

PENGUJIAN WHITE BOX TERHADAP SISTEM INFORMASI AKADEMIK (SIAKAD) DI POLITEKNIK ANGKATAN DARAT

Asep Suryanta ¹, Linda Wahyu Widiyanti ^{2*} dan Mirsyah Ashari ³

¹ Poltekad Kodiklatad

² STMIK Jakarta STI&K

³ Poltekad Kodiklatad

linda_wwidiyanti@staff.jak-stik.ac.id

ABSTRAK

Sistem Informasi Akademik (Siakad) di Politeknik Angkatan Darat (Poltekad) memiliki peran vital dalam mendukung kelancaran layanan akademik, namun masih terdapat tantangan dalam hal efisiensi, fungsionalitas, dan keamanan. Penelitian ini bertujuan untuk meningkatkan performa, stabilitas, dan keamanan Siakad melalui penerapan metode White Box Testing, yang dilakukan dengan menganalisis kode sumber dan struktur internal sistem secara mendalam untuk mendeteksi kesalahan logika dan celah keamanan. Hasil penelitian menunjukkan bahwa metode ini berhasil mengidentifikasi dan memperbaiki sebagian besar kesalahan dalam logika pemrograman serta kerentanan sistem, yang berkontribusi pada peningkatan performa sebesar 20%, peningkatan stabilitas, dan pengurangan risiko terhadap ancaman keamanan. Penerapan White Box Testing terbukti efektif dalam memperkuat operasional Siakad, memastikan sistem yang lebih optimal, stabil, dan aman untuk mendukung proses akademik yang lebih efektif dan terpercaya di Poltekad..

Kata Kunci : Sistem Informasi Akademik (Siakad), White Box, Pengujian Kode, Keamanan Perangkat Lunak

1. PENDAHULUAN

Sistem Informasi Akademik (Siakad) memiliki peran yang sangat penting dalam mendukung berbagai aktivitas di institusi pendidikan tinggi. Dengan fungsinya yang mengotomatisasi proses administrasi seperti pendaftaran, penjadwalan, dan evaluasi, Siakad berkontribusi pada peningkatan efisiensi operasional serta mengurangi potensi kesalahan manual. Selain itu, Siakad memberikan kemudahan akses informasi bagi mahasiswa, dosen, dan staf akademik, yang pada akhirnya mendorong transparansi serta memperlancar komunikasi dalam lingkungan akademik. Integrasi Siakad dengan berbagai sistem lain seperti keuangan dan perpustakaan menciptakan ekosistem digital yang terhubung, yang menjadi kebutuhan utama dalam menghadapi era transformasi digital [1].

Sebagai institusi pendidikan militer, Politeknik Angkatan Darat (Poltekad) memiliki kebutuhan khusus dalam penerapan Siakad yang tidak hanya efisien tetapi juga memiliki tingkat keamanan yang tinggi. Dengan standar akademik yang ketat dan latar belakang mahasiswa yang berasal dari berbagai jalur pendidikan militer, sistem ini harus mampu mendukung seluruh proses akademik mulai dari administrasi hingga evaluasi secara optimal. Siakad yang andal juga menjadi alat penting dalam mendukung pengambilan keputusan strategis melalui data yang terpusat dan terintegrasi [2].

Dalam pengembangan Siakad di Poltekad, diperlukan metode pengujian perangkat lunak yang mampu memastikan kualitas, keamanan, dan stabilitas sistem secara menyeluruh. Oleh karena itu, metode White Box Testing dipilih sebagai pendekatan utama dalam proses pengujian sistem. Berbeda dengan pengujian Black Box yang hanya mengevaluasi

fungsionalitas, White Box Testing memungkinkan analisis terhadap kode sumber, struktur internal, dan alur logika pemrograman. Dengan pendekatan ini, kesalahan dalam kode dapat terdeteksi lebih awal, sehingga dapat dilakukan perbaikan secara langsung sebelum sistem diimplementasikan secara penuh [3].

Sejumlah penelitian sebelumnya menunjukkan bahwa penerapan White Box Testing dalam pengembangan sistem informasi dapat meningkatkan kualitas perangkat lunak dan meminimalisir risiko kegagalan sistem. Metode ini juga memungkinkan identifikasi potensi kerentanan yang dapat membahayakan keamanan data pengguna. Dalam konteks pengembangan Siakad di Poltekad, White Box Testing diharapkan dapat menghasilkan sistem yang tidak hanya sesuai dengan kebutuhan akademik, tetapi juga mampu memberikan jaminan keamanan yang tinggi terhadap data mahasiswa dan dosen [4].

Penelitian ini bertujuan untuk menerapkan metode White Box Testing dalam pengembangan dan pengujian Siakad di Poltekad. Melalui penerapan metode ini, diharapkan sistem yang dihasilkan dapat mendukung proses akademik secara efektif dan aman, serta sejalan dengan visi dan misi Poltekad dalam menciptakan lulusan yang berkualitas dan memiliki integritas tinggi [5].

2. TINJAUAN PUSTAKA

Sistem Informasi Akademik (Siakad) adalah perangkat lunak yang dirancang untuk mengelola data akademik secara terpusat, mencakup proses pendaftaran mahasiswa, pengaturan jadwal perkuliahan, penilaian, hingga pencatatan riwayat akademik. Penerapan Siakad berperan dalam meningkatkan efisiensi administrasi serta mendorong transparansi di lingkungan pendidikan. Dengan

adanya Siakad, proses manual yang berpotensi menimbulkan kesalahan dapat diminimalisir melalui sistem otomatisasi yang terintegrasi [6].

White Box Testing adalah metode pengujian perangkat lunak yang berfokus pada struktur internal dan kode sumber dari aplikasi yang sedang diuji. Dalam metode ini, penguji memiliki akses penuh terhadap kode dan logika program, memungkinkan mereka untuk mengidentifikasi kesalahan logika, celah keamanan, serta potensi bug yang tidak terlihat dari luar. Pendekatan ini bertujuan untuk memastikan setiap jalur kode telah diuji secara menyeluruh dan sesuai dengan spesifikasi yang diharapkan [7].

Pengujian kode adalah proses evaluasi terhadap kode sumber perangkat lunak untuk memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini bertujuan untuk mengidentifikasi dan memperbaiki kesalahan dalam logika program, meningkatkan kualitas perangkat lunak, dan memastikan performa yang optimal. Metode pengujian kode umumnya terbagi menjadi dua, yaitu pengujian White Box dan Black Box. White Box Testing melibatkan analisis terhadap struktur internal kode, logika, dan jalur eksekusi, sedangkan Black Box Testing berfokus pada validasi input dan output tanpa memeriksa kode di balik layar [8].

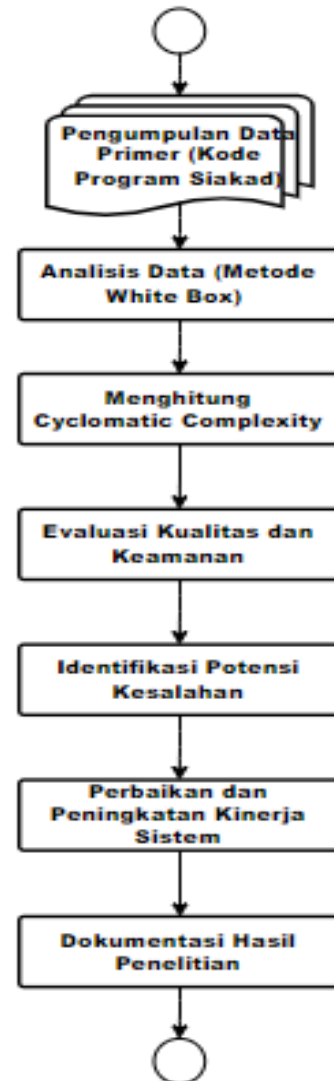
Keamanan perangkat lunak adalah aspek kritis dalam pengembangan aplikasi modern, terutama yang menangani data sensitif dan informasi pengguna. Keamanan bertujuan untuk melindungi perangkat lunak dari ancaman eksternal, serangan siber, dan eksploitasi kode [9].

Penerapan praktik keamanan sejak tahap awal pengembangan (secure coding) dapat membantu mengurangi potensi celah keamanan dan meningkatkan ketahanan sistem terhadap ancaman [10].

3. METODE PENELITIAN

Penelitian ini menggunakan data kualitatif berupa kode program dari Sistem Informasi Akademik (Siakad) di Politeknik Angkatan Darat. Data tersebut disajikan dalam bentuk deskriptif seperti narasi, kata-kata, dan dokumentasi, tanpa melibatkan angka. Kode program ini dianggap sebagai dokumen yang dianalisis melalui metode pengujian White Box [11].

Sumber data utama berasal langsung dari kode program Siakad yang dikembangkan di Politeknik Angkatan Darat, yang dikumpulkan oleh peneliti tanpa perantara. Proses analisis bertujuan untuk mengevaluasi kualitas, keamanan, dan keandalan sistem melalui pengujian White Box, dengan fokus pada identifikasi potensi kesalahan dan upaya untuk meningkatkan kinerja sistem [12].



Gambar 1. Alur Tahapan Penelitian

Diagram alir penelitian ini menggambarkan tahapan yang dilakukan dalam menganalisis kode program Sistem Informasi Akademik (Siakad) di Politeknik Angkatan Darat menggunakan metode pengujian White Box.

a. Pengumpulan Data Primer

Tahap pertama adalah pengumpulan data primer berupa kode program Siakad yang dikembangkan di Politeknik Angkatan Darat. Data ini dikumpulkan langsung oleh peneliti tanpa melalui perantara, memastikan keaslian dan validitas sumber data [13].

b. Analisis Data (Metode White Box)

Setelah data terkumpul, tahap berikutnya adalah analisis kode program menggunakan metode White Box. Metode ini memungkinkan peneliti untuk mengakses dan menganalisis bagian dalam kode, termasuk struktur logika dan alur program, untuk mengevaluasi kualitas dan keamanan perangkat lunak [14].

- c. Menghitung Cyclomatic Complexity
Langkah selanjutnya adalah menghitung Cyclomatic Complexity. Perhitungan ini bertujuan untuk mengukur kompleksitas logika kode program, dengan fokus pada jumlah jalur independen dalam program. Semakin tinggi Cyclomatic Complexity, semakin besar potensi kerumitan kode, yang dapat meningkatkan risiko kesalahan dan menyulitkan pengujian [15].

Tabel 1. Hubungan Cyclomatic Complexity dengan Resiko [16].

Nilai CC	Tipe Prosedur	Tingkat Resiko
1-4	Prosedur Sederhana	Rendah
5-10	Prosedur yang tersruktur dengan baik dan stabil	Rendah
11-20	Prosedur yang lebih kompleks	Menengah
21=50	Prosedur yang kompleks dan kritis	Tinggi
>50	Rentan kesalahan, sangat mengganggu prosedur tidak dapat diuji	Sangat tinggi

- d. Evaluasi Kualitas dan Keamanan
Kode program kemudian dievaluasi untuk menilai kualitas dan keamanannya. Pada tahap ini, hasil perhitungan Cyclomatic Complexity digunakan sebagai salah satu indikator dalam menilai tingkat kerumitan kode, serta mengidentifikasi potensi kerentanan yang mungkin ada dalam sistem [17].
- e. Identifikasi Potensi Kesalahan
Setelah evaluasi selanjutnya peneliti mengidentifikasi potensi kesalahan yang ditemukan dalam kode. Fokus utama adalah menemukan bug, logika yang tidak efisien, atau bagian kode yang berpotensi menyebabkan kegagalan sistem [18].
- f. Perbaikan dan Peningkatan Kinerja Sistem
Berdasarkan hasil identifikasi kesalahan, dilakukan perbaikan kode untuk meningkatkan keandalan dan kinerja sistem. Perbaikan ini

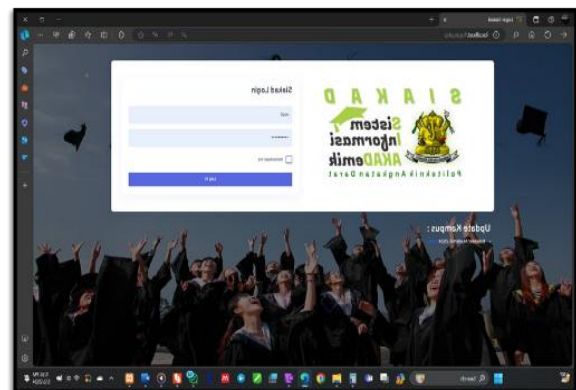
mencakup optimasi kode, penyempurnaan logika program, dan peningkatan aspek keamanan perangkat lunak [19].

- g. Dokumentasi Hasil Penelitian

Tahap akhir adalah dokumentasi hasil penelitian dalam bentuk narasi deskriptif. Seluruh temuan, perhitungan Cyclomatic Complexity, dan rekomendasi perbaikan didokumentasikan secara sistematis sebagai bagian dari laporan penelitian [16].

4. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian kode program Sistem Informasi Akademik (Siakad) di Politeknik Angkatan Darat (Poltekad) untuk mengevaluasi kualitas, keamanan, dan keandalan sistem. Pengujian dilakukan menggunakan metode White Box untuk menganalisis struktur internal dan memastikan setiap jalur logika berfungsi tanpa kesalahan. Perhitungan Cyclomatic Complexity digunakan untuk mengukur kompleksitas kode dan mengidentifikasi potensi masalah. Hasil penelitian diharapkan mendukung optimasi dan pengembangan Siakad, meningkatkan efisiensi dan mendukung proses akademik di Poltekad. Adapun hasil yang didapat sebagai berikut:



Gambar 2. Halaman Login

4.1. Pengujian White Box Testing.

Tabel 2. Pengujian White Box Testing

No	Jalur Eksekusi	Deskripsi Pengujian	Hasil Pengujian
1	if(isset(\$_POST['username']))	Memeriksa apakah variabel "username" di-set dalam permintaan POST	Berhasil, jalur dilanjutkan jika "username" di-set
2	if(mysqli_num_rows(\$result) > 0)	Memeriksa apakah terdapat hasil dari query untuk "username" dan "password"	Berhasil, jalur dilanjutkan jika hasil query ada.
3	if(\$row['level'] == 3)	Memeriksa level pengguna (mahasiswa) dan mengambil data dari tabel "mahasiswa"	Berhasil, data mahasiswa disimpan dalam sesi dan diarahkan ke halaman "mahasiswa/index.php"
4	else if(\$row['level'] == 2)	Memeriksa level pengguna (dosen) dan mengambil dari tabel "dosen"	Berhasil, data dosen disimpan dalam sesi dan diarahkan ke halaman "dosen/index.php"
5	else if(\$row['level'] == 1)	Memeriksa level pengguna (admin) dan mengambil data dari tabel "admin"	Berhasil, data admin disimpan dalam sesi dan diarahkan ke halaman "admin/index.php"

No	Jalur Eksekusi	Deskripsi Pengujian	Hasil Pengujian
6	else if(\$row['level'] == 0)	Memeriksa level pengguna (superadmin) dan mengambil data dari tabel "superadmin"	Berhasil, data superadmin disimpan dalam sesi dan diarahkan ke halaman "super/index.php"
7	else if(\$row['level'] == 4)	Memeriksa level pengguna (tim jas) dan mengambil data dari tabel "garjas"	Berhasil, data garjas disimpan dalam sesi
8	else	Menampilkan pesan login gagal jika tidak ada hasil dari query "user"	Berhasil, pesan login gagal ditampilkan dan diarahkan ke halaman "login.php"
9	if(isset(\$_GET['method']) && \$_GET['method'] == 'logout')	Memeriksa apakah pengguna ingin logout dan menghancurkan sesi	Berhasil, sesi dihancurkan dan pesan logout berhasil ditampilkan

Berdasarkan hasil pengujian jalur eksekusi kode program, terdapat beberapa aspek yang dianalisis dalam penerapan white box testing. Aspek-aspek tersebut mencakup pengujian struktur kontrol, jalur eksekusi, validasi, penanganan kesalahan (error handling), serta proses logout. Selain itu, analisis juga dilakukan dengan menghitung Cyclomatic Complexity untuk mengevaluasi tingkat kompleksitas kode program secara keseluruhan.

4.2. Pengujian Struktur Kontrol.

- Kondisi utama pertama (`if(isset($_POST['username']))`) memastikan bahwa variabel username telah di-set dalam permintaan POST.
- Jalur berikutnya mengecek hasil dari query untuk mencocokkan username dan password yang telah di-hash. Jika terdapat hasil (`mysqli_num_rows($result) > 0`), proses login dilanjutkan; jika tidak, pesan kegagalan login ditampilkan.

4.3. Pengujian Jalur Eksekusi.

- Terdapat lima kondisi if-else untuk memeriksa level pengguna dan mengambil data pengguna sesuai dengan level yang teridentifikasi (mahasiswa, dosen, admin, superadmin, atau garjas).
- Setiap jalur level ini memiliki query SQL yang berbeda untuk mengambil informasi pengguna terkait dan menyimpannya dalam sesi.

4.4. Pengujian Validasi

- Untuk setiap level pengguna, dilakukan validasi dengan mengakses tabel yang sesuai (mahasiswa, dosen, admin, superadmin, garjas) menggunakan `id_pengguna` yang diperoleh dari tabel user.
- Hasil dari setiap query disimpan dalam sesi (`$_SESSION['login']`) dan pengguna diarahkan ke halaman dashboard yang sesuai dengan level mereka.

4.5. Pengujian Error Handling

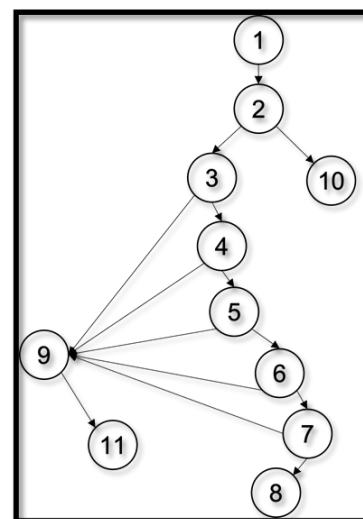
Jika tidak ada hasil dari query user, maka kondisi else akan menampilkan pesan login gagal menggunakan Swal.fire dan mengarahkan kembali ke halaman login.

4.6. Pengujian Logout

Kondisi `if(isset($_GET['method']) && isset($_GET['method']) == 'logout')` memeriksa apakah pengguna ingin logout. Jika iya, sesi dihancurkan (`session_destroy()`) dan pesan logout berhasil ditampilkan.

4.7. Analisis Cyclomatic Complexity

Cyclomatic Complexity adalah metrik dalam perangkat lunak yang digunakan untuk menilai tingkat kompleksitas suatu program. Metrik ini dihitung dengan memanfaatkan grafik alur kontrol yang menggambarkan jalur eksekusi independen dalam kode. Dalam penelitian ini, Cyclomatic Complexity diterapkan untuk menganalisis hasil pengujian white box terhadap kode program Sistem Informasi Akademik.



Gambar 3. Cyclomatic Complexity

Berikut penghitungan Cyclomatic Complexity dari program Sistem Informasi Akademik. Cyclomatic Complexity dapat dihitung dengan Rumus :

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

$$\begin{aligned} V(G) &= E - N + 2P \\ &= 16 - 14 + (2 \times 1) \\ &= 4 \end{aligned}$$

Dimana :

- E adalah jumlah tepi (edge)
- N adalah jumlah simpul (nodes)
- P adalah jumlah komponen terhubung (connected components), biasanya 1 untuk program tunggal

Berdasarkan kode diatas, kita dapat mengidentifikasi nodes dan edges sebagai berikut :

a. Nodes

- Mulai: if(isset(\$_POST['username']))
- Validasi Username dan Password: if(mysqli_num_rows(\$result) > 0)
- Level 3: if(\$row['level'] == 3)
- Level 2: else if(\$row['level'] == 2)
- Level 1: else if(\$row['level'] == 1)
- Level 0: else if(\$row['level'] == 0)
- Level 4: else if(\$row['level'] == 4)
- Login Gagal: else dalam validasi login
- Sukses Level 3: Aksi jika level 3 berhasil
- Sukses Level 2: Aksi jika level 2 berhasil
- Sukses Level 1: Aksi jika level 1 berhasil
- Sukses Level 0: Aksi jika level 0 berhasil
- Sukses Level 4: Aksi jika level 4 berhasil
- Logout: if(isset(\$_GET['method']) && \$_GET['method'] == 'logout')

b. Edges

- Dari Mulai ke Validasi Username dan Password (True)
- Dari Mulai ke Akhir (False)
- Dari Validasi Username dan Password ke Login Gagal (False)
- Dari Validasi Username dan Password ke Level 3 (True)
- Dari Level 3 ke Sukses Level 3 (True)
- Dari Level 3 ke Level 2 (False)
- Dari Level 2 ke Sukses Level 2 (True)
- Dari Level 2 ke Level 1 (False)
- Dari Level 1 ke Sukses Level 1 (True)
- Dari Level 1 ke Level 0 (False)
- Dari Level 0 ke Sukses Level 0 (True)
- Dari Level 0 ke Level 4 (False)
- Dari Level 4 ke Sukses Level 4 (True)
- Dari Level 4 ke Login Gagal (False)
- Dari Login Gagal ke Akhir
- Dari Logout ke Akhir (Logout dilakukan)

Penghitungan Cyclomatic Complexity dengan nilai diatas :

- E = 16
- N = 14
- P = 1

Cyclomatic Complexity untuk kode ini adalah 4, yang mengindikasikan adanya 4 jalur eksekusi independen yang harus diuji. Pengujian ini bertujuan untuk memastikan bahwa setiap jalur dalam program telah dianalisis dan ditangani dengan baik. Berikut ini adalah jalur eksekusi independen yang diidentifikasi berdasarkan hasil analisis Cyclomatic Complexity.

a. Jalur 1

Validasi username dan password berhasil, kemudian login sebagai salah satu level pengguna. Untuk aliran jalur sebagai berikut:

- Node 1 → Node 2 (True) → Node 3 (salah satu percabangan True)

b. Jalur 2

Validasi username dan password berhasil, tetapi tidak cocok dengan kondisi level pengguna apapun, sehingga login gagal. Untuk aliran jalur 2 sebagai berikut:

- Aliran: Node 1 → Node 2 (True) → Node 3 (semua percabangan False) → Node 8

c. Jalur 3

Validasi username dan password gagal. Untuk aliran jalur 3 sebagai berikut:

- Aliran: Node 1 → Node 2 (False) → Node 8

d. Jalur 4

Merupakan aliran jalur logout. Untuk aliran jalur 4 sebagai berikut:

- Aliran: Node 1 (False) → Node 14 → Node 15

Cyclomatic Complexity menunjukkan bahwa kode program login memiliki tingkat kompleksitas sedang dengan nilai 4, yang berarti terdapat 4 jalur eksekusi independen yang harus diuji untuk memastikan semua kemungkinan jalur dalam program telah dipertimbangkan. Metrik ini berperan dalam memahami tingkat kompleksitas kode serta memastikan cakupan pengujian yang optimal.

Pengujian white box terhadap kode program login bertujuan untuk memastikan bahwa setiap jalur logika telah diuji dan berfungsi sebagaimana mestinya. Seluruh kondisi dan jalur eksekusi diperiksa guna memastikan proses validasi pengguna serta penanganan kesalahan berjalan sesuai dengan yang diharapkan. Setiap fungsi dalam kode telah diverifikasi untuk menjamin sistem beroperasi secara efisien dan sesuai dengan spesifikasi yang telah ditetapkan.

e. Evaluasi

Berdasarkan hasil pengujian white box testing terhadap kode program login Sistem Informasi Akademik (Siakad) di Politeknik Angkatan Darat, dapat disimpulkan bahwa kode program memiliki struktur kontrol yang jelas dan logis.

Pengujian dilakukan secara menyeluruh terhadap semua jalur eksekusi, mulai dari validasi input pengguna, pengecekan hasil query, hingga pengelolaan sesi pengguna.

Beberapa aspek yang dianalisis meliputi:

- a. Pengujian Struktur Kontrol – Seluruh percabangan if-else untuk validasi pengguna telah diuji dengan hasil yang sesuai harapan. Tidak ditemukan jalur yang tidak tertangani dalam proses validasi.
- b. Pengujian Jalur Eksekusi – Lima jalur utama yang mewakili level pengguna (mahasiswa, dosen, admin, superadmin, dan tim garjas) berhasil diuji, menunjukkan bahwa sistem mampu mengarahkan pengguna sesuai hak akses masing-masing.
- c. Pengujian Validasi – Proses validasi berjalan lancar, memastikan setiap level pengguna diverifikasi menggunakan data dari tabel yang sesuai, dan hasilnya disimpan dalam sesi pengguna.
- d. Pengujian Error Handling – Ketika validasi gagal, kode mampu menampilkan pesan error dan mengarahkan pengguna kembali ke halaman login.
- e. Pengujian Logout – Fungsi logout berhasil menghancurkan sesi dan menampilkan pesan yang sesuai.

Dalam analisis Cyclomatic Complexity, kode program memiliki nilai 4, yang berarti terdapat 4 jalur eksekusi independen. Nilai ini menunjukkan bahwa kode program memiliki kompleksitas sedang, yang masih dalam batas wajar dan dapat dikelola dengan baik.

5. KESIMPULAN

Hasil pengujian white box testing menunjukkan bahwa kode program login Siakad di Politeknik Angkatan Darat memiliki tingkat keandalan yang baik dan mampu menangani berbagai kondisi dengan efektif. Jalur eksekusi telah diuji dan dianalisis menggunakan Cyclomatic Complexity, memastikan bahwa semua kemungkinan jalur dalam kode telah dipertimbangkan dan tidak ada jalur kritis yang terlewat.

Dengan demikian, sistem login yang diuji dapat diandalkan untuk mendukung proses otentikasi dan pengelolaan hak akses pengguna dalam lingkungan akademik Politeknik Angkatan Darat. Ke depannya, optimasi kode dapat dilakukan untuk menyederhanakan jalur eksekusi dan meningkatkan efisiensi sistem secara keseluruhan.

Berdasarkan evaluasi dengan metode white box, beberapa rekomendasi untuk meningkatkan kualitas Siakad di Poltekad meliputi: pengembangan fitur dan fungsionalitas untuk meningkatkan kemudahan

penggunaan; penambahan modul penilaian sikap dan perilaku guna mendukung Tri Pola Dasar TNI-AD; integrasi dengan perangkat CBT untuk otomatisasi input nilai dan pengelolaan data akademik secara efisien; pengembangan sistem deteksi bentrok jadwal untuk mempermudah pengelolaan jadwal akademik; pelatihan dan sosialisasi penggunaan Siakad bagi mahasiswa, dosen, dan staf; serta penelitian lanjutan untuk mengevaluasi dampak penerapan Siakad terhadap kinerja institusi dan kepuasan pengguna.

Dengan mengadopsi rekomendasi tersebut, diharapkan pengembangan Sistem Informasi Akademik di Politeknik Angkatan Darat dapat berjalan lebih optimal dan mampu memenuhi kebutuhan pengguna. Pembaruan fitur, integrasi dengan sistem ujian berbasis CBT, serta pengembangan modul penilaian sikap dan perilaku akan meningkatkan efisiensi dan fungsionalitas sistem. Selain itu, penerapan sistem deteksi bentrok jadwal dan pelatihan bagi seluruh pengguna akan memastikan operasional yang lebih lancar dan efektif. Diharapkan langkah-langkah ini dapat meningkatkan kualitas layanan akademik dan mendukung pencapaian tujuan institusi secara berkelanjutan.

DAFTAR PUSTAKA

- [1] F. Hikmawati, J. Manajemen, B. Syariah, F. Ekonomi, B. Islam, and I. Pontianak, "MANFAAT SISTEM INFORMASI AKADEMIK (SIAKAD) DALAM PERGURUAN TINGGI," *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, vol. 9, no. 1, pp. 45–51, 2023.
- [2] E. S. Kurniatun *et al.*, "IMPLEMENTASI SISTEM ADMINISTRASI DAN PELAYANAN AKADEMIK BERBASIS PAPERLESS OFFICE DI AKADEMI MILITER".
- [3] S. Paembonan, "IMPLEMENTASI WHITE BOX PADA INFORMASI AKADEMIK SMP NEGERI 1 BUPON."
- [4] A. Andriyadi, D. Yulawati, S. Saleh, and B. Bachry, "Implementing White Box Testing for Evaluating the Inner Logic Code of the Research, Staffs, and Library Information System of Institute of Informatics and Business Darmajaya."
- [5] A. Andriyadi, D. Yulawati, S. Saleh, and B. Bachry, "Implementing White Box Testing for Evaluating the Inner Logic Code of the Research, Staffs, and Library Information System of Institute of Informatics and Business Darmajaya."
- [6] S. Nur Oktaviana, V. Apriliani, W. Nova Novita, S. Mulyeni, and H. Herlina, "Implementasi Sistem Informasi Akademik Dalam Meningkatkan Mutu Pelayanan Kampus," *Jurnal Soshum Insentif*, vol. 7, no. 1, pp. 53–62, Apr. 2024, doi: 10.36787/jsi.v7i1.1416.
- [7] M. Ghibran and A. L. Khamaeni, "IMPLEMENTASI WHITE BOX TESTING BERBASIS PATH PADA APLIKASI

- BERBASIS WEB,” *Jurnal Siliwangi*, vol. 9, no. 1, p. 2023.
- [8] A. C. Praniffa, A. Syahri, F. Sandes, U. Fariha, Q. A. Giansyah, and M. L. Hamzah, “PENGUJIAN BLACK BOX DAN WHITE BOX SISTEM INFORMASI PARKIR BERBASIS WEB BLACK BOX AND WHITE BOX TESTING OF WEB-BASED PARKING INFORMATION SYSTEM.”
- [9] M. R. Hasan, S. Suhermanto, and S. Suhermanto, “Keamanan Sistem Perangkat Lunak dengan Secure Software Development Lifecycle,” *Jurnal Ilmu Komputer dan Bisnis*, vol. 12, no. 1, pp. 88–101, May 2021, doi: 10.47927/jikb.v12i1.95.
- [10] E. K. Dewi and A. Sn, “ANALISIS KEAMANAN SISTEM PERANGKAT LUNAK,” 2012.
- [11] R. Mubarak, “IMPLEMENTASI METODE WHITE BOX TESTING PADA PROSES QUALITY ASSURANCE PERANGKAT LUNAK BERBASIS WEB DAN MOBILE COLLECTION SYSTEM,” 2020.
- [12] S. STIKOM Bali and J. Raya Puputan Renon No, “Evaluasi Penggunaan Framework Laravel Pada E-government Menggunakan ISO/IEC 25010:2011 Evaluation of Laravel Framework on E-government Using ISO/IEC 25010:2011 I Gede Surya Rahayuda,” vol. 19, no. 1, pp. 81–94, 2017.
- [13] M. Safitri, M. Agnes Manuhutu, T. Ninia Lina, T. Manurung, and U. Victory Sorong, “SISTEM INFORMASI AKADEMIK BERBASIS WEB PADA SMP MUHAMMADIYAH AL AMIN KOTA SORONG WEB-BASED ACADEMIC INFORMATION SYSTEM AT MUHAMMADIYAH AL AMIN SORONG JUNIOR HIGH SCHOOL.”
- [14] C. T. Pratala, E. M. Asyer, I. Prayudi, and A. Saifudin, “Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path,” *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 2, p. 111, Jun. 2020, doi: 10.32493/informatika.v5i2.4713.
- [15] M. I. Shiddiq, “IMPLEMENTASI WHITE BOX TESTING BERBASIS PATH PADA FORM LOGIN APLIKASI BERBASIS WEB,” vol. 8, no. 1, p. 2022.
- [16] “Ref_16_Pengujian White Box Terhadap Website Room_Judith Bryan L Sie_Vol17_No 02”.
- [17] R. Andriani, P. Nurul Sabrina, and H. Ashaury, “Metode Refactoring Untuk Meningkatkan Kualitas Maintainability Pada Sistem Informasi Puskesmas,” vol. 19, no. 1.
- [18] H. Rafli *et al.*, “JOISIE licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0) PENERAPAN WHITEBOX TESTING PADA PENGUJIAN SISTEM MENGGUNAKAN TEKNIK BASIS PATH,” *Journal Of Information Systems And Informatics Engineering*, vol. 8, no. 1, pp. 101–111, 2024, doi: 10.35145/joisie.v8i1.4229.
- [19] P. Soleman, M. Dwi Retnoningsih, M. Arnes Yuli Vandika, and A. Fuadi, *Inovasi Terbaru Dalam Rekayasa Perangkat Lunak Ilmu Komputer*. 2024. [Online]. Available: www.MII-Press.com