untuk memperoleh normalisasi matriks dan menjumlahkan nilai-nilai dari setiap baris lalu membainya dengan jumlah elemen untuk mendapatkan rata-rata.

Apabila A adalah matriks perbandingan berpasangan, maka vector bobot yang berbentuk:

$$A(i,j) = n \rightarrow A(j,i) = \frac{1}{n}, i \neq j$$

$$A(i,j) = A(j,i) = 1, i = j$$
(1)

Dapat didekati dengan cara:

1. Menormalkan setiap kolom j dalam matriks A sedemikian hingga:

$$\sum_i a_{ij} = 1$$
(2)

2. Hitung nilai rata-rata untuk setiap baris i dalam A' dengan  $w_i$  adalah bobot tujuan ke-I dari vector bobot.

$$w_i = \frac{1}{n} \sum_j a_{ij}$$
(3)

Dalam penelitian ini, metode AHP digunakan pada perhitungan bobot karakteristik dan bobot subkarakteristik, yang selanjutnya akan disebut sebagai bobot relatif. Perhitungan bobot yang telah didapatkan dengan menggunakan data kuesioner pakar akan dimasukkan ke model kualitas ISO 25010. Model kualitas ini akan menjadi dasar dari pengukuran kualitas perangkat lunak.

#### D. Perhitungan Total Kualitas Perangkat Lunak

Nilai total kualitas perangkat lunak dihitung dari nilai kuesioner yang dimasukkan ke dalam model kualitas, selanjutnya dihitung bobot absolut dengan cara mengalikan bobot karakteristik kualitas dengan bobot relatif subkarakteristik. Kemudian, nilai absolut dihitung dengan mengalikan bobot absolut dengan nilai subkarakteristik kualitas. Total kualitas perangkat lunak dapat dihitung dengan menjumlahkan nilai absolut keseluruhan subkarakteristik [1].

$$\begin{aligned} BA &= BK \times BR \\ NS &= R \times 25 \\ NA &= BA \times NS \end{aligned}$$
(4)

$$\text{Total Kualitas} = NA1 + NA2 + NA3 + \dots + NAn$$

Dimana:

BA = Bobot Absolut

BK = Bobot Karakteristik Kualitas

BR = Bobot Relatif Subkarakteristik

R = Rata-rata hasil kuesioner per subkarakteristik

NS = Nilai Subkarakteristik Kualitas

NA = Nilai Absolut

#### E. Lack of Cohesion in Methods (LCOM)

Cohesion menggambarkan hubungan elemen-elemen dalam suatu modul. LCOM menghitung selisih antara jumlah *method* yang tidak terkait satu sama lain dalam menjalankan tugasnya, dengan jumlah *method* yang saling berkaitan setidaknya menggunakan satu variabel instans bersama. Semakin besar nilai LCOM, semakin besar pula kompleksitas class tersebut dan berpengaruh pada rendahnya tingkat *reusability* dan *efficiency*. LCOM dapat dihitung dengan persamaan berikut [6].

$$\sigma() = \{I_1\} \cap \{I_2\}$$
(5)

$I_1$  = himpunan parameter atau return type yang dimiliki method 1.

$I_2$  = himpunan parameter atau return type yang dimiliki method 2.

Catatan : a() dilakukan terhadap semua perpaduan method dalam satu class.

$$LCOM = \begin{cases} 0, & P < Q \\ P - Q, & P \geq Q \end{cases}$$
(6)

P = jumlah a() dengan nilai 0

Q = jumlah a() dengan nilai lebih dari 0

#### F. Library lcom

Library lcom dapat digunakan untuk menghitung nilai lcom yang dibuat oleh Asis Pattisahusiwa dan merupakan sebuah library hasil ekstraksi dari PHPMetrics yang telah dimodifikasi [7]. PHPMetrics adalah sebuah open source tool yang dapat digunakan untuk mengukur dan menganalisa suatu project PHP [8]. Dalam Tugas Akhir ini penggunaan library lcom terdapat pada lapisan kontrol cohesi dan digunakan untuk menghitung nilai *Cohesion* subkarakteristik *Modularity*.

### III. ANALISIS DAN PERANCANGAN

#### A. Analisis Permasalahan

Permasalahan yang diangkat dalam penelitian ini adalah bagaimana membangun suatu kakas bantu pengukuran kualitas perangkat lunak karakteristik *Maintainability*, *Portability*, dan *Compatibility* berdasarkan ISO 25010, dan pengukuran otomatis *cohesion* pada subkarakteristik *Modularity* karakteristik *Maintainability* pada Sistem Akademik Integra ITS.

Sistem Informasi Akademik (SIKAD) Integra ITS merupakan salah satu situs penting dalam penyediaan data, informasi, dan layanan akademik bagi mahasiswa Institut Teknologi Sepuluh Nopember. Karenanya dibutuhkan suatu kakas bantu untuk mengukur kualitas SIKAD Integra ITS. Dari studi yang dilakukan sebelumnya, Nida Regita Fauzianti (2020) menggunakan metode GQM untuk mengukur kualitas produk layanan DPTSI, namun metode yang digunakan dalam pengukuran kualitas tidak memungkinkan untuk memberi bobot kepentingan antar karakteristik dan subkarakteristik sesuai dengan kebutuhan perangkat lunak.

Untuk menjawab permasalahan diatas, pada penelitian ini akan dibuat kakas bantu pengukuran kualitas berdasarkan ISO 25010 dan menggunakan metode AHP untuk memberi bobot kepentingan yang disesuaikan dengan kebutuhan SIKAD Integra ITS untuk karakteristik *Maintainability*, *Portability*, dan *Compatibility*, serta terdapat fitur pengukuran otomatis untuk subkarakteristik *Modularity* karakteristik *Maintainability* yang mengukur nilai *Cohesion*. *Maintainability* menilai seberapa efektif dan efisien SIKAD Integra ITS dapat dimodifikasi, *Portability* menilai kemudahan SIKAD Integra untuk dipindahkan dari suatu lingkungan operasional tertentu, sedangkan *Compatibility* menilai sejauh mana SIKAD Integra dapat bertukar informasi dan menjalankan fungsi yang dibutuhkan dalam lingkup perangkat keras yang sama.

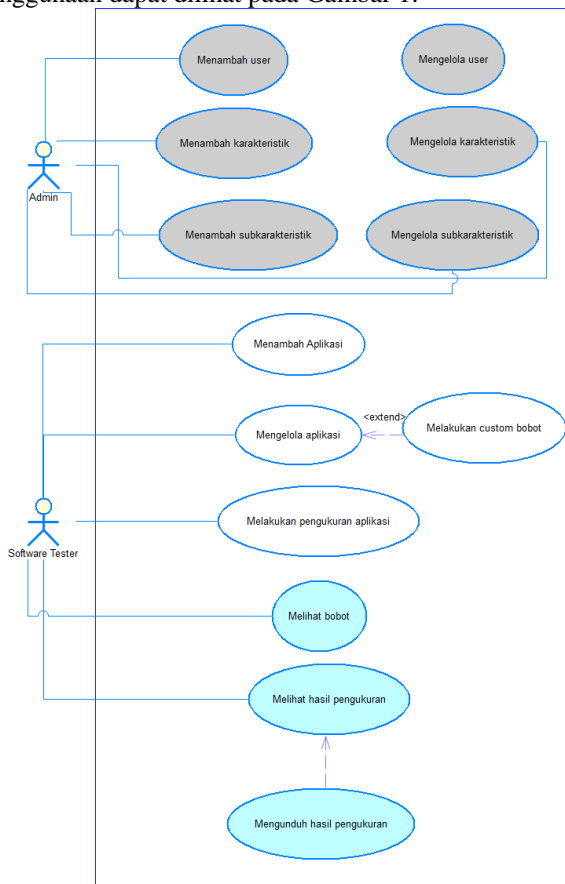
### B. Deskripsi Umum Perangkat Lunak

Pada penelitian ini akan dibuat sebuah kakas bantu berupa aplikasi berbasis web yang dapat digunakan untuk mengukur kualitas SIAKAD Integra ITS karakteristik *Maintainability*, *Portability*, dan *Compatibility* dan fitur pengukuran otomatis *Cohesion* subkarakteristik *Modularity*. Deskripsi Umum sistem dapat dilihat sebagai berikut:

- **Admin**  
Admin berperan sebagai pengguna yang mengelola data pengukuran, mengelola karakteristik dan subkarakteristik.
- **Software Tester**  
*Software tester* berperan sebagai pengguna yang melakukan pengukuran kualitas perangkat lunak. *Software tester* juga dapat melihat hasil analisis laporan pengukuran.
- **Database**  
*Database* digunakan untuk menyimpan data pengguna, nilai bobot karakteristik, nilai bobot relatif, nilai hasil pengukuran subkarakteristik, nilai hasil pengukuran karakteristik, dan nilai hasil pengukuran aplikasi.
- **Website**  
*Website* digunakan sebagai kakas bantu pengukuran dan menampilkan hasil pengukuran aplikasi.

### C. Spesifikasi Perangkat Lunak

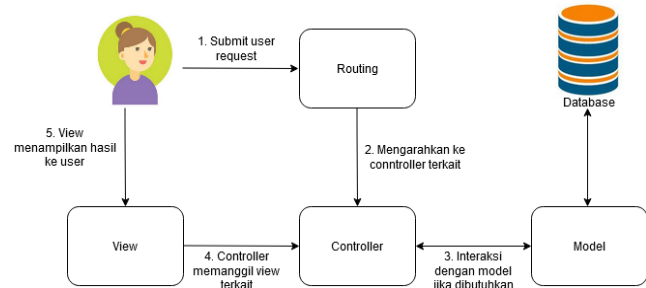
Subbab ini membahas spesifikasi kebutuhan perangkat lunak dari hasil analisis yang telah dilakukan. Kasus penggunaan dapat dilihat pada Gambar 1.



Gambar 1 Kasus Penggunaan

### D. Perancangan Arsitektur Sistem

Arsitektur sistem yang digunakan pada platform web pada perangkat lunak ini menggunakan kerangka kerja Laravel dengan pola perancangan *Model View Controller* (MVC). Ilustrasi arsitektur sistem dapat dilihat pada Gambar 2.



Gambar 2 Arsitektur Perangkat Lunak

Dalam arsitektur MVC, model merupakan penghubung antara *controller* dan *database*, saat pengguna melakukan *request* data, maka *route* akan menerima permintaan dan menghubungkan ke *controller* terkait. *Controller* akan menerima *request* kemudian menghubungkan dengan model terkait untuk kemudian model mengambil data yang diperlukan dari *database*. Setelah menerima data, model mengirimkan data kepada *controller* untuk diteruskan ke *view*. *View* berperan untuk menampilkan hasil dari *controller* kepada pengguna melalui *browser*. Berikut merupakan daftar model, *view*, dan *controller* yang digunakan dalam pembuatan kakas bantu pengukuran perangkat lunak ISO 25010.

### E. Perancangan Algoritma Perhitungan Bobot

Pada bagian ini akan dijelaskan mengenai perancangan algoritma perhitungan bobot karakteristik dan subkarakteristik dengan menggunakan metode AHP. Berikut merupakan pseudocode dari AHP.

```

for i=1 to 8
  input ki
  read (k1, k2, k3, k4, k5, k6, k7, k8)
  for k=1 to 8
    for ka=1 to 8
      matriks [ka] [k]
    normalisasi [ka] [k] = matriks [ka] [k] /
    jumlah [ka]
  for k=1 to 8
    jumlah_normalisasi = normalisasi [k] [k]
    bobot_karakteristik = jumlah_normalisasi
    / count [k]
  output
  bobot_karakteristik

```

### F. Perancangan Algoritma Pengukuran

Pada bagian ini akan dijelaskan perancangan algoritma pengukuran kualitas perangkat lunak karakteristik *Maintainability*, *Portability*, dan *Compatibility*. Pseudocode algoritma pengukuran dapat dilihat sebagai berikut.

```

m = banyak karakteristik
for i=1 to m
  n = banyak subkarakteristik suatu
  karkteristik
  for i=1 to n
    input total nilai kuesioner (JS)
    input total responden (r)
    nilai sub = JS / r *25

```

```

output nilai sub (NS)
nilai absolut (NA) =BK x BS x NS
output NA
nilai karakteristik (NK)= NA1+NA2+...Nam
output NK

```

*Software tester* dapat melakukan pengukuran setelah memasukkan data aplikasi dan memilih bobot pengukuran baik menggunakan bobot patokan maupun *custom*. Setelah itu akan diarahkan ke halaman pengukuran. Di halaman ini *software tester* harus mengisi data kuesioner yang berisi total hasil kuesioner dan jumlah responden yang mengisi kuesioner untuk setiap subkarakteristik. Input dari *software tester* akan disimpan dan dihitung untuk mendapatkan hasil akhir yaitu nilai karakteristik.

#### G. Perancangan Algoritma Pengukuran Otomatis

Pada bagian ini akan dijelaskan mengenai perancangan algoritma pengukuran otomatis *Cohesion* subkarakteristik *Modularity* karakteristik *Maintainability*.

```

input
file source code
open (file source code)
content = read (file source code)
for content
count lcom
output nilai lcom

```

Dari pseudocode diatas untuk pengukuran otomatis *Software tester* telah mengunggah *file source code* yang akan diukur nilai *Cohesion*-nya saat awal menambah data aplikasi baru. *File* ini nanti akan dibaca, dan dihitung nilai *cohesion*-nya dengan menggunakan *library lcom*.

### IV. IMPLEMENTASI

#### A. Implementasi Kakas Bantu Pengukuran

Pembuatan Kakas Bantu Pengukuran Kualitas Perangkat Lunak Karakteristik *Maintainability*, *Portability*, dan *Compatibility* diimplementasikan menggunakan bahasa pemrograman PHP, dengan *framework* Laravel, dan basis data MySQL.

#### B. Implementasi Pengukuran Otomatis

Pengukuran otomatis *Cohesion* Subkarakteristik *Modularity* Karakteristik *Maintainability* diimplementasikan menggunakan *library lcom*.

### V. UJI COBA DAN EVALUASI

Pengujian digunakan dengan metode *black box* untuk menguji masing-masing fungsionalitas yang sudah dirancang pada sistem. Metode ini merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas dari suatu perangkat lunak tanpa memandang struktur internalnya. Rincian rekapitulasi kasus penggunaan dijelaskan pada Tabel 2.

Tabel 2.  
Rekapitulasi Hasil Pengujian

Kasus Uji	Nama Kasus Pengujian	Skenario	Terpenuhi
UJ-001	Melihat Bobot	-	✓

Kasus Uji	Nama Kasus Pengujian	Skenario	Terpenuhi
UJ-002	Melihat Hasil Pengukuran	-	✓
UJ-003	Mengunduh Hasil Pengukuran	-	✓

Pengukuran otomatis nilai *Cohesion* subkarakteristik *Modularity* karakteristik *Maintainability* dilakukan dengan langkah-langkah berikut:

1. Memasukkan data aplikasi yang akan diukur.
2. Mengunggah file *source code* aplikasi berekstensi .php yang akan diukur. Pada penelitian ini data uji yang digunakan adalah *source code* SIMPEG.
3. Menjalankan menu pengukuran pada subkarakteristik *Modularity* karakteristik *Maintainability*.

Dari pengukuran yang telah dilakukan dengan *file* masukan *simpeg.php*. Hasil yang diperoleh dari pengukuran oleh Kakas Bantu Pengukuran adalah terdapat 1 *class* bernama *Simpeg* dengan jumlah fungsi 204 dan nilai LCOM sebesar 204.

Dengan input file yang sama, dilakukan juga perhitungan nilai *lcom* dengan menggunakan aplikasi *PhpMetrics*. Dari perhitungan yang dilakukan, didapatkan hasil terdapat 1 *class*, dan nilai LCOM sebesar 204.

Hasil perhitungan nilai LCOM oleh kakas bantu dan *PhpMetrics* memiliki hasil yang sama yaitu 204. Perbandingan hasil pengukuran dapat dilihat pada Tabel 3.

Tabel 3.  
Perbandingan Hasil Pengukuran Otomatis

	Jumlah Class	Jumlah Fungsi	Nilai LCOM
Kakas Bantu	1	204	204
Phpmetric	1	-	204

Nilai Subkarakteristik *Modularity* yang sebelumnya diperoleh dari perhitungan otomatis dan telah dikonversikan menjadi skala 1-100 adalah 0. Nilai ini diperoleh dengan menghitung nilai 1 dikurangi selisih antara nilai LCOM yang diperoleh dan nilai LCOM maksimal kemudian dikalikan 100.

Pengukuran karakteristik *Maintainability*, *Portability*, dan *Compatibility* dilakukan dengan langkah-langkah berikut:

1. Memasukkan data aplikasi yang akan diukur.
2. Menjalankan menu pengukuran pada subkarakteristik yang akan diukur.
3. Memasukkan nilai total kuesioner per subkarakteristik dan jumlah responden.

Tabel 4  
Rekapitulasi Hasil Pengukuran

No	Karakteristik	Deskripsi	Nilai	Keterangan
1	Maintainability	Kemudahan perangkat lunak untuk dimodifikasi oleh pengelola.	59.69	Tidak Terpenuhi
2	Portability	Kemampuan perangkat lunak	68.00	Tidak Terpenuhi

No	Karakteristik	Deskripsi	Nilai	Keterangan
		dapat ditransfer dari satu perangkat keras, perangkat lunak, lingkungan operasional atau penggunaan lainnya ke yang lain. Kemampuan perangkat lunak untuk bertukar informasi dengan produk, sistem atau komponen lain, dan / atau melakukan fungsi yang disyaratkan saat berbagi lingkungan perangkat keras atau perangkat lunak yang sama		
3	Compatibility		81.25	Terpenuhi

Hasil pengukuran karakteristik *Maintainability* mendapatkan nilai 7.76. Bobot karakteristik *Maintainability* adalah 0.14, jika dikonversikan dalam skala 1-100, maka nilai maksimal dari karakteristik *Maintainability* adalah 13. Maka nilai dari pengukuran Sistem Informasi Akademik Integra ITS karakteristik *Maintainability* dihitung dengan nilai *Maintainability* dibagi dengan nilai maksimal *Maintainability* dikali seratus, hasilnya adalah 59.69.

Hasil pengukuran karakteristik *Portability* mendapatkan nilai 2.72. Bobot karakteristik *Portability* adalah 0.04, jika dikonversikan dalam skala 1-100, maka nilai maksimal dari karakteristik *Portability* adalah 4. Maka nilai dari pengukuran Sistem Informasi Akademik Integra ITS karakteristik *Portability* dihitung dengan nilai *Portability* dibagi dengan nilai maksimal *Portability* dikali seratus, hasilnya adalah 68.00.

Hasil pengukuran karakteristik *Compatibility* mendapatkan nilai 3.25. Bobot karakteristik *Compatibility* adalah 0.04, jika dikonversikan dalam skala 1-100, maka nilai maksimal dari karakteristik *Compatibility* adalah 4. Maka nilai dari pengukuran Sistem Informasi Akademik Integra ITS karakteristik *Compatibility* dihitung dengan nilai *Compatibility* dibagi dengan nilai maksimal *Compatibility* dikali seratus, hasilnya adalah 81.25.

## VI. KESIMPULAN

Berdasarkan hasil yang didapatkan pada tahap uji coba dan evaluasi, penulis mengambil beberapa kesimpulan sebagai berikut:

1. Dari pengukuran yang telah dilakukan pada Sistem Informasi Akademik ITS, menurut nilai standar untuk aplikasi web yaitu 70 [9], dapat disimpulkan bahwa Sistem Akademik ITS telah memenuhi standar kualitas situs web untuk karakteristik *Compatibility* dengan nilai 81.25, namun belum memenuhi standar untuk karakteristik *Maintainability* 59.69 dan *Portability* 68.00.
2. Hasil pengukuran nilai *lcom Cohesion* pada SIMPEG adalah 204 dari 1 class dan 204 fungsi. Maka nilai nilai subkarakteristik *Modularity* adalah 0.00.

## UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih kepada orang tua dan keluarga penulis yang telah memberikan dukungan doa, moral, dan material, juga kepada Ibu Sarwosri dan Ibu Umi Laili Yuhana selaku pembimbing I dan II yang telah membimbing, memberikan motivasi, dan masukan serta semua pihak yang yang tidak dapat penulis sebutkan satu-persatu.

## DAFTAR PUSTAKA

- [1] Sugiyanto, Perbaikan Model Penilaian Kualitas Perangkat Lunak Pada Domain Situs Web Perguruan Tinggi Berbasis Pendekatan Multi Perspektif, Surabaya: Program Magister Bidang Keahlian Rekayasa Perangkat Lunak Jurusan Teknik Informatika Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, 2015.
- [2] N. R. Fauzianti, Rancang Bangun Kakas Pengukuran Kualitas pada Produk Manajemen Proyek Perangkat Lunak Berbasis Extended Goal Question Metric, Surabaya, 2021.
- [3] "myITS Single Sign On," [Online]. Available: <https://my.its.ac.id/>. [Accessed 2 November 2020].
- [4] "ISO/IEC 25022:2016 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use," ISO, 2011. [Online]. Available: <https://www.iso.org/>. [Accessed 7 Maret 2020].
- [5] Wagner, "Software Product Quality Control," 2013.
- [6] A. X. A. S. G. H. Arwin Halim, Mengukur Tingkat Reusability dan Efficiency dari, Denpasar: Prodi Pendidikan Teknik Informatika Jurusan Teknik Informatika Fakultas Teknik dan Kejuruan Universitas Pendidikan Ganesha, 2014.
- [7] A. Pattisahusiwa, "lcom," 13 December 2020. [Online]. Available: <https://github.com/pts-archive/lcom>.
- [8] J.-F. Lépine, "PhpMetrics, a static analysis tool for PHP - by Jean-François Lépine," 5 December 2020. [Online]. Available: <https://www.phpmetrics.org/index.html>.
- [9] S. A. R. T. H. R. Anita Hidayati, "Analisa Pengembangan Model Kualitas Berstruktur Hirarki Dengan Kustomisasi ISO 9126 Untuk Evaluasi Aplikasi Perangkat Lunak B2B," *Analisa Pengembangan Model Kualitas Berstruktur Hirarki Dengan Kustomisasi ISO 9126 Untuk Evaluasi Aplikasi Perangkat Lunak B2B*, 2009.