

IMPLEMENTASI WHITEBOX TESTING DENGAN TEKNIK *BASIS PATH* PADA PENGUJIAN HALAMAN PENCARIAN PROGRAM PROMO

Yuanita Pratiwi^{*1}, Linda Wahyu Widianti²

^{1,2}Magister Teknologi Informasi, Fakultas Ilmu Komputer, STMIK Jakarta STI&K, Jakarta, Indonesia

Email: ¹yuanitapратиwi76@gmail.com, ²linda_wwidiанти@staff.jak-stik.ac.id

(Diterima : 12 Januari 2025, Direvisi : 10 Mei 2025, Disetujui : 21 Mei 2025)

Abstrak

Aplikasi Monitoring Promo merupakan sistem berbasis web milik salah satu perbankan BUMN yang digunakan untuk memantau program promo yang dibuat oleh kantor cabang di 18 wilayah di seluruh Indonesia. Aplikasi ini memfasilitasi pemantauan status promo, realisasi biaya, serta efektivitas promo dalam meningkatkan sales volume. Untuk memastikan aplikasi berfungsi dengan baik dan sesuai harapan, dilakukan pengujian perangkat lunak. Pengujian ini bertujuan menemukan *bug*, cacat, dan kekurangan, serta meningkatkan kinerja sistem. Salah satu metode pengujian yang digunakan adalah *WhiteBox* dengan teknik basis path. Teknik ini menganalisis jalur independen dalam kode program melalui pembuatan *flowgraph* dan perhitungan *Cyclomatic Complexity* untuk menentukan tingkat kompleksitas sistem. Semakin rendah nilai kompleksitas, semakin kecil kemungkinan terjadinya kesalahan. Fokus pengujian ini adalah form pencarian program promo, yang memungkinkan pencarian promo yang sedang berlangsung maupun yang telah selesai. Proses pengujian dilakukan dengan membandingkan hasil aktual dengan hasil yang diharapkan melalui tabel test case. Hasil menunjukkan bahwa tidak ditemukan *error* pada form pencarian, sehingga seluruh pengujian dinyatakan berhasil. Hasil ini menjadi acuan bahwa form pencarian telah berfungsi dengan baik dan sesuai harapan pengguna, serta dapat digunakan sebagai dasar dalam perbaikan dan pengembangan sistem lebih lanjut.

Kata kunci: aplikasi monitoring promo, *basis path*, *cyclomatic complexity*, *test case*, *whitebox testing*.

WHITE BOX TESTING WITH PATH-BASED TECHNIQUES IN TESTING THE SEARCH PAGE OF THE WEBSITE PROMO MONITORING PROGRAM

Abstract

Promo Monitoring Application is a web-based system owned by a state-owned bank used to monitor promotional programs created by branch offices in 18 regions across Indonesia. This application facilitates monitoring the status of promos, realization of promo expenses, and the effectiveness of promos in increasing sales volume. To ensure the application functions properly and meets expectations, software testing is conducted. This testing aims to identify bugs, defects, and shortcomings, as well as improve system performance. One of the testing methods used is *WhiteBox* testing with the basis path technique. This technique analyzes independent paths in the program code by creating a *flowgraph* and calculating the *Cyclomatic Complexity* to determine the system's complexity level. The lower the complexity value, the smaller the likelihood of errors. The focus of this testing is the promo program search form, which allows searching for ongoing and completed promos. The testing process compares actual results with expected results using a test case table. The results show no errors were found in the search form, so all tests were successful. These results serve as a reference that the search form functions well and meets user expectations, and can be used as a basis for further system improvement and development.

Keywords: application monitoring promo, *basis path*, *cyclomatic complexity*, *test case*, *whitebox testing*.

1. PENDAHULUAN

Perangkat lunak hadir dengan memberikan berbagai macam jenis layanan yang berbeda [1], layanan yang memberikan hiburan, layanan yang memudahkan aktivitas kita, atau layanan yang membantu pekerjaan kita. Sering kali kita menjadi tergantung pada perangkat lunak [2]. Namun tidak semua perangkat lunak yang dikembangkan memiliki kualitas yang baik, dikarenakan tidak melalui proses pengujian perangkat lunak, sehingga terdapat banyak *bug* dan *error* pada suatu perangkat lunak. Seperti yang terjadi pada Therac-25, yang merupakan alat pengolah radiasi

yang berfungsi untuk pengobatan pasien kanker, kemudian ternyata pada alat ini terdapat bug yang serius, dimana alat ini gagal berfungsi dengan seharusnya dan menyebabkan dosis radiasi 10 kali lebih tinggi dari seharusnya, yang menyebabkan pasien keracunan radiasi dan bahkan kehilangan nyawa [3].

Website monitoring promo merupakan sebuah website aplikasi yang memiliki tujuan untuk membantu dan memudahkan pengguna dalam melakukan pencarian program promo yang sudah dibuat oleh user dari kantor wilayah. Untuk mengetahui apakah website monitoring promo sudah dibangun dengan baik, benar dan juga terbebas dari *error* yaitu dengan melakukan testing atau pengujian perangkat lunak. Pengujian perangkat lunak merupakan metode untuk melakukan pengujian apakah perangkat lunak sudah memenuhi kebutuhan yang diharapkan dan juga untuk memeriksa apakah produk perangkat lunak bebas cacat atau *error*. Pengujian merepresentasikan ketidaknormalan yang terjadi pada pengembangan perangkat lunak [4]. Pengujian perangkat lunak merupakan serangkaian proses yang dirancang untuk memastikan kode program sudah melakukan sesuai dengan apa yang telah dirancang [5].

Dalam melakukan pengujian perangkat lunak ada 2 metode yang biasanya digunakan yaitu *whitebox* dan *blackbox testing*. *Blackbox testing* merupakan metode yang menguji fungsionalitas suatu perangkat lunak tanpa pengetahuan tentang rincian implementasi dan kode program perangkat lunak tersebut [6]. Sedangkan *whitebox testing* merupakan metode yang menguji struktur internal perangkat lunak, rancangan dan kode program perangkat lunak terkait [7]. *WhiteBox* dapat mengungkapkan kesalahan dalam implementasi dari sebuah perangkat lunak [8]. Pengujian yang menggunakan metode *whitebox* dalam pengujian perangkat lunak harus memiliki pengetahuan atau pemahaman penuh mengenai sumber kode perangkat lunak.

Beberapa penelitian terdahulu mengenai pengujian website menggunakan *whitebox testing* dengan Teknik path diantaranya, penelitian [8] membahas pengujian sistem informasi perpustakaan berbasis web menggunakan metode *Basis Path Testing*, salah satu teknik dalam *WhiteBox Testing*. Tujuannya adalah untuk memastikan alur logika program berjalan dengan benar dan mendeteksi kesalahan (*error*) sebelum sistem digunakan. Dengan membuat *flow graph* dan menghitung *cyclomatic complexity*, peneliti menentukan jalur-jalur independen yang harus diuji. Hasil menunjukkan bahwa metode ini efektif dalam menemukan *bug*, terutama pada bagian login dan dashboard. Meski demikian, pengujian ini tidak sepenuhnya menjamin bebas *bug*, sehingga disarankan untuk melengkapinya dengan pengujian *BlackBox* agar meningkatkan kualitas sistem. [9] Penelitian ini membahas penggunaan *WhiteBox Testing* dengan teknik *Basis Path* pada aplikasi *Cash Flow* berbasis Android. Pengujian dilakukan untuk memastikan alur program bekerja dengan benar dengan cara membuat *flowgraph*, menghitung *cyclomatic complexity* (CC), dan menentukan jalur-jalur independen yang harus diuji. Hasil menunjukkan bahwa semua fungsi seperti menyimpan data pengeluaran, kategori, dan mendapatkan nama bulan berjalan sesuai harapan tanpa *error*. Meski demikian, disarankan untuk melengkapi pengujian menggunakan metode *BlackBox* agar meningkatkan jaminan kualitas aplikasi. Penelitian [10] Penelitian ini membahas pengujian form login pada sebuah aplikasi menggunakan metode *WhiteBox Testing* dengan teknik *Basis Path*. Pengujian dilakukan dengan membuat *flowgraph*, menghitung *cyclomatic complexity* (CC), dan menentukan jalur-jalur independen untuk memastikan semua logika program diuji secara menyeluruh. Hasil menunjukkan bahwa tidak ditemukan *error* selama pengujian pada form login. Semua skenario seperti input username dan password yang benar, salah, atau kosong berjalan sesuai harapan. Meski demikian, sistem masih menyimpan data dalam file teks biasa tanpa enkripsi, sehingga disarankan untuk menggunakan database terenkripsi guna meningkatkan keamanan dan kualitas aplikasi.

Dalam penelitian ini, metode yang digunakan adalah *WhiteBox Testing*, yaitu pendekatan pengujian perangkat lunak yang bertujuan untuk mengevaluasi struktur internal serta logika alur program. *WhiteBox Testing* memiliki berbagai teknik pengujian, di antaranya *Loop Testing* [11], yang berfokus pada validasi struktur kontrol perulangan; *Data Flow Testing* [12], yang mengamati pergerakan data dalam suatu program; *Control Flow Testing*, yang memanfaatkan aliran kontrol program sebagai model acuan dalam pembentukan *test case*; *Branch Testing*, yang menitikberatkan pengujian pada percabangan logika program; serta *Basis Path Testing*, yang menjadi teknik utama dalam penelitian ini. *Basis Path Testing* dipilih karena mampu menghasilkan jumlah *test case* dengan cakupan pengujian yang lebih menyeluruh dibandingkan teknik lainnya [13]. Teknik ini memungkinkan perancang *test case* untuk mengukur kompleksitas logika dari desain prosedural dan menggunakan ukuran tersebut sebagai dasar untuk menentukan jalur eksekusi dasar yang harus diuji. Komponen utama dalam *Basis Path Testing* meliputi *Flow Graph Notation*, yaitu notasi sederhana yang menggambarkan alur kontrol program; *Cyclomatic Complexity*, yaitu metrik untuk menghitung jumlah jalur independen dalam program; *Independent Path*, yaitu jalur eksekusi yang dilewati setidaknya satu kali selama pengujian; serta *Test Case*, yaitu skenario pengujian yang disusun berdasarkan jalur independen tersebut, lengkap dengan definisi output yang diharapkan [14], [15].

Form pencarian program promo merupakan salah satu komponen aplikasi yang sangat penting, karena memungkinkan pengguna untuk mencari informasi mengenai program promo yang belum dimulai, sedang berlangsung, maupun yang sudah selesai. Untuk memastikan bahwa aplikasi dapat berfungsi dengan baik dan sesuai harapan, diperlukan pengujian sistematis. Metode *WhiteBox Testing* dipilih dalam penelitian ini guna melakukan evaluasi terhadap logika internal aplikasi, berbeda dengan *BlackBox Testing* yang lebih fokus pada aspek fungsionalitas dari perspektif pengguna. Teknik *Basis Path Testing* digunakan untuk mengukur kompleksitas logika

program [7] dan membantu dalam merancang skenario pengujian yang efektif sehingga meningkatkan kualitas dan keandalan sistem yang dikembangkan.

2. METODE PENELITIAN

2.1. PHP (*Hypertext Preprocessor*)

PHP merupakan suatu bahasa pemrograman tingkat tinggi yang berorientasi objek dan dapat dijalankan di berbagai sistem operasi karena bersifat *multi-platform*. PHP banyak digunakan untuk membuat berbagai aplikasi dan *script* untuk otomatisasi. PHP sekarang sudah sangat populer dan komunitas pengguna bahasa pemrograman ini sudah tersebar di seluruh dunia. PHP juga bersifat *opensource* yang berarti *source code*-nya terbuka untuk umum sehingga semua orang dapat berkontribusi dalam mengembangkan bahasa PHP.

2.2. Pengujian *WhiteBox*

Pengujian *White Box*, adalah suatu metode pengujian aplikasi yang menggunakan penjelasan struktur kontrol sebagai bagian dari *component-level design* untuk membuat *test cases*. *White Box* sendiri mempunyai beberapa teknik di dalam pengujiannya, seperti : *Data Flow Testing*, *Control Flow Testing*, *Basic Path / Path Testing*, dan *Loop Testing* [8]. Dalam Pengujian *White Box* para penguji perlu mengetahui secara dalam *source code* yang akan diuji. Pengujian *White Box* dapat mengungkap kesalahan implementasi dari sebuah aplikasi. Pengujian ini dapat diterapkan pada tingkatan integrasi, unit dan sistem [7]. Ada beberapa kelebihan dan kekurangan dalam pengujian menggunakan metode *WhiteBox* antara lain:

Kelebihan:

- Metode *WhiteBox* dapat memperlihatkan galat pada kode yang dibuat dengan menghapus baris yang tidak diperlukan.
- Maksimalnya cakupan pengujian aplikasi saat uji coba sebuah skenario.

Kekurangan:

- Biaya pengujian menggunakan metode *WhiteBox* sangatlah tinggi karena dibutuhkan penguji yang berpengalaman dalam bidang ini
- Beberapa alur program akan dibiarkan tidak diuji karena tidak memungkinkan untuk menguji setiap baris kode untuk menemukan kesalahan

2.3. Teknik *Basis Path*

Teknik *Basis Path* merupakan salah satu teknik Pengujian *WhiteBox* yang pertama kali dikemukakan oleh Tom McCabe. Teknik ini memungkinkan penguji untuk mengukur kompleksitas logika dari rancangan prosedural. Skenario uji coba yang dibuat untuk menguji Teknik *Basis Path* ini dijamin akan mengeksekusi setiap statement dari aplikasi yang diujikan setidaknya satu kali saat tahap pengujian [14].

Cyclomatic Complexity

Cyclomatic Complexity adalah besaran perangkat lunak menyediakan acuan kuantitatif kompleksitas suatu logika dalam program. Rumus menghitung *Cyclomatic Complexity* adalah sebagai berikut [11].

$$V(G) = E - N + 2$$

Keterangan :

E = jumlah *edges* pada *flowgraph*

N = jumlah *nodes* pada *flowgraph*

P = jumlah *predicates nodes* pada *flowgraph Flowchart*

Flowgraph

Flowgraph adalah grafik program yang dihasilkan dari pemetaan *flowchart* program yang ada untuk merepresentasikan aliran kontrol logika program yang ada [11]. *Flowgraph* digunakan pada tahapan pengujian yang berfokus pada penggambaran aliran dari sebuah program [14].

Penelitian ini dilakukan dengan pengujian secara langsung terhadap aplikasi yang telah dirancang dengan pendekatan kuantitatif. Oleh karena itu hasil dari penelitian ini sudah dilakukan hingga akhir untuk mengumpulkan fakta sebagai bukti penelitian. Penelitian ini dibagi menjadi 2 tahapan yaitu:

- Pengujian Form Pencarian

aplikasi ini dimulai dengan menghitung jumlah skenario yang akan diuji dengan menggunakan *Cyclomatic Complexity* dengan rumus $V(G) = E - N + 2$ dengan melihat hasil pembuatan *flowgraph*. Setelah mendapatkan nilai *Cyclomatic Complexity* maka dibuat skenario uji. Pengujian menggunakan dilakukan secara manual dengan melakukan pengujian satu per satu dan melihat hasil keluaran yang muncul dari setiap jalur independen tersebut.

b. Analisis Form Login

Selanjutnya untuk tahap analisis dibuat setelah percobaan dengan menggunakan tabel *test case*. Bentuk header format tabel *test case* antara lain :

1. Kegiatan
2. Hasil yang diharapkan
3. Hasil
4. Keterangan

Setelah selesai maka dapat ditarik kesimpulan dari penelitian ini dalam pengujian aplikasi form login menggunakan metode *WhiteBox*.

3. HASIL DAN PEMBAHASAN

3.1. Source Code Validasi

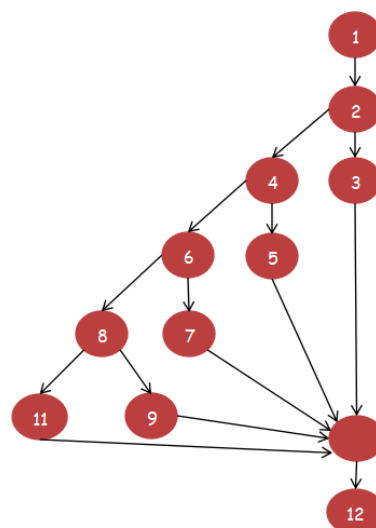
Dalam pembuatan aplikasi form pencarian program promo ini, aplikasi melakukan validasi dengan memeriksa inputan yang di masukkan. Secara umum *source code function* pencarian ditampilkan pada gambar 1.

```
function cari_promo() {  
    $tahun2 = $_POST['tahun2'];  
    $jenis_program = $_POST['jenis_program'];  
    $wilayah_rdb = $_POST['wilayah_rdb'];  
    $nama_program = $_POST['nama_program'];  
    $cariNotin = $_POST['cariNotin'];  
  
    // Validasi input  
    if (!empty($tahun2) && !empty($jenis_program)) {  
        $print_list = testSearchByYearAndTypeProgram();  
    } elseif (!empty($tahun2) && !empty($wilayah_rdb)) {  
        $print_list = testSearchByYearAndRegion();  
    } elseif (!empty($nama_program)) {  
        $print_list = testSearchByProgramName();  
    } elseif (!empty($cariNotin)) {  
        $print_list = testSearchByNotinNumber();  
    } else {  
        $print_list = testSearchByYearNow();  
    }  
  
    echo $print_list;  
}
```

Gambar 1. Function Pencarian Program Promo

3.2. Flowgraph Aplikasi Form Pencarian

Tujuan dibuatnya flowgraph ini untuk menghitung *Cyclomatic Complexity* yang membutuhkan jumlah *node* dan jumlah *edge* dari suatu *flowgraph*.



Gambar 2. FlowGraph Form Pencarian Program Promo

3.3. Menghitung Jalur Independen Menggunakan Cyclomatic Complexity

Untuk menghitung CC digunakan formula:

$$V(G) = E - N + 2$$

Dari hasil pembuatan *flowgraph* sebelumnya dapat diketahui nilai berikut:

$$E = 15 \text{ dan } N = 12$$

Setelah nilai tersebut dimasukan ke dalam formula akan menghasilkan:

$$\begin{aligned} V(G) &= E - N + 2 \\ V(G) &= 15 - 12 + 2 \\ V(G) &= 5 \end{aligned}$$

Dari hasil perhitungan tersebut didapatkan 5 jalur independen yaitu:

- a. 1-2-3-12 (Skenario pencarian berdasarkan tahun dan jenis program)
- b. 1-2-4-5-12 (Skenario pencarian berdasarkan tahun dan wilayah regional)
- c. 1-2-4-6-7-12 (Skenario pencarian berdasarkan nama program promo)
- d. 1-2-4-6-8-9-12 (Skenario pencarian berdasarkan nomor notin)
- e. 1-2-4-6-8-11-12 (Skenario pencarian default bila tidak memilih kriteria)

Setelah selesai menentukan jalur independen tahap berikutnya, membuat tabel *test case* dan bandingkan hasil luaran nyata dengan hasil yang diharapkan. Tabel *test case* yang telah dibuat ditampilkan pada tabel 2.

Tabel 1. *Test case* Form Pencarian Program Promo

No	Kegiatan	Hasil yang diharapkan	Hasil	Keterangan
1.	Memilih kriteria tahun dan jenis program	Muncul nama program promo dan detail berdasarkan pemilihan tahun dan jenis program	Muncul nama program promo dan detail berdasarkan pemilihan tahun dan jenis program	<i>valid</i>
2.	Memilih kriteria tahun dan wilayah	Muncul nama program promo dan detail berdasarkan pemilihan tahun dan wilayah regional	Muncul nama program promo dan detail berdasarkan pemilihan tahun dan wilayah regional	<i>valid</i>
3.	Menginput kriteria nama program promo	Muncul nama program promo dan detail berdasarkan nama program yang diinput	Muncul nama program promo dan detail berdasarkan nama program yang diinput	<i>valid</i>
4.	Menginput kriteria nomor notin	Muncul nama program promo dan detail berdasarkan nomor notin yang diinput	Muncul nama program promo dan detail berdasarkan nomor notin yang diinput	<i>valid</i>
5.	Tidak menginput kriteria apapun	Muncul nama program promo dan detail berdasarkan tahun saat ini	Muncul nama program promo dan detail berdasarkan tahun saat ini	<i>valid</i>

Pada kriteria penginputan pencarian program promo, user memilih kriteria penginputan tahun dengan *value* tahun 2024 dan memilih jenis program yaitu *regular*, maka akan tampil list semua program promo yang penginputan di tahun 2024 dan berjenis *regular*, seperti pada gambar 3.

Yuanita pratiwi, dkk, implementasi whitebox testing dengan teknik basis path pada pengujian halaman pencarian program promo

N	NAMA PROGRAM	NO NOTIN	JEN.	WILAYAH	TARGET PROG..
1	Promo Discount 10% di Inhara Resto	R08/09/649/2024	Reguler	W09 - Banjarmasin	45,560,008
2	Promo Berybenka Potongan Up to Rp.500 ribu periode Nov-Des 2024	R08/09/222/2024	Reguler	W09 - Banjarmasin	70,000,000
3	BNi AUTOMOTIVE EXPERIENCE WITH BNi CREDIT CARD	W05/16/763	Reguler	W03 - Palembang	53,270,825
4	Promo Cashback 100 Rb di Diamond	R08/09/649/2024	Reguler	W09 - Banjarmasin	1,500,000,000
5	Promo Sinarco Periode April 2024	W03/12/123	Reguler	W03 - Palembang	500,000,000
6	Promo Setiabudi Supermarket Periode Maret & April 2024	W03/12/124	Reguler	W03 - Palembang	800,000,000
7	Promo Yava Mart Periode Maret & April 2024	W03/12/125	Reguler	W03 - Palembang	550,000,000
8	Promo Nataru Restoran di Banjarmasin Periode Desember 2024	R08/09/123/2024	Reguler	W09 - Banjarmasin	35,000,000
9	Promo Voucher Belanja Rp100 Rb di Pasar Buah Pekanbaru	W03/15/1234	Reguler	W03 - Palembang	800,000,000

Gambar 3. Tampilan berdasarkan pemilihan tahun 2024 dan jenis program Reguler

Pada kriteria penginputan pencarian program promo, user memilih kriteria penginputan tahun dengan *value* tahun 2024 dan memilih wilayah yaitu palembang, maka akan tampil list semua program promo yang penginputan di tahun 2024 dan promo di wilayah palembang, seperti pada gambar 4.

N	NAMA PROGRAM	NO NOTIN	JENIS	WILAYAH	TARGET PROG..
1	Promo Diskon Rp.400 Ribu Erlane	W03/18/123	Clickan	W03 - Palembang	10,000,000
2	Program BNi Reward Point Botik Air Periode Desember 2024	R08/03/123/2024	BNi Reward Point	W03 - Palembang	120,000,000
3	Program Nataru Happy Fresh Periode Desember 2024	R08/03/1234/2024	BNi Reward Point	W03 - Palembang	40,000,000
4	BNi AUTOMOTIVE EXPERIENCE WITH BNi CREDIT CARD	W05/16/763	Reguler	W03 - Palembang	53,270,825
5	Program Nataru Agoda Periode Desember 2024	R08/03/11/2024	Taktikal	W03 - Palembang	90,000,000
6	Promo Sinarco Periode April 2024	W03/12/123	Reguler	W03 - Palembang	500,000,000
7	Promo Setiabudi Supermarket Periode Maret & April 2024	W03/12/124	Reguler	W03 - Palembang	800,000,000
8	Promo Yava Mart Periode Maret & April 2024	W03/12/125	Reguler	W03 - Palembang	550,000,000
9	Promo Informa di Palembang Periode Juli - Oktober 2024	W03/12/195	Clickan	W03 - Palembang	7,600,000,000

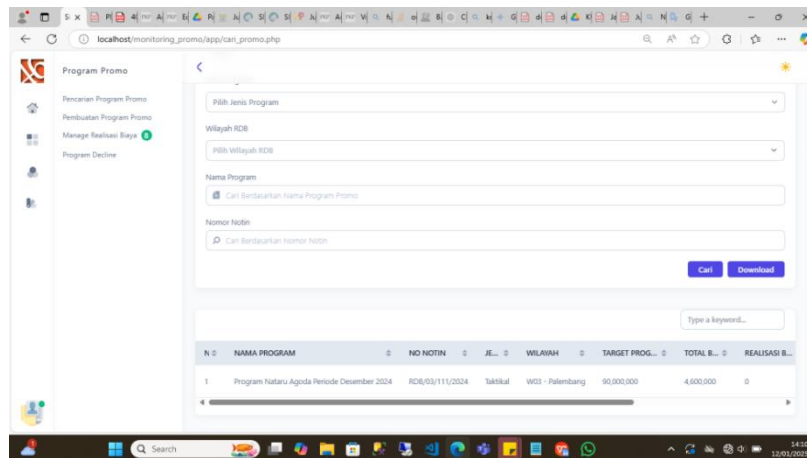
Gambar 4. Tampilan berdasarkan pemilihan tahun 2024 dan wilayah palembang

Pada kriteria penginputan pencarian program promo, user menginputkan *value* nataru pada *textbox* pencarian bagian nama program, maka akan tampil list semua program promo yang mengandung nama program nataru, seperti pada gambar 5.

N	NAMA PROGRAM	NO NOTIN	JENIS	WILAYAH	TARGET PROG..	TOTAL
1	Program Nataru Happy Fresh Periode Desember 2024	R08/03/1234/2024	BNi Reward Point	W03 - Palembang	40,000,000	970,000
2	Program Nataru Agoda Periode Desember 2024	R08/03/11/2024	Taktikal	W03 - Palembang	90,000,000	4,600
3	Promo Nataru Restoran di Banjarmasin Periode Desember 2024	R08/09/123/2024	Reguler	W09 - Banjarmasin	35,000,000	1,000
4	Promo Nataru Chatime Diskon Up To 25%	R08/03/777/2024	Reguler	W03 - Palembang	120,000,000	3,150

Gambar 5. Tampilan berdasarkan penginputan nama program yang mengandung kata nataru

Pada kriteria penginputan pencarian program promo, user menginputkan value RDB/03/111/2024 pada *textbox* pencarian bagian nomor notin, maka akan tampil list semua program promo yang mengandung nomor notin RDB/03/111/2024, seperti pada gambar 6.



Gambar 6. Tampilan berdasarkan penginputan nomor notin

Dari hasil keluaran nyata dan hasil keluaran yang diharapkan dalam *test case* dan sudah benar semua dan tidak ada kesalahan dalam Form Pencarian program promo yang telah dibuat.

4. KESIMPULAN

Pengujian pada aplikasi form pencarian program promo menggunakan metode *WhiteBox - Basis Path* menunjukkan bahwa tidak ada kesalahan yang ditemukan dalam proses pencarian program promo. Hasil ini mencerminkan bahwa alur logika dan implementasi kode pada aplikasi telah dirancang dengan baik sehingga mampu menangani berbagai skenario pencarian sesuai dengan fungsi yang diharapkan. Pengujian ini memastikan bahwa aplikasi dapat diandalkan dalam memberikan hasil yang akurat dan konsisten. Tahapan pengujian yang dilakukan melibatkan langkah-langkah sistematis seperti pemetaan *source code*, pembuatan *flowgraph*, penghitungan *Cyclomatic Complexity*, identifikasi jalur independen, dan pelaksanaan skenario uji berdasarkan rencana yang telah ditentukan. Metode *WhiteBox - Basis Path* yang digunakan tidak hanya berhasil mengidentifikasi semua jalur logika dalam aplikasi tetapi juga memastikan bahwa setiap jalur tersebut berfungsi sebagaimana mestinya. Dengan demikian, pengujian ini memberikan jaminan bahwa aplikasi siap digunakan dan memiliki tingkat kesalahan yang sangat rendah.

DAFTAR PUSTAKA

- [1] Safitri, A., Avini, T., Sugeng. (2024). "Eksplorasi Desain User Interface Online Public Access Catalog (Opac) Perpustakaan Berbasis Website Dengan Figma,". *Jurnal Kecerdasan Buatan dan Teknologi Information*. vol. 3, no. 2, pp. 85–89. doi: 10.69916/jkbt.v3i2.132.
- [2] Triady, D., Musdar, I. A., Surasa, H.. (2023). "Pengujian Blackbox Pada Website Worker'S Menggunakan Metode Equivalence Partitioning," *KHARISMA Tech*, vol. 18, no. 1, pp. 84–98, doi: 10.55645/kharismatech.v18i1.292.
- [3] Syihab, L. N & Sugiarti, Y. (2024). "Pengujian Strategi Perangkat Lunak : Tinjauan Literatur Sistematis Strategy Testing Software : Systematic Literature Review," *Informatech J. Ilm. Inform. dan Komput.*, vol. 1.
- [4] Taley, D. S. (2020). "Comprehensive Study of Software Testing Techniques and Strategies: A Review," *Int. J. Eng. Res.*, vol. V9, no. 08, pp. 817–822. doi: 10.17577/ijertv9is080373.
- [5] Umar, M. A., & Zhanfang, C. (2020). "A Comparative Study Of Dynamic Software Testing Techniques,". *International Journal of Advanced Networking and Applications*. vol. 12, no. 03, pp. 4575–4584. doi: 10.35444/ijana. 12301.
- [6] Izzat, S., & Saleem, N. N. (2023). "Software Testing Techniques and Tools: A Review,". *Journal of Education and Science*. vol. 32, no. 2, pp. 31–40. doi: 10.33899/edusj.2023.137480.1305.
- [7] AL Khamaeni, M. G. (2023). "Implementasi White Box Testing Berbasis Path Pada Aplikasi Berbasis Web," *Jurnal Siliwangi*, vol. 9, no. 1, pp. 8–13.
- [8] Ndaumanu, R. I. (2023). "Pengujian Sistem Informasi Perpustakaan Berbasis Website dengan Basis Path Testing," *Justek Jurnal Sains dan Teknologi*. vol. 6, no. 1, p. 123. doi: 10.31764/justek.v6i1.13808.
- [9] Pratalla, C. T., Asyer, E. M., Prayudi, I., & Saifudin, A. (2020). "Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path,". *Jurnal Informatika Univ. Pamulang*. vol. 5, no. 2, p. 11. doi: 10.32493/informatika.v5i2.4713.
- [10] Londjo, M. F. (2021). "Implementasi White Box Testing Dengan Teknik Basis Path Pada Pengujian Form

- Login,” *Jurnal Siliwaangi*. vol. 7, no. 2, pp. 35–40.
- [11] Dhaifullah, I. R., Muttanifudin H. M., Salsabila, A. A., & Yaqin, M. A. (2022). “Survei Teknik Pengujian Software,” *Jurnal Automation Computer Information System*. vol. 2, no. 1, pp. 31–38. doi: 10.47134/jacis.v2i1.42.
- [12] Fauziah, C., Sawitri, F., Azahara, F. N., & Permatasari, H. (2023). “Implementasi Pengujian Form Transaksi Laporan Penjualan Sistem Kasir Pos Codekop Cv Darut Taqwa Ujung Harapan Menggunakan Metode White Box Testing Dan Black Box Testing,” in *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis (SENATIB)*. pp. 456–464.
- [13] Huda, M., Wiyono, S., Hidayatullah, M. F., & Bahri, S. (2020). “Studi Kasus: Sistem Informasi dan Pelayanan Administrasi Kependudukan,” *Komputika Jurnal Sistem Komputer*. vol. 9, no. 1, pp. 59–65. doi: 10.34010/komputika.v9i1.2518.
- [14] Mintarsih, M. (2023). “Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation,” *Jurnal Teknologi Dan Sistem Informasi Bisnis*. vol. 5, no. 1, pp. 33–35. doi: 10.47233/jteksis.v5i1.727.
- [15] Uminingsih, Ichsanudin, M. N., Yusuf, M., & Suraya, S. (2022). “Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula,” *STORAGE J. Ilmu Teknologi dan Ilmu Komputer*. vol. 1, no. 2, pp. 1–8. doi: 10.55123/storage.v1i2.270.