

Analisis Kualitas Perangkat Lunak E-Voting dengan McCall's metode

Fransiskus Panca Juniawan ke-1, Dwi Yuny Sylfania ke-2, Laurentinus ke-3, Rahmat Sulaiman ke-4, Rendy Rian Chrisna Putra ke-5, Harizki ke-6
Arie Pradana, Tri Sugihartono ke-7, Hengki ke-8

1,2,3,4,5,6,7 Jurusan Teknik Informatika Fakultas Teknologi Informasi

8Departemen Bisnis Digital, Fakultas Ekonomi Bisnis
ISB Atma Luhur

Pangkalpinang, Indonesia

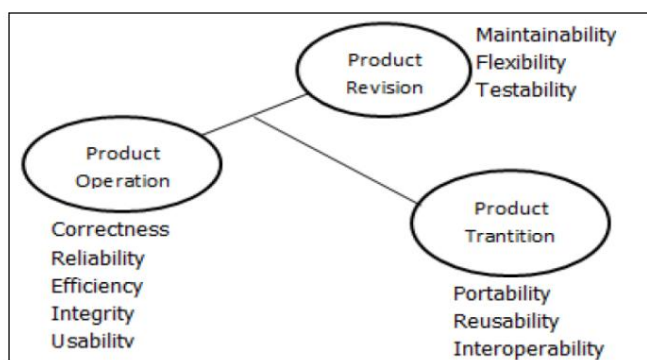
1fransiskus.pj@atmaluhur.ac.id, 2 dysylfania@atmaluhur.ac.id, laurentinus@atmaluhur.ac.id, 4rahmatsulaiman@atmaluhur.ac.id,
5rendyriancp@atmaluhur.ac.id, 6harizkiariep@atmaluhur.ac.id, 7trisugihartono@atmaluhur.ac.id, 8hengki@atmaluhur.ac.id

Intisari—Saat ini pengujian perangkat lunak merupakan suatu hal yang penting, terutama untuk mengetahui kualitasnya. Penelitian ini bertujuan untuk menguji kualitas perangkat lunak e-voting dengan menggunakan kerangka metode McCall yang berfokus pada aspek Product Operation perangkat lunak. Dengan mengetahui kualitas software e-voting saat ini, kita dapat mengetahui seberapa besar dampak positif dari software yang ada, selain itu metrik yang paling rendah dapat diperbaiki kedepannya. Ada lima metrik yang diuji yaitu Correctness, Reliability, Efficiency, Integrity, dan Usability. Hasil pengujian menunjukkan nilai metrik tertinggi adalah kebenaran sebesar 89,85%, dan metrik terendah adalah efisiensi sebesar 48,3%. Rata-rata, perangkat lunak e-voting dinilai berkualitas baik dengan skor rata-rata sebesar 79,932%.

Kata Kunci—e-Voting, McCall, Pengujian Kualitas Perangkat Lunak

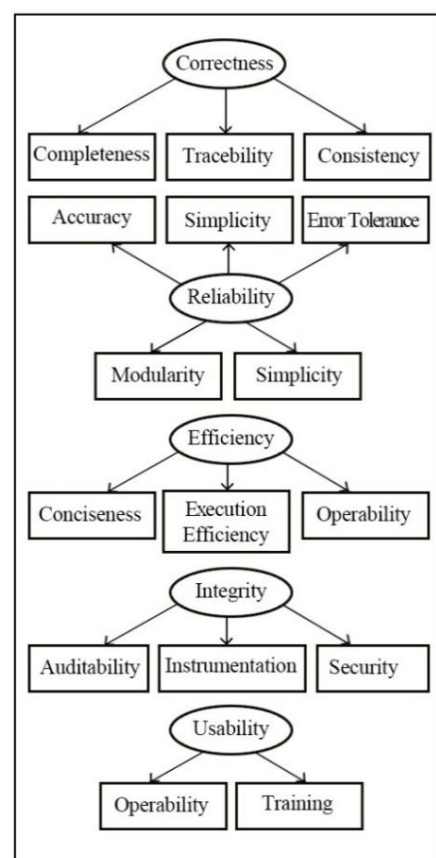
I. PENDAHULUAN

Dalam membangun perangkat lunak, banyak aspek yang harus diperhatikan. Salah satunya adalah kualitas perangkat lunak [1]. Hal ini penting karena dalam pengembangan perangkat lunak, kualitas dijadikan jaminan dalam setiap siklus hidup perangkat lunak [2]. Ada banyak metode yang dapat digunakan dalam menguji kualitas perangkat lunak. Salah satunya adalah metode McCall. Penelitian ini akan menguji kualitas software e-voting berbasis android yang telah diterapkan pada ISB Atma Luhur untuk pemilihan Ketua BEM [3]. Tujuan pengujian kualitas perangkat lunak adalah untuk mengetahui apakah implementasi perangkat lunak sudah sesuai dengan kebutuhan [1]. Saat ini, software e-voting belum pernah diuji kualitasnya. Untuk itu pengujian kualitas software ini penting untuk dilakukan. Metode McCall dipilih karena mempunyai 3 pilihan aspek penting yaitu Product Operation, Product Revision, dan Product Transition yang dapat disesuaikan dengan perangkat lunak yang akan diuji [4]. Selain itu metode ini mempunyai kriteria yang paling lengkap dan mendalam [5].



Gambar 1. Faktor Kualitas Perangkat Lunak McCall

Penelitian ini akan fokus pada pengujian pengoperasian produk yang terdiri dari aspek Correctness, Reliability, Efficiency, Integrity, dan Usability karena mengacu pada pengoperasian perangkat lunak oleh setiap pengguna. Masing-masing aspek tersebut juga memiliki kriteria tersendiri seperti terlihat pada Gambar 2 [6].



Gambar 2. Hubungan Kriteria Produk Operasi

Beberapa penelitian sebelumnya melakukan pengujian perangkat lunak berbasis web dengan menggunakan metode McCall. Salah satunya dengan mengintegrasikan operabilitas dan metrik pelatihan pada aspek Usability pada perangkat lunak [6]. Selanjutnya terdapat pengujian perangkat lunak yang berfokus pada Product Operations dengan lima metriknya. Hasil penelitian ini cukup baik dengan nilai 55,22% [5] dan mempunyai nilai baik dengan rata-rata 77,1% [7]. Penelitian lain menguji perangkat lunak sistem informasi akademik [8] dan menghasilkan nilai yang baik, namun sebaliknya, penelitian lain memiliki kekurangan dalam metrik kebenaran, efisiensi, dan integritas [2]. Aspek Usability menjadi fokus pengujian dengan mengutamakan operabilitas dan metrik pelatihan pada perangkat lunak sistem bonus dan mendapatkan hasil yang baik [9].

Metode McCall juga digunakan untuk menguji modul BAQ (Business Activity Queries) pada sistem ERP bernama Epicor. Hasil pengujian adalah kebenaran, dan metrik kegunaan memiliki nilai yang baik [10]. Selain itu metode McCall juga dapat dikombinasikan dengan white box, black box, alpha pengujian, dan pengujian beta dalam melakukan pengujian yang mempunyai hasil pengujian yang sangat baik [4]. Metode McCall juga dapat dipadukan dengan metode AHP dan SMART dalam sistem pendukung keputusan. Penelitian ini terdiri dari 12 kriteria untuk menentukan dosen terbaik. Hasilnya bagus, dengan nilai 78,2% [11].

II. METODOLOGI

Penelitian ini dilakukan dengan metode kuantitatif dengan mengutamakan objektivitas data berupa angka/skor/nilai atau pernyataan yang dinilai dan dianalisis dengan analisis statistik [9]. Dalam prosesnya, penelitian ini dilakukan melalui beberapa tahap, yaitu:

A. Pengumpulan Data

Pada tahap ini kita mengumpulkan data yang akan digunakan untuk mendukung penelitian. Data juga diperoleh dari studi literatur yang mengumpulkan pengujian teori dan penelitian terdahulu. Data juga diperoleh dari buku atau artikel ilmiah dari jurnal yang berkaitan dengan tema yang diambil. Selanjutnya informasi dikumpulkan dari perangkat lunak yang akan diuji. Objek penelitian adalah software e-voting Ketua BEM berbasis Android yang digunakan oleh mahasiswa ISB Atma Luhur dalam memilih ketua BEM setiap tahunnya.

B. Populasi

Populasi penelitian ini adalah mahasiswa aktif semester 2 sampai dengan 6 di ISB Atma Luhur yang berjumlah 763 orang. Sampel yang diambil dari penelitian ini menggunakan simple random sampling yang berjumlah 60 orang.

C. Analisis Kuesioner

Tahap selanjutnya adalah menganalisis hasil kuesioner untuk mendapatkan nilai kualitas perangkat lunak. Pengujian dilakukan dengan Metode McCall yang berfokus pada Product Operations yang terdiri dari lima kriteria. Yaitu, Kebenaran, Keandalan, Efisiensi, Integritas, dan Kegunaan [6].

Persamaan 1 adalah rumus yang digunakan untuk mengukur faktor-faktor kualitas perangkat lunak dalam penelitian ini.

$$\bar{y} = \frac{\sum y}{n} \quad (1)$$

Di mana:

Fq = Faktor Kualitas Perangkat Lunak

c1 = Skala pertama yang bergantung pada produk m1 =

Metrik pertama yang mempengaruhi faktor kualitas perangkat lunak

Persamaan 2 digunakan untuk menguji validitas sampel populasi yang menggunakan tingkat kepercayaan 95%.

$$(1 - \alpha) / (2) = \frac{1}{\sqrt{n}} \quad (2)$$

Di mana:

N = Jumlah Populasi n = Jumlah

Sampel s = Standar Deviasi

z = Tingkat Keyakinan (95% = 1,96 dan 99% = 2,58)

$$(1 - \alpha) / (2) = \frac{1}{\sqrt{n}} \quad (2)$$

$$\frac{1}{\sqrt{n}} = \frac{1,96}{\sqrt{763}} = 0,022$$

$$0,022 = 2,2\%$$

Dari perhitungan margin of error dengan tingkat kepercayaan 95% diperoleh hasil sebesar 0,02%. Jadi dapat disimpulkan bahwa jumlah koresponden yang digunakan dalam penelitian ini sudah sesuai dengan standar [9].

AKU AKU AKU. HASIL DAN PEMBAHASAN

Hasil kuesioner yang diperoleh kemudian dianalisis dengan menggunakan metode McCall yang berfokus pada pengoperasian produk dengan lima metrik yang digunakan seperti pada Gambar 2 dengan pembahasan singkat yang akan diuraikan sebagai berikut.

A. Kebenaran

Pengujian ini untuk mengetahui apakah perangkat lunak memenuhi spesifikasi dan tujuan misi pengguna [6]. Metrik ini terdiri dari tiga faktor pengujian yaitu Kelengkapan, Ketertelusuran, dan Konsistensi.

1) *Kelengkapan* : Faktor ini untuk mengetahui sejauh mana capaian pelaksanaan dan fungsi-fungsi yang diperlukan telah tercapai [8]. Perangkat lunak ini memiliki 23 fitur yang diuji. Terdapat 21 fitur yang sesuai input dan outputnya serta 2 fitur yang tidak sesuai. Dengan demikian nilai kelengkapannya adalah 21/23 atau sebesar 91,3% dengan kesimpulan perangkat lunak mempunyai nilai kelengkapan yang sangat baik.

2) *Traceability* : Faktor ini merupakan kemudahan untuk merujuk kembali implementasi atau komponen program terhadap kebutuhan pengguna perangkat lunak [6]. Dari 23 fitur perangkat lunak yang diuji, terdapat 18 fitur yang sesuai dengan sistem dan dokumen analisisnya serta lima fitur yang tidak kompatibel. Nilai traceability sebesar 18/23 atau sebesar 78,26% dengan kesimpulan perangkat lunak mempunyai nilai kesesuaian yang baik.

3) *Konsistensi*: Faktor ini adalah tingkat penggunaan teknik desain dan dokumentasi yang tepat dalam semua proyek pengembangan perangkat lunak [8]. Pengujian dilakukan terhadap empat menu perangkat lunak. Total keterangan kesesuaian ada empat menu. Jadi nilai konsistensinya adalah 4/4 dari 100% dengan kesimpulan perangkat lunak mempunyai konsistensi yang sangat baik antara dokumen analisis dengan sistem. Berdasarkan hasil pengujian ketiga faktor Correctness maka dapat dihitung persentase hasil akhir seperti pada persamaan 3.

$$\text{Correctness} = \frac{91,3 + 78,26 + 100}{3} \quad (3)$$

Dapat disimpulkan bahwa nilai Correctness perangkat lunak sangat baik [10].

B. Keandalan

Pengujian ini dilakukan untuk mengetahui sejauh mana perangkat lunak dapat diharapkan dapat menjalankan fungsinya dengan keakuratan yang diperlukan. Pengujian ini memiliki lima faktor yaitu, Akurasi, Kesederhanaan, Toleransi Kesalahan, Modularitas, dan Konsistensi.

1) *Akurasi*: Faktor ini digunakan untuk menentukan nilai akurasi dalam komputasi dan pengendalian program [8]. Pengujian dilakukan terhadap menu yang ada pada perangkat lunak. Dari 29 fitur perangkat lunak, 24 fitur memiliki kesesuaian antara input dan output komputasi, sedangkan lima fitur tidak sesuai. Hasil pengujian akurasi sebesar 24/29 atau sebesar 82,75% dengan kesimpulan akurasi perangkat lunak sangat baik.

2) *Simplicity* : Faktor ini merupakan uji kesederhanaan dalam penggunaan source code yang mengacu pada seberapa banyak suatu modul dipanggil, jika lebih besar atau sama dengan jumlah pemanggilan suatu modul maka semakin mudah untuk dipahami (sederhana) [6]. Total fitur yang diuji sebanyak 30 fitur dengan 21 fitur sederhana dan sembilan fitur tidak sederhana. Nilai kesederhanaannya sebesar 21/30 atau 70%, dengan kesimpulan tingkat kesederhanaan penggunaan source code pada perangkat lunak adalah baik.

3) *Error Tolerance* : Faktor ini merupakan pengujian yang dilakukan untuk mengetahui nilai toleransi terhadap kesalahan yang terjadi pada perangkat lunak [8]. Pengujian dilakukan terhadap 29 fitur perangkat lunak dimana terdapat 26 fitur yang sesuai sehingga nilai toleransi kesalahan sebesar 26/29 atau 89,65% dengan kesimpulan sangat baik intoleransi terhadap kesalahan yang terjadi pada saat penggunaan.

4) *Modularitas* : Pengujian ini dilakukan untuk mengetahui seberapa independen modul-modul yang merupakan komponen perangkat lunak [6]. Pengujian dilakukan pada lima kelas dengan satu kelas mandiri dan empat kelas dependen. Nilai faktor modularitas sebesar 1/5 atau 20%, dengan kesimpulan bahwa perangkat lunak mempunyai nilai independensi modul yang sangat buruk.

5) *Konsistensi*: Tes ini sama seperti yang dijelaskan pada tes metrik Kebenaran sebelumnya. Jadi nilai konsistensinya adalah 4/4 dari 100% dengan kesimpulan perangkat lunak mempunyai konsistensi yang sangat baik antara dokumen analisis dengan sistem. Berdasarkan hasil pengujian kelima faktor Reliabilitas maka hasil akhir dapat dihitung seperti persamaan 4.

$$\begin{aligned} \bar{y}_{\text{keandalan}} &= \frac{82,75 + 70 + 89,65 + 20 + 100}{5} \\ &= 72,48\% = \bar{y}_{\text{keandalan}} \end{aligned} \quad (4)$$

Dari hasil perhitungan dapat disimpulkan bahwa nilai Reliability perangkat lunak adalah baik [10].

C. Efisiensi

Pengujian ini bertujuan untuk mengetahui jumlah sumber daya komputasi dan kode program yang dibutuhkan perangkat lunak untuk menjalankan fungsinya. Metrik ini memiliki tiga faktor yaitu Ringkas, Efisiensi Eksekusi, dan Operabilitas.

1) *Ringkas* : Faktor ini merupakan pengujian untuk mengetahui keringkas program sesuai dengan ukuran perintah (LOC) [8]. Tes ini menggunakan 4 dari total 67 kelas yang mempunyai 3419 LOC dengan menggunakan perhitungan persamaan 6. Untuk menentukan jumlah kelas yang digunakan dapat dengan persamaan 5.

$$\bar{y}_{\text{ringkas}} = \frac{67}{3419} = 0,0196 \quad (5)$$

$$\begin{aligned} \bar{y}_{\text{ringkas}} &= \frac{67}{3419} = 0,0196 \\ \bar{y}_{\text{ringkas}} &= \frac{0,0196}{0,49} = 49\% = 100\% \bar{y}_{\text{ringkas}} \end{aligned} \quad (6)$$

Hasil uji keringkas sebesar 49% menunjukkan rata-rata kepadatan LOC cukup baik.

2) *Efisiensi Eksekusi* Faktor ini digunakan untuk menghitung efisiensi run-time perangkat lunak ketika digunakan pada perangkat [6]. Perhitungannya meliputi RAM yang digunakan saat instalasi hingga software dijalankan. Pengujian dilakukan menggunakan fitur realtime profilers pada Android Studio dengan persamaan 7.

$$\bar{y}_{\text{efisiensi}} = \frac{12,8}{100} = 12,8\% \quad (7)$$

Dari persamaan 7 nilainya sebesar 12,8%. Artinya perangkat lunak tersebut cukup efisien [8].

3) *Operabilitas*: Faktor ini menguji kemudahan pengoperasian perangkat lunak. Nilai faktor ini diperoleh dengan menyebarkan kuesioner seperti dijelaskan pada bab 2. Hasilnya sebesar 83,2% dengan kesimpulan pengguna sangat setuju dengan kemudahan penggunaan perangkat lunak. Berdasarkan hasil pengujian ketiga faktor Reliabilitas maka dapat dihitung persentase hasil akhir sebagai berikut:

$$\begin{aligned} \bar{y}_{\text{efisiensi}} &= \frac{83,2 + 12,8 + 49\%}{3} \\ &= 48,3\% = \bar{y}_{\text{efisiensi}} \end{aligned} \quad (8)$$

Dapat disimpulkan bahwa nilai Efisiensi perangkat lunak masih cukup baik.

D. Integritas

Pengujian ini bertujuan untuk mengetahui jumlah sumber daya komputasi dan kode program yang dibutuhkan perangkat lunak untuk menjalankan fungsinya. Metrik ini memiliki tiga faktor yaitu Audibilitas, Instrumentasi, dan Keamanan.

1) *Audibilitas* : Faktor ini untuk menguji apakah perangkat lunak memenuhi standar yang telah ditentukan [8]. Dari 39 fitur yang diuji, 31 fitur telah sesuai dengan standar yang ditentukan. Perhitungan nilai audibilitas menggunakan persamaan sembilan sebagai berikut:

$$\begin{aligned} \bar{y}_{\text{audibilitas}} &= \frac{31}{39} = 79,49\% = 100\% \bar{y}_{\text{audibilitas}} \end{aligned} \quad (9)$$

Hasil pengujian audibilitas sebesar 79,49% fitur yang memenuhi standar yang telah ditentukan dikategorikan baik.

2) *Instrumentasi*: Faktor ini digunakan untuk memantau dan mengidentifikasi kesalahan pengoperasian program [8]. Terdapat 41 fitur yang diuji, 36 fitur diantaranya mengidentifikasi kesalahan pengoperasian program. Perhitungannya dilakukan dengan menggunakan rumus berikut:

$$\text{Persamaan 10} = \frac{\bar{y}}{y} \times 100\% \quad (10)$$

$$\frac{87,8}{100} \times 100\% = 87,8\%$$

Dari persamaan 10 diperoleh nilai sebesar 87,8% yang berarti perangkat lunak sangat baik dalam mengidentifikasi pengoperasian program kesalahan.

3) *Keamanan*: Faktor ini menguji ketersediaan mekanisme untuk mengontrol dan melindungi perangkat lunak dan data dari pihak yang tidak berwenang [8]. Terdapat lima aktivitas keamanan yang diuji dengan hasil 4 fitur yang memiliki keamanan.

Dengan menggunakan persamaan tersebut, nilai keamanan dihitung sebagai berikut:

$$\text{Persamaan 11} = \frac{\bar{y}}{y} \times 100\% \quad (11)$$

$$\frac{4}{5} \times 100\% = 80\%$$

Kesimpulan dari pengujian keamanan bagus. Selanjutnya berdasarkan hasil pengujian ketiga faktor Integritas diperoleh persentase hasil akhir dengan menggunakan persamaan 12.

$$\text{Persamaan 12} = \frac{79,49\% + 87,8 + 80}{3} \quad (12)$$

$$82,43\% = \text{Persamaan 12}$$

Dapat disimpulkan bahwa nilai integritas perangkat lunak sangat baik.

E. Kegunaan

Faktor ini untuk mempelajari, mengoperasikan, menyiapkan masukan, dan menafsirkan keluaran perangkat lunak [6]. Ada dua metrik untuk pengujian ini, yaitu Operabilitas dan Pelatihan.

1) *Operabilitas*: Metrik ini digunakan untuk menguji kemudahan pengoperasian perangkat lunak. Pengujian ini telah dibahas pada Faktor Efisiensi sebelumnya dan mempunyai nilai yang sama yaitu 83,2% yang berarti pengguna sangat setuju bahwa perangkat lunak ini mudah digunakan.

2) *Pelatihan*: Faktor ini untuk mengetahui sejauh mana perangkat lunak membantu pengguna baru memahami sistem [8]. Menunya ada lima, 4 diantaranya ada pelatihannya. Hasilnya dihitung menggunakan persamaan 13.

$$\text{Persamaan 13} = \frac{\bar{y}}{y} \times 100\% \quad (13)$$

$$\text{Persamaan 13} = \frac{4}{5} \times 100\% = 80\%$$

Hasil pengujian perangkat lunak sudah baik dalam memberikan kemudahan pemahaman oleh pengguna baru.

Berdasarkan hasil pengujian kedua faktor kegunaan tersebut, diperoleh persentase hasil akhir dengan menggunakan persamaan 14.

$$\text{Persamaan 14} = \frac{83,2\% + 80}{2} = 81,6\% \quad (14)$$

Kesimpulan dari aspek usability adalah perangkat lunak ini sangat baik dalam memberikan kemudahan penggunaan.

Dari hasil perhitungan diperoleh nilai kebenaran metrik sebesar 89,85%, nilai reliabilitas metrik sebesar 72,48%, nilai efisiensi metrik sebesar 48,3%, nilai integritas metrik sebesar 82,43%, dan nilai kegunaan metrik sebesar 81,6%. Setelah diperoleh seluruh nilai dari kelima aspek tersebut maka dapat dihitung faktor kualitasnya menggunakan persamaan satu sehingga diperoleh hasil sebesar 74,932% dan masuk dalam kategori baik [5]. Hasil ini sama dengan penelitian sebelumnya yang mempunyai hasil kategori baik [7]-[11].

IV. KESIMPULAN

Dari serangkaian uji kualitas yang telah dilakukan, dapat disimpulkan bahwa metode McCall dapat diterapkan untuk menguji kualitas perangkat lunak e-voting. Dari kelima faktor Product Operation yang terfokus secara umum mempunyai hasil yang baik berdasarkan teori kualitas McCall. Metrik dengan kualitas terbaik adalah ketepatan dengan nilai 89,85%, dan metrik dengan kualitas paling rendah adalah efisiensi dengan nilai 48,3%. Hal ini menunjukkan bahwa perangkat lunak e-voting masih kurang dari segi efisiensi, terutama dari segi keringkasan, efisiensi pelaksanaan, dan pengoperasian. Dengan mengetahui kekurangan software e-voting, penelitian lebih lanjut dapat mengatasi kekurangan tersebut.

PENGAKUAN

Penulis mengucapkan terima kasih kepada ISB Atma Luhur atas mendanai penelitian ini agar dapat terselesaikan dengan baik.

REFERENSI

- [1] A. Basu, *Jaminan Kualitas Perangkat Lunak, Pengujian, dan Metrik*. PHI Pembelajaran Pvt., 2015.
- [2] A. Hidayati, E. Oktariza, F. Rosmaningsih, dan SA Lathifah, "Analisa Kualitas Perangkat Lunak Sistem Informasi Akademik Menggunakan McCall," *Multinetics*, vol. 3, tidak. 1, hal. 47–51, 2017, doi: 10.32722/vol3.no1.2017.pp48-53.
- [3] FP Juniawan, "Implementasi RSA untuk Keamanan Transmisi Data pada Aplikasi E-voting Ketua BEM Berbasis Android," dalam *Prosiding - 2016 1st International Conference on Information Technology, Information Systems and Electrical Engineering, ICITISEE 2016*, 2016, 93–98, DOI: 10.1109/ICITISEE.2016.7803054.
- [4] DP Mandala dan A. Dewanto, "Uji Kelayakan Sistem Informasi Unit Kesehatan Sekolah Berbasis Website Di Smk Muhammadiyah 1 Bantul Dengan Faktor Kualitas McCall," *Elinvo (Elektronik, Informatika, Vocat. Educ.*, vol. 2, no. 2, hal.195–203, 2017, doi: 10.21831/elinvo.v2i2.17314.
- [5] C. Juliane, R. Dzulkarnaen, dan W. Susanti, "Metode McCall 's untuk Pengujian Kualitas Sistem Informasi Administrasi Tugas Akhir (SIATA)," *J. Resti*, vol. 3, tidak. 3, hal.488–495, 2019.
- [6] B. Sugiantoro dan M. Mustakim, "Analisa Usabilitas Sistem Deteksi Akses Pornografi Pengguna Internet Menggunakan Metode McCall'S,"

J. Informasi Online., vol. 2, tidak. 1, hal. 56–62, 2017, doi: 10.15575/join.v2i1.60.

- [7] A. Andrianti, "Pengukuran Kualitas Aplikasi Rekap Indikator Mutu Harian RS Bhayangkara Jambi Menggunakan Metode McCall," vol. 14, tidak. 1, hlm. 24–34, 2020.
- [8] M. Agus, "Pengujian Sistem Informasi Akademik menggunakan McCall's Software Quality Framework," *J. Inform. Sunan Kalijaga*, vol. 1, tidak. 1, hal. 47–57, 2016.
- [9] AH Arif dan MU Siregar, "Analisis Kualitas Penggunaan Software McCall pada Sistem Bonus PT Surya Pratama Alam," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 3, tidak. 1, hal.63–72, 2018, DOI: 10.14421/jiska.2018.31-07.
- [10] F. Gunadi dan SR Widiyanto, "Evaluasi Kualitas Pelaporan Manajemen pada Sistem Epicor Perusahaan Manufaktur Berbasis McCall," *J. MULTINETICS*, vol. 6, tidak. 1, hlm. 21–31, 2020.
- [11] D. Mahdiana dan N. Kusumawardhany, "Kombinasi Proses Hierarki Analitik dan Teknik Pemingkatan Multi Atribut Sederhana untuk Pemilihan Dosen Terbaik," *Prosiding - ICoSTA 2020 2020 Int. Konf. Teknologi Cerdas. Aplikasi. Memberdayakan. Ind. IoT oleh Implement. Teknologi Hijau. Mempertahankan. Pengembangan*, 2020, DOI: 10.1109/ICoSTA48221.2020.1570615695.