

PENGUJIAN WHITEBOX PADA SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB DI SMA DAARUL QUR'AN

Muh. Asdar Ahsan¹, Linda Wahyu Widiyanti²

Magister Teknologi Informasi, STMIK Jakarta STI&K
Jalan Bri Radio Dalam No. 17, Jakarta Selatan, Indonesia
linda_wwidiyanti@staff.jak-stik.ac.id

ABSTRAK

Sistem informasi perpustakaan berbasis web di SMA Daarul Qur'an menangani autentikasi, data anggota dan buku, peminjaman, pengembalian, denda, pencatatan pengunjung, serta validasi masukan. Percabangan pada fungsi-fungsi tersebut perlu diperiksa karena pengujian antarmuka belum menunjukkan jalur kode yang benar-benar dilewati. Penelitian ini bertujuan menganalisis struktur logika delapan fungsi melalui *white box testing* dengan teknik *basis path testing* dan *cyclomatic complexity*. Pengujian dilakukan pada aplikasi PHP prosedural dan MySQL di lingkungan *localhost* serta LAN sekolah. Peneliti menelusuri kode, menyusun *flowgraph*, mencatat *node* dan *edge*, menghitung $V(G)$, kemudian menjalankan 20 *test case* dengan bantuan *Xdebug*. Hasil perhitungan menunjukkan 32 jalur independen dengan nilai $V(G)$ antara 3 dan 5. Peminjaman dan pengembalian memiliki nilai tertinggi, masing-masing 5, sedangkan *counter* pengunjung dan validasi form bernilai 3. Seluruh *actual output* pada 20 *test case* sesuai dengan *expected output* sehingga berstatus PASS. Analisis $V(G)$ membantu menentukan prioritas pengujian dan menunjukkan jalur yang masih memerlukan *test case* tersendiri. Hasil berlaku pada lingkungan uji yang digunakan dan belum mencakup pengujian beban serta keamanan tingkat lanjut.

Kata kunci : *Basis Path Testing, Cyclomatic Complexity, Pengujian White Box, Sistem Informasi Perpustakaan, Website*

1. PENDAHULUAN

Sistem informasi perpustakaan berbasis web membantu sekolah mengelola koleksi, anggota, dan transaksi dalam satu aplikasi. Guo et al. [1] menjelaskan bahwa sistem berbasis web dapat menghubungkan pencarian, peminjaman, pengembalian, dan administrasi data.

Ardhana et al. [2] menunjukkan bahwa akses berbasis web mempercepat pengelolaan dibandingkan pencatatan manual. Pada praktiknya, manfaat tersebut sangat bergantung pada ketepatan stok, status pinjaman, riwayat anggota, dan hasil perhitungan denda.

Kebutuhan terhadap layanan perpustakaan yang tertata ikut meningkat bersama kegiatan literasi. Badan Pusat Statistik [3] mencatat tingkat kegemaran membaca masyarakat Indonesia pada 2024 sebesar 72,44, naik dari 66,77 pada tahun sebelumnya. Perpustakaan Nasional Republik Indonesia [4] juga melaporkan peningkatan indeks pembangunan literasi masyarakat.

Sementara itu, Badan Pusat Statistik [5] menunjukkan bahwa perpustakaan sekolah menjadi bagian besar dari perpustakaan terakreditasi. Kondisi ini memperkuat kebutuhan terhadap aplikasi yang mampu menjaga data layanan secara konsisten.

Sistem Informasi Perpustakaan SMA Daarul Qur'an dibangun dengan PHP dan MySQL. Saat menelaah kode, peneliti menemukan rangkaian *if/else* pada autentikasi, validasi anggota, pemeriksaan stok, penentuan pinjaman aktif, perhitungan selisih hari, dan pembaruan *counter*. Pada peminjaman, transaksi hanya boleh disimpan setelah anggota ditemukan, statusnya aktif, dan stok masih tersedia. Pada

pengembalian, program harus memastikan adanya pinjaman aktif, menghitung keterlambatan, menyimpan transaksi, lalu menambah stok. Kesalahan pada satu cabang dapat membuat stok tidak berubah, transaksi diterima untuk anggota yang tidak sah, atau denda dihitung keliru.

Pengujian dari *interface* saja belum cukup untuk menjawab apakah seluruh cabang tersebut benar-benar dilewati. Sebuah pesan penolakan dapat muncul dari beberapa kondisi yang berbeda, sementara satu skenario positif dapat melewati banyak keputusan sekaligus. Karena itu, peneliti menggunakan *white box testing* untuk mengikuti alur internal program. *Basis path testing* dipakai untuk memilih jalur dasar, sedangkan *cyclomatic complexity* digunakan untuk menentukan jumlah jalur independen yang perlu diperhatikan.

Penelitian terdahulu lebih banyak membahas pembangunan dan manfaat sistem perpustakaan. Susanto et al. [6] menilai kemudahan pencarian buku dan transaksi di SMK PGRI Sukodadi Lamongan. Astuti dan Samudra [7] membahas penggunaan *Rapid Application Development* pada aplikasi sekolah, sedangkan Satrio et al.

[8] menjelaskan perancangan modul anggota, buku, dan transaksi. Santoso et al. [9] menekankan hubungan *interface*, struktur data, dan alur proses. Aulia et al. [10] menggunakan model Waterfall untuk membangun sistem perpustakaan sekolah. Kajian tersebut memberi dasar fungsional, tetapi belum menunjukkan bagaimana *node*, *edge*, jalur independen, dan hasil eksekusi kode dipetakan pada sistem yang telah digunakan.

Kebaruan penelitian ini terletak pada pengujian langsung terhadap delapan fungsi operasional melalui pemetaan *flowgraph*, perhitungan $V(G)$, dan pencocokan *expected output* dengan *actual output*. Analisis tidak berhenti pada angka *pass rate*. Setiap nilai $V(G)$ digunakan untuk melihat fungsi yang memerlukan perhatian lebih besar, lalu jalur independen dipasangkan dengan *test case* yang tersedia. Dengan cara tersebut, penelitian menunjukkan dua hal sekaligus: jalur yang telah memiliki bukti eksekusi dan jalur yang masih membutuhkan skenario tersendiri. Penelitian bertujuan mengukur $V(G)$, menjalankan *test case* pada jalur utama, menilai hasil setiap skenario, dan menetapkan prioritas pengujian berikutnya.

2. TINJAUAN PUSTAKA

2.1. Sistem Informasi Perpustakaan Berbasis Web

Sistem informasi perpustakaan menggabungkan data koleksi, anggota, petugas, transaksi, dan laporan. Susanto et al. [6] menyebutkan bahwa pengelolaan berbasis web mempercepat pencarian serta pencatatan peminjaman. Satrio et al. [8] menempatkan data anggota, buku, dan transaksi sebagai komponen yang harus saling terhubung. Ardhana et al. [2] menambahkan bahwa perubahan kebutuhan pengguna tetap memerlukan alur proses yang jelas agar data yang telah tersimpan tidak terganggu.

Mutu aplikasi tidak hanya ditentukan oleh tampilan. Santoso et al. [9] menekankan keterkaitan *interface*, struktur data, dan proses bisnis. Aulia et al. [10] menunjukkan bahwa analisis, desain, implementasi, dan pengujian perlu dijalankan secara berurutan.

Komara et al. [14] juga menghubungkan mutu sistem perpustakaan dengan konsistensi pengolahan data dan kestabilan transaksi. Pengujian struktur kode melengkapi pemeriksaan fungsi karena dapat menunjukkan cabang yang belum pernah dijalankan.

2.2. Kualitas Perangkat Lunak dan Pengujian Struktural

International Software Testing Qualifications Board (ISTQB) [15] membedakan teknik pengujian berdasarkan sumber informasi yang digunakan. Pengujian struktural menyusun *test case* dari struktur internal komponen, seperti pernyataan, keputusan, kondisi, dan alur kendali. Cakupan kemudian dinilai dari bagian kode yang benar-benar dilewati. Pendekatan ini tidak menggantikan pengujian kebutuhan pengguna, tetapi membantu penguji menemukan cabang yang belum tersentuh oleh skenario fungsional.

Pada aplikasi perpustakaan, keluaran yang sama dapat berasal dari jalur yang berbeda. Penolakan peminjaman, misalnya, dapat terjadi karena anggota tidak ditemukan, status anggota tidak aktif, atau stok habis. Jika hanya pesan pada layar yang diperiksa, ketiga kondisi tersebut mudah dianggap sebagai satu

kasus. Pengujian struktural memisahkan keputusan itu agar setiap cabang memperoleh *input* yang sesuai.

2.3. White Box Testing

White box testing memeriksa cara kode menerima *input*, mengambil keputusan, dan menghasilkan *output*. Menurut ISTQB [15], penguji harus memahami struktur program untuk menentukan bagian yang perlu dilewati. Objek yang ditelaah dapat berupa pernyataan, cabang keputusan, kondisi majemuk, perulangan, dan jalur eksekusi. Keunggulannya terletak pada kemampuan menunjukkan bagian kode yang belum dijalankan. Hasilnya tetap bergantung pada ketepatan *flowgraph*, kelengkapan kode, dan mutu data uji.

Dalam penelitian ini, *white box testing* diterapkan pada autentikasi, pengelolaan anggota dan buku, peminjaman, pengembalian, kalkulasi denda, *counter* pengunjung, dan validasi form. Delapan fungsi tersebut dipilih karena memiliki keputusan yang memengaruhi sesi, transaksi, stok, atau nilai yang tersimpan. Dashboard, data karyawan, dan laporan tidak masuk ke pengujian jalur pada penelitian ini.

2.4. Basis Path Testing

Wallace, Watson, dan McCabe [16] menjelaskan *basis path testing* sebagai teknik untuk membentuk himpunan jalur dasar dari *flowgraph*. Sebuah jalur disebut independen apabila memuat sedikitnya satu *edge* yang belum terdapat pada jalur sebelumnya. Jumlah jalur dalam basis set ditentukan oleh *cyclomatic complexity*. Teknik ini membuat pemilihan skenario lebih terarah tanpa harus mencoba seluruh kombinasi jalur yang mungkin.

Penerapan *basis path testing* dimulai dengan membaca kode atau pseudocode, mengelompokkan pernyataan berurutan sebagai *node*, lalu menghubungkannya melalui *edge*. Setelah nilai $V(G)$ diperoleh, setiap jalur independen diterjemahkan menjadi kondisi *input*. Satu *test case* dapat melewati beberapa keputusan, tetapi jalur yang berbeda tetap perlu ditelusuri secara jelas. Karena itu, daftar jalur dan hasil eksekusi harus disajikan bersama agar cakupan pengujian dapat diperiksa.

2.5. Flowgraph

Flowgraph merupakan graf berarah yang menggambarkan perpindahan kendali. *Node* mewakili satu atau beberapa pernyataan yang berjalan berurutan, sedangkan *edge* menunjukkan perpindahan dari satu *node* ke *node* lain. *Node* keputusan mempunyai lebih dari satu *edge* keluar, misalnya cabang benar dan salah pada pernyataan *if*. Wallace et al. [16] menggunakan hubungan *node* dan *edge* tersebut sebagai dasar penghitungan jalur independen.

Flowgraph berbeda dari *flow of document*. *Flowgraph* dibentuk dari struktur kode, sedangkan *flow of document* menggambarkan perpindahan data atau dokumen pada proses bisnis. Perbedaan ini menentukan keabsahan perhitungan $V(G)$: jumlah

node dan *edge* harus berasal dari keputusan di dalam program. Pada penelitian ini, *flowgraph* disusun dengan mengikuti percabangan fungsi yang diuji, terutama pada peminjaman dan pengembalian yang mempunyai nilai $V(G)$ tertinggi.

2.6. Cyclomatic Complexity

McCabe [17] memperkenalkan *cyclomatic complexity* sebagai ukuran jumlah jalur independen dalam graf kontrol. Wallace et al. [16] kemudian menggunakannya sebagai dasar structured testing. Untuk graf dengan P komponen terhubung, rumus yang digunakan adalah:

$$V(G) = E - N + 2P \quad (1)$$

E menyatakan jumlah *edge*, N jumlah *node*, dan P jumlah komponen terhubung. Setiap fungsi pada penelitian ini dibentuk sebagai satu *flowgraph*, sehingga $P=1$ dan rumusnya menjadi

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

Nilai yang lebih besar menandakan lebih banyak jalur dasar yang perlu diperiksa. $V(G)$ bukan jumlah defect dan tidak dapat dipakai sendirian untuk menyatakan bahwa kode baik atau buruk.

Telaumbanua dan Juliandri [11] menempatkan peminjaman serta pengembalian sebagai inti layanan perpustakaan. Pramarta dan Efendi [12] menekankan validasi data pada transaksi, sedangkan Rifqi et al. [13] membahas penerapan validasi pada sistem perpustakaan berbasis web. Penelitian ini mengambil posisi berbeda: fungsi yang sudah digunakan tidak dibangun ulang, tetapi ditelusuri jalur internalnya. Kontribusi yang diberikan berupa nilai $V(G)$ per fungsi, daftar jalur independen, hasil 20 *test case*, serta pemetaan jalur yang belum memiliki skenario mandiri.

3. METODE PENELITIAN

3.1. Rancangan Penelitian

Penelitian ini dilaksanakan sebagai studi kasus deskriptif pada Sistem Informasi Perpustakaan Berbasis Web SMA Daarul Qur'an. Peneliti membuka kode PHP pada setiap fungsi, menandai keputusan yang memengaruhi alur, menyusun *flowgraph*, lalu menghitung *node*, *edge*, dan $V(G)$. Setelah jalur independen ditetapkan, skenario uji dijalankan pada aplikasi dan hasilnya diperiksa melalui *interface*, sesi, serta perubahan *database*.

3.2. Objek dan Lingkungan Pengujian

Tabel 1. Konfigurasi sistem dan lingkungan pengujian

Atribut	Keterangan
Nama sistem	Sistem Informasi Perpustakaan Berbasis Web pada SMA Daarul Qur'an
Platform	Web-based
Bahasa / basis data	PHP / MySQL
Web server	Apache pada XAMPP v3.3.0

Atribut	Keterangan
Browser	Google Chrome dan Mozilla Firefox
Lingkungan	Localhost (offline) dan LAN sekolah
Periode / penguji	2026 / peneliti dan guru atau staf perpustakaan

Pengujian dilakukan di Perpustakaan SMA Daarul Qur'an menggunakan Google Chrome dan Mozilla Firefox. Aplikasi dijalankan melalui Apache pada XAMPP v3.3.0 dengan *database* MySQL, baik pada *localhost* maupun LAN sekolah. *Xdebug* digunakan untuk mengikuti eksekusi kode. *Flowgraph* digambar dengan Draw.io, PHP CodeSniffer membantu pemeriksaan kode, dan hasil pengujian direkap dalam Excel.

3.3. Ruang Lingkup Fungsi

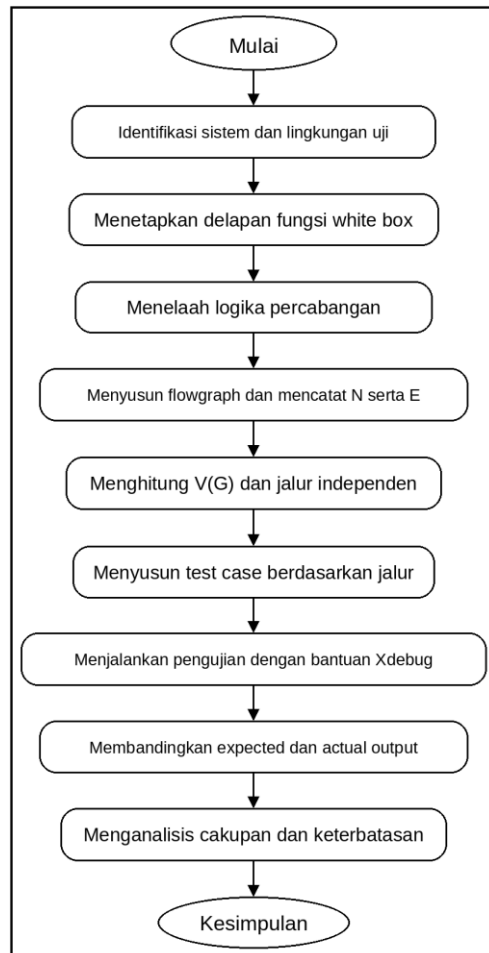
Tabel 2. Fungsi yang dianalisis

Kode	Fungsi	Fokus logika
M-01	Login dan autentikasi	Login admin/anggota, sesi, logout
M-02	Tambah/edit anggota	Validasi field, duplikasi ID, <i>insert/update</i>
M-03	Tambah/edit buku	Validasi ISBN, duplikasi, pembaruan stok
M-04	Transaksi peminjaman	Validasi anggota, stok, penyimpanan transaksi
M-05	Transaksi pengembalian	Pinjaman aktif, tanggal kembali, pembaruan stok
M-06	Kalkulasi denda	Selisih hari dan tarif Rp500 per hari
M-07	Counter pengunjung	Pembacaan dan penambahan <i>counter</i>
M-08	Validasi form	Pemeriksaan field kosong dan <i>return value</i>

Delapan fungsi pada Tabel 2 dipilih karena mempunyai percabangan yang langsung memengaruhi akses, data, stok, transaksi, denda, atau nilai *counter*. Dashboard, data karyawan, dan laporan tidak diuji dengan *basis path testing* pada penelitian ini. Pembatasan tersebut menjaga agar hasil hanya membahas fungsi yang benar-benar menjalani pengujian white box.

3.4. Tahapan Penelitian

Pengujian dimulai dengan memastikan aplikasi dan *database* dapat diakses pada lingkungan uji. Peneliti kemudian memilih delapan fungsi, membaca percabangannya, dan menyusun *flowgraph*. Jumlah *node* serta *edge* dicatat untuk menghitung $V(G)$. Jalur independen diterjemahkan menjadi skenario, lalu setiap *test case* dijalankan dengan *input* yang telah ditentukan. *Expected output* dibandingkan dengan *actual output* sebelum status PASS atau FAIL diberikan. Urutan kerja tersebut ditunjukkan pada Gambar 1.



Gambar 1. Diagram alur tahapan penelitian

3.5. Perhitungan V(G) dan Jalur Independen

Perhitungan menggunakan $V(G)=E-N+2$ karena setiap fungsi diperlakukan sebagai satu graf terhubung. Nilai $V(G)$ menentukan jumlah jalur independen yang perlu dipertimbangkan. Jalur diberi nama berdasarkan kondisi utama, seperti login valid, sandi salah, stok tersedia, stok habis, pengembalian tepat waktu, dan pengembalian terlambat.

Perhitungan pada delapan fungsi menghasilkan 32 jalur independen. Pelaksanaan pengujian didokumentasikan melalui 20 test case. Beberapa test case melewati lebih dari satu kondisi dalam satu rangkaian eksekusi, sedangkan beberapa jalur belum mempunyai skenario mandiri. Agar cakupan tidak ditafsirkan berlebihan, hubungan setiap jalur dengan test case ditampilkan secara rinci pada Tabel 4.

3.6. Pelaksanaan Test Case dan Kriteria Penilaian

Pada setiap test case, peneliti memasukkan data sesuai kondisi yang hendak dicapai, menjalankan fungsi, lalu mengamati pesan, sesi, perubahan transaksi, stok, nilai denda, atau counter. Status PASS diberikan ketika *actual output* sama dengan *expected output*. Status FAIL diberikan apabila hasil, pesan, sesi, atau perubahan database berbeda dari yang ditetapkan.

$$\text{Pass rate} = (\text{jumlah PASS} / \text{jumlah test case}) \times 100\%$$

WB-04 menggunakan satu payload SQL injection pada halaman login. Payload tersebut ditolak, tetapi hasil ini hanya berlaku untuk skenario yang dijalankan dan tidak disamakan dengan penetration testing. Pengujian pada localhost dan LAN juga tidak digunakan untuk menarik kesimpulan mengenai performa aplikasi di internet publik.

4. HASIL DAN PEMBAHASAN

4.1. Nilai V(G) dan Jalur Independen

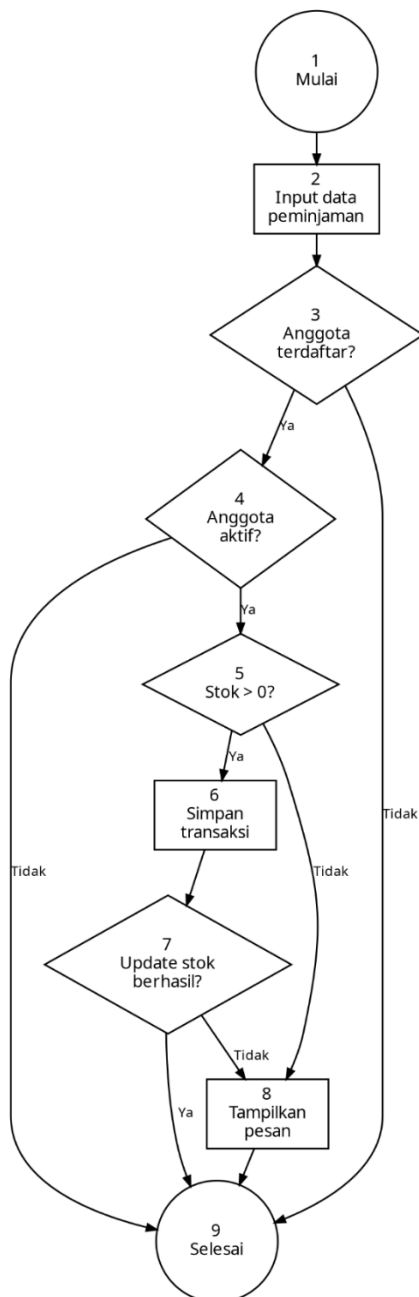
Tabel 3. Analisis cyclomatic complexity per fungsi

Kode / fungsi	N-E-V(G)	Jalur independen	Bukti TC
M-01 Login & autentikasi	8-10-4	P1 valid; P2 sandi salah; P3 sesi aktif; P4 SQL injection	WB-01-WB-04
M-02 Tambah/edit anggota	7-9-4	P1 semua valid; P2 field kosong; P3 ID duplikat; P4 edit sukses	WB-05-WB-07
M-03 Tambah/edit buku	7-9-4	P1 tambah valid; P2 ISBN kosong; P3 ISBN duplikat; P4 edit stok	WB-08-WB-10
M-04 Peminjaman	9-12-5	P1 stok ada; P2 stok habis; P3 anggota tidak ada; P4 anggota aktif; P5 sukses	WB-11-WB-13
M-05 Pengembalian	10-13-5	P1 tepat waktu; P2 terlambat; P3 anggota tidak ada; P4 pinjaman tidak ada; P5 sukses	WB-14-WB-15
M-06 Kalkulasi denda	6-8-4	P1 selisih 0; P2 >0; P3 negatif; P4 return value	WB-16-WB-17
M-07 Counter pengunjung	4-5-3	P1 counter 0; P2 >0; P3 update sukses	WB-18
M-08 Validasi form	5-6-3	P1 semua valid; P2 satu kosong; P3 banyak kosong	WB-19-WB-20

Nilai $V(G)$ berada pada rentang 3 sampai 5. Peminjaman (M-04) dan pengembalian (M-05) masing-masing bernilai 5. Pada kedua fungsi itu, peneliti harus melewati keputusan tentang anggota, status transaksi, stok, tanggal kembali, dan pembaruan data. Counter pengunjung (M-07) serta validasi form (M-08) bernilai 3 karena alurnya lebih singkat. Jumlah seluruh nilai $V(G)$ adalah 32 jalur independen.

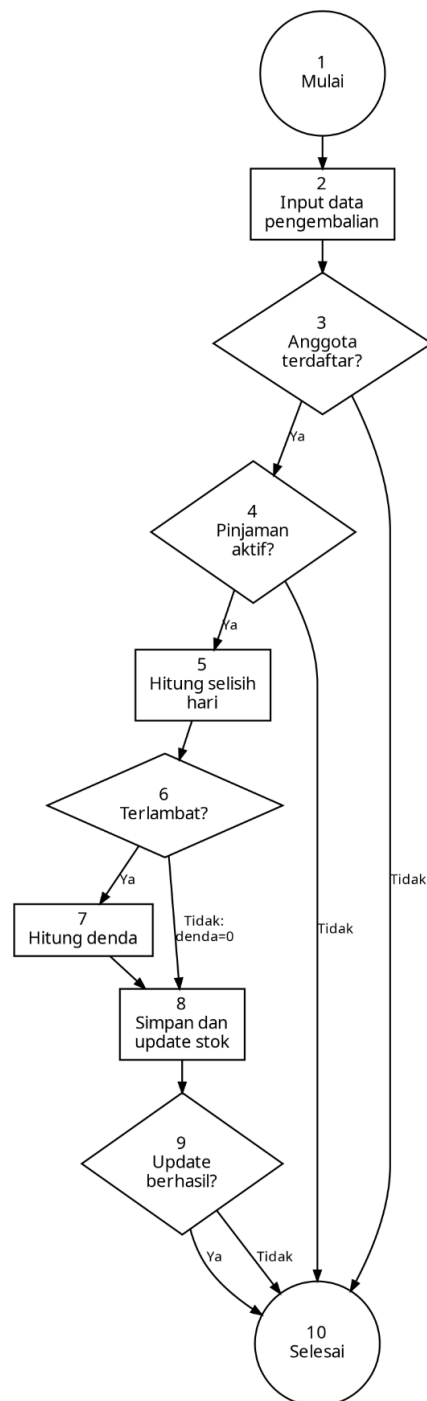
4.2. Flowgraph Transaksi Peminjaman dan Pengembalian

Flowgraph dibentuk saat peneliti menelusuri urutan keputusan pada kode. Gambar 2 memperlihatkan alur peminjaman, sedangkan Gambar 3 memperlihatkan alur pengembalian. Kedua flowgraph dipilih karena mempunyai nilai $V(G)=5$ dan menjadi fungsi dengan jalur paling banyak dalam penelitian ini.



Gambar 2. *Flowgraph* transaksi peminjaman
(N=9, E=12, V(G)=5)

Pada alur peminjaman, proses dimulai dari *input* data, pemeriksaan anggota, pemeriksaan status aktif, dan pengecekan stok. Jika seluruh syarat terpenuhi, transaksi disimpan dan stok dikurangi. Saat menjalankan WB-11, transaksi anggota A001 tersimpan dan stok berubah dari 3 menjadi 2. WB-12 mengikuti cabang stok habis, sedangkan WB-13 berhenti ketika ID anggota Z999 tidak ditemukan.



Gambar 3. *Flowgraph* transaksi pengembalian
(N=10, E=13, V(G)=5)

Pada alur pengembalian, program memeriksa anggota dan pinjaman aktif sebelum menghitung selisih hari. WB-14 melewati cabang tepat waktu sehingga denda bernilai 0. WB-15 melewati cabang terlambat tiga hari dan menghasilkan denda Rp1.500. Kedua skenario kemudian menyimpan pengembalian serta menambah stok.

4.3. Keterlacakan Jalur Independen dan Test Case

Tabel 4. Keterlacakan 32 jalur independen dan test case

Jalur	Kondisi dan test case	Hasil / cakupan
M-01/P1	Kredensial valid (WB-01)	Dashboard dan sesi aktif; PASS
M-01/P2	Password salah (WB-02)	Pesan error, sesi tidak dibuat; PASS
M-01/P3	Sesi aktif (WB-03)	<i>Redirect</i> ke dashboard; PASS
M-01/P4	<i>SQL injection</i> (WB-04)	Login ditolak; PASS
M-02/P1	Semua field valid (WB-05)	A001 tersimpan; PASS
M-02/P2	Nama kosong (WB-06)	<i>INSERT</i> tidak dijalankan; PASS
M-02/P3	ID duplikat (WB-07)	Duplikasi terdeteksi; PASS
M-02/P4	Edit anggota sukses	Belum memiliki TC mandiri
M-03/P1	Data buku valid (WB-08)	Buku tersimpan; PASS
M-03/P2	ISBN kosong	Belum memiliki TC mandiri
M-03/P3	ISBN duplikat (WB-09)	Duplikasi terdeteksi; PASS
M-03/P4	Edit stok (WB-10)	Stok menjadi 10; PASS
M-04/P1	Stok tersedia (WB-11)	Transaksi tersimpan, stok 2; PASS
M-04/P2	Stok 0 (WB-12)	<i>INSERT</i> dilewati; PASS
M-04/P3	Anggota tidak ada (WB-13)	Proses berhenti; PASS
M-04/P4	Anggota aktif (tercakup WB-11)	Dilewati pada alur berhasil
M-04/P5	Peminjaman sukses (tercakup WB-11)	Transaksi dan stok terbaru
M-05/P1	Tepat waktu (WB-14)	Denda 0, stok bertambah; PASS
M-05/P2	Terlambat (WB-15)	Denda Rp1.500, stok bertambah; PASS
M-05/P3	Anggota tidak ada	Belum memiliki TC mandiri
M-05/P4	Pinjaman aktif tidak ada	Belum memiliki TC mandiri
M-05/P5	Pengembalian sukses (WB-14/WB-15)	Terintegrasi pada dua skenario
M-06/P1	Selisih hari 0 (WB-16)	<i>Return</i> 0; PASS
M-06/P2	Selisih hari >0 (WB-17)	<i>Return</i> 2.500; PASS
M-06/P3	Selisih hari negatif	Belum memiliki TC mandiri
M-06/P4	<i>Return value</i> (WB-16/WB-17)	Terintegrasi; nilai sesuai rumus
M-07/P1	Counter awal 0	Belum memiliki TC mandiri
M-07/P2	Counter >0 (tercakup WB-18)	Nilai awal 42 terbaca
M-07/P3	<i>Update</i> sukses (WB-18)	Counter menjadi 43; PASS
M-08/P1	Semua field valid (WB-20)	<i>Return true</i> ; PASS
M-08/P2	Satu field kosong (WB-19)	<i>Return false</i> ; PASS
M-08/P3	Banyak field kosong	Belum memiliki TC mandiri

Tabel 4 menghubungkan setiap jalur dengan bukti eksekusinya. Status PASS berarti jalur mempunyai test case tersendiri. Status terintegrasi menunjukkan bahwa kondisi tersebut dilewati di dalam skenario lain yang berhasil. Jalur berstatus belum mandiri belum mempunyai test case khusus, sehingga tidak digunakan sebagai dasar untuk menyatakan full path coverage.

4.4. Hasil Pelaksanaan Test Case

Tabel 5. Hasil test case WB-01 sampai WB-05

ID / jalur	Skenario dan input	Harapan / aktual
WB-01 (PASS) M-01/P1	admin/admin123	Dashboard; sesi user_id=1 dan role=admin Dashboard tampil; sesi aktif
WB-02 (PASS) M-01/P2	admin/sandi salah	Kembali ke login; sesi kosong; pesan error Pesan error; sesi tidak dibuat

ID / jalur	Skenario dan input	Harapan / aktual
WB-03 (PASS) M-01/P3	akses login saat sesi aktif	Dialihkan otomatis ke dashboard <i>Redirect</i> terjadi tanpa form
WB-04 (PASS) M-01/P4	payload OR '1'='1' -	Login gagal; tidak ada row valid Input ditolak; login gagal
WB-05 (PASS) M-02/P1	A001, semua field terisi	Satu record anggota tersimpan Record A001 tersimpan

Tabel 5 memuat empat jalur autentikasi dan satu skenario penambahan anggota; seluruh hasil aktual sesuai harapan.

Tabel 6. Hasil test case WB-06 sampai WB-10

ID / jalur	Skenario dan input	Harapan / aktual
WB-06 (PASS) M-02/P2	nama kosong, field lain terisi	Validasi menolak; <i>INSERT</i> tidak jalan Validasi menolak

ID / jalur	Skenario dan input	Harapan / aktual
WB-07 (PASS) M-02/P3	ID A001 sudah tersedia	Pesan ID terdaftar; <i>INSERT</i> dilewati Duplikasi terdeteksi
WB-08 (PASS) M-03/P1	ISBN 978-xxx, stok 5	Satu record buku tersimpan Buku tersimpan
WB-09 (PASS) M-03/P3	ISBN 978-xxx duplikat	Pesan duplikasi; <i>INSERT</i> dilewati Duplikasi ISBN terdeteksi
WB-10 (PASS) M-03/P4	stok baru 10	Stok database menjadi 10 Stok terbaru menjadi 10

Tabel 6 menunjukkan penolakan data kosong dan duplikat serta keberhasilan penyimpanan dan pembaruan buku.

Tabel 7. Hasil test case WB-11 sampai WB-15

ID / jalur	Skenario dan input	Harapan / aktual
WB-11 (PASS) M-04/P1-P4-P5	A001, stok 3	Transaksi tersimpan; stok menjadi 2 Transaksi tersimpan; stok 2
WB-12 (PASS) M-04/P2	stok 0	Pesan stok habis; <i>INSERT</i> dilewati Cabang false berjalan
WB-13 (PASS) M-04/P3	anggota Z999	Pesan anggota tidak ditemukan; proses berhenti Early exit terpicu
WB-14 (PASS) M-05/P1-P5	kembali tepat waktu	Denda 0; pengembalian tersimpan; stok +1 Denda 0; stok bertambah
WB-15 (PASS) M-05/P2-P5	terlambat 3 hari	Denda 3 x 500 = Rp1.500; stok +1 Denda Rp1.500; stok bertambah

Tabel 7 merangkum cabang peminjaman dan pengembalian; stok dan denda berubah sesuai kondisi uji.

Tabel 8. Hasil test case WB-16 sampai WB-20

ID / jalur	Skenario dan input	Harapan / aktual
WB-16 (PASS) M-06/P1-P4	hari=0	<i>Return</i> 0 Fungsi mengembalikan 0
WB-17 (PASS) M-06/P2-P4	hari=5	<i>Return</i> 2500 Fungsi mengembalikan 2500
WB-18 (PASS) M-07/P2-P3	counter awal 42	Counter menjadi 43 dan ditampilkan Counter bertambah menjadi 43

WB-19 (PASS) M-08/P2	nama kosong	Loop berhenti; <i>return</i> false; pesan error Field kosong terdeteksi; false
WB-20 (PASS) M-08/P1	semua field terisi	Loop selesai; <i>return</i> true Semua valid; true

Tabel 8 memuat hasil fungsi denda, counter pengunjung, dan validasi form.

4.5. Rekapitulasi Pengujian

Tabel 9. Rekapitulasi pengujian white box

Test case	PASS / FAIL	Pass rate	Jalur / V(G) maks.
20	20 / 0	100%	32 / 5

Seluruh *actual output* pada Tabel 5 sampai Tabel 8 sama dengan *expected output*. Dua puluh test case berstatus PASS dan tidak ada skenario yang berstatus FAIL. *Pass rate* 100% hanya menggambarkan hasil 20 test case tersebut. Ketercakupan jalur tetap dibaca bersama Tabel 4.

4.6. Makna Nilai V(G) dan Prioritas Pengujian

Nilai V(G) membantu peneliti menentukan prioritas. Peminjaman dan pengembalian bernilai 5 karena mempunyai lebih banyak keputusan dibandingkan fungsi lain. Saat pengujian dilakukan, kedua fungsi memerlukan penyiapan kondisi anggota, stok, pinjaman aktif, dan tanggal yang berbeda. Temuan ini sejalan dengan Telaumbanua dan Juliandri [11], yang menempatkan peminjaman serta pengembalian sebagai proses utama dengan beberapa kondisi transaksi.

Counter pengunjung dan validasi form bernilai 3, tetapi keduanya tetap memerlukan pemeriksaan. Pada WB-18, counter awal 42 berubah menjadi 43 dan nilai baru tampil pada layar. WB-19 berhenti ketika nama kosong ditemukan, sedangkan WB-20 menyelesaikan seluruh perulangan dan mengembalikan nilai true. Perbedaan hasil tersebut menunjukkan bahwa jalur benar dan salah pada validasi bekerja pada *input* yang digunakan.

4.7. Autentikasi dan Pengelolaan Sesi

Empat skenario autentikasi dijalankan secara berurutan. WB-01 menghasilkan dashboard dan sesi admin. WB-02 menolak sandi salah tanpa membuat sesi. Pada WB-03, akses ke halaman login saat sesi aktif langsung dialihkan ke dashboard. WB-04 menolak *payload* OR '1'='1'-. Keempat hasil tersebut sesuai dengan *expected output*.

Penolakan satu *payload* belum cukup untuk menyatakan aplikasi aman dari *SQL injection*. ISTQB [15] membedakan pengujian struktural dari pengujian keamanan yang menggunakan tujuan, teknik, dan variasi serangan yang lebih luas. Karena itu, WB-04 diperlakukan sebagai pemeriksaan satu cabang

otentikasi, bukan sebagai pengganti vulnerability assessment.

4.8. Validasi Data Anggota dan Buku

WB-05 sampai WB-10 digunakan untuk memeriksa penambahan anggota, nama kosong, ID duplikat, penambahan buku, ISBN duplikat, dan pembaruan stok. Saat WB-05 dijalankan, anggota A001 tersimpan. WB-06 menolak form tanpa nama dan tidak menjalankan *INSERT*. WB-07 mendeteksi ID A001 yang sudah tersedia. Pada data buku, WB-08 menyimpan buku baru, WB-09 menolak ISBN duplikat, dan WB-10 mengubah stok menjadi 10.

Rifqi et al. [13] menekankan pentingnya aturan validasi pada sistem perpustakaan berbasis web. Walaupun aplikasi SMA Daarul Qur'an menggunakan PHP, tujuannya sama: mencegah data kosong atau ganda masuk ke *database*. Pramatha dan Efendi [12] juga menempatkan validasi sebagai bagian penting dalam menjaga ketepatan transaksi.

4.9. Transaksi Peminjaman dan Pengembalian

Tiga skenario peminjaman memperlihatkan cabang yang berbeda. WB-11 menyimpan transaksi ketika stok tersedia dan mengurangi stok dari 3 menjadi 2. WB-12 berhenti pada kondisi stok 0 tanpa menjalankan *INSERT*. WB-13 berhenti lebih awal karena anggota Z999 tidak ditemukan. Jalur anggota aktif dan jalur sukses ikut dilewati pada WB-11, tetapi belum memiliki *test case* mandiri.

Pengembalian diuji pada kondisi tepat waktu dan terlambat. WB-14 menghasilkan denda 0 dan menambah stok. WB-15 menghasilkan denda Rp1.500 untuk keterlambatan tiga hari dengan tarif Rp500 per hari. Fungsi kalkulasi diperiksa kembali melalui WB-16 dengan hari=0 dan WB-17 dengan hari=5; nilai yang dikembalikan masing-masing 0 dan 2.500. Hasil ini mendukung pembahasan Telaumbanua dan Juliandri [11] mengenai banyaknya kondisi pada transaksi peminjaman dan pengembalian.

4.10. Hubungan dengan Penelitian Terdahulu

Susanto et al. [6], Astuti dan Samudra [7], Satrio et al. [8], serta Aulia et al. [10] terutama membahas pembangunan dan manfaat fungsional sistem perpustakaan berbasis web. Penelitian ini melengkapi kajian tersebut dengan bukti eksekusi pada alur internal. Komara et al. [14] mengaitkan mutu sistem dengan konsistensi pengolahan data; hasil pengujian pada anggota, buku, transaksi, dan *counter* memperlihatkan konsistensi pada skenario yang dijalankan.

Basis path testing memberi dasar yang jelas untuk memilih skenario. Wallace et al. [16] menempatkan V(G) sebagai jumlah jalur independen, sedangkan McCabe [17] menjelaskan dasar matematis ukuran tersebut. Dalam pengujian ini, V(G) tidak hanya menjadi angka pelengkap. Nilainya menunjukkan bahwa peminjaman dan pengembalian

perlu mendapat porsi regresi lebih besar, sekaligus mengungkap jalur yang belum memiliki *test case* khusus.

4.11. Implikasi dan Batasan

Hasil pengujian dapat dipakai sebagai daftar regresi awal. WB-01 sampai WB-20 perlu dijalankan kembali ketika kode autentikasi, validasi, transaksi, denda, atau *counter* berubah. Fungsi dengan V(G)=5 sebaiknya mendapat prioritas karena perubahan kecil dapat memengaruhi lebih banyak jalur. Pencatatan *expected output* dan *actual output* juga perlu dipertahankan agar perubahan antarversi dapat dibandingkan.

Pengujian ini belum mengukur beban akses serentak, penggunaan memori, waktu respons, atau perilaku aplikasi di internet publik. Pemeriksaan keamanan baru mencakup satu *payload* pada autentikasi. Selain itu, tujuh jalur pada Tabel 4 belum mempunyai *test case* mandiri. Keterbatasan tersebut mencegah kesimpulan bahwa seluruh jalur telah tercakup atau aplikasi bebas dari semua defect.

5. KESIMPULAN DAN SARAN

White box testing pada delapan fungsi Sistem Informasi Perpustakaan Berbasis Web SMA Daarul Qur'an menghasilkan nilai V(G) antara 3 dan 5 serta mengidentifikasi 32 jalur independen. Kontribusi utama analisis *cyclomatic complexity* adalah memberikan dasar terukur untuk menentukan prioritas pengujian: fungsi peminjaman dan pengembalian dengan V(G)=5 memerlukan perhatian lebih besar daripada *counter* pengunjung dan validasi form yang bernilai 3. Sebanyak 20 *test case* dijalankan dan seluruh *actual output* sesuai dengan *expected output* sehingga *pass rate* mencapai 100% pada lingkungan *localhost* dan LAN sekolah. Hasil tersebut menunjukkan bahwa skenario yang mempunyai bukti eksekusi berjalan sesuai rancangan, tetapi belum membuktikan *full path coverage* karena beberapa jalur belum memiliki *test case* mandiri. Pengujian berikutnya perlu melengkapi jalur tersebut, mengulang WB-01 sampai WB-20 sebagai pengujian regresi, serta menambahkan *load testing*, *security testing*, dan otomatisasi agar perubahan kode dapat diperiksa secara konsisten.

DAFTAR PUSTAKA

- [1] X. Guo, T. Hong, Z. Chen, and M. Kadoch, "Design and implementation of web-based library management system," in *2023 International Conference on Information Processing and Network Provisioning (ICIPNP)*, 2023, pp. 207–211. doi: 10.1109/ICIPNP62754.2023.00050.
- [2] V. Y. P. Ardhana, M. Sapi'i, H. Hasbullah, and E. A. Sampetoding, "Web-based library information system using Rapid Application Development (RAD) method at Qamarul Huda University," *The IJICS (International Journal of*

- Informatics and Computer Science*), vol. 6, no. 1, pp. 43–50, 2022, doi: 10.30865/ijics.v6i1.4031.
- [3] Badan Pusat Statistik, “Tingkat kegemaran membaca masyarakat dan unsur penyusunnya menurut provinsi, 2024,” 2025. [Online]. Available: <https://www.bps.go.id/id/statistics-table/3/TIROMldrTjVjVzEwVWtkbmRUSk5abkk0T0U5Q1FUMDkjMyMwMDAw/tingkat-kegemaran-membaca-masyarakat-dan-unsur-penyusunnya-menurut-provinsi.html?year=2024>
- [4] Perpustakaan Nasional Republik Indonesia, “IPLM 2024 catat rekor tinggi, literasi nasional semakin meningkat,” Dec. 2024. [Online]. Available: <https://www.perpusnas.go.id/berita/iplm-2024-catat-rekor-tinggi-literasi-nasional-semakin-meningkat>
- [5] Badan Pusat Statistik, “Indeks pembangunan literasi masyarakat dan unsur penyusunnya menurut provinsi, 2024,” 2025. [Online]. Available: <https://www.bps.go.id/id/statistics-table/3/VEd0V05FTjBaRVJuYzA1bVkwchlhV k5KUjJGTIVUMDkjMyMwMDAw/indeks-pembangunan-literasi-masyarakat-dan-unsur-penyusunnya-menurut-provinsi.html?year=2024>
- [6] F. A. Susanto, T. Herlambang, F. Yudianto, T. D. Wulan, P. R. Ningsih, and A. I. Permatasari, “Web-based library information system at SMK PGRI Sukodadi Lamongan,” *IJRSM: International Journal of Scientific Research and Management*, vol. 10, no. 4, pp. 827–831, 2022.
- [7] P. Astuti and A. Samudra, “Sistem Informasi Perpustakaan Berbasis Web pada SDIT Insan Mandiri Kalisari Jakarta Timur dengan Metode RAD,” *INFORMATION SYSTEM FOR EDUCATORS AND PROFESSIONALS*, vol. 7, no. 2, pp. 137–148, 2023, doi: 10.51211/isbi.v7i2.2206.
- [8] M. Satrio, N. P. Handoko, and S. Ernawati, “Perancangan Sistem Informasi Perpustakaan Berbasis Web Pada MI Al-Khairiyah Jakarta Selatan,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 3, pp. 5216–5222, 2025, doi: 10.36040/jati.v9i3.13747.
- [9] H. Santoso, S. Wati, S. Sugesti, and S. Sobiyanto, “Perancangan Sistem Informasi Perpustakaan Berbasis Web,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 14, no. 1, 2026, doi: 10.23960/jitet.v14i1.8774.
- [10] A. Aulia, M. Tsani, and W. Suharso, “Design a Web-Based Library Information System Using the Waterfall Method (Case Study of SMA Muhammadiyah 2),” *Astonjadro*, vol. 13, no. 1, pp. 169–181, 2024, doi: 10.32832/astonjadro.v13i1.14562.
- [11] C. C. A. Telaumbanua and J. Juliandri, “Design Of a Web-Based Library Book Borrowing and Return Information System,” *Journal of Technology Informatics and Engineering*, vol. 3, no. 1, pp. 61–68, 2024, doi: 10.51903/jtie.v3i1.161.
- [12] Y. E. Pramatha and M. M. Efendi, “Web-Based Library Information System at Maraqitta’limat Suela Vocational School,” *International Journal of Science, Technology and Applications*, vol. 1, no. 1, pp. 29–39, 2023, doi: 10.70115/ijsta.v1i1.22.
- [13] M. N. I. Rifqi, R. Nindiyasari, and A. C. Murti, “Rancang Bangun Sistem Informasi Perpustakaan Berbasis Website Menggunakan Framework Laravel,” *bit-Tech*, vol. 7, no. 3, pp. 817–825, 2025, doi: 10.32877/bt.v7i3.2200.
- [14] R. G. Komara, M. R. Pahlevi, and R. F. R. Hernawan, “Web-Based Library Information System Design,” *International Journal of Research and Applied Technology (INJURATECH)*, vol. 3, no. 1, pp. 161–170, 2023, doi: 10.52661/jict.v6i2.349.
- [15] International Software Testing Qualifications Board, “Certified Tester Foundation Level Syllabus v4.0.1,” Sep. 2024. [Online]. Available: https://www.istqb.org/wp-content/uploads/2024/11/ISTQB_CTFL_Syllabus_v4.0.1.pdf.
- [16] D. R. Wallace, A. H. Watson, and T. J. McCabe, “Structured Testing: A Testing Methodology Using the Cyclomatic Complexity Metric,” NIST Special Publication 500-235, 1996, doi: 10.6028/NIST.SP.500-235.
- [17] T. J. McCabe, “A Complexity Measure,” *IEEE Transactions on Software Engineering*, vol. SE-2, no. 4, pp. 308–320, 1976, doi: 10.1109/TSE.1976.233837