



Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode *Blackbox Testing* Bagi Pemula

Bintang Mujahidin Haqqoni^{1*}, Irgi Winarno², Muhammad Naufal Musthofa³, Muzammil Sakdi⁴, Aries Saifudin⁵

¹Ilmu komputer, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

E-mail: ^{1*}Bintangmujahidin@gmail.com, ²Irgiwinarno680@gmail.com, ³mnaufalmusthofa16@gmail.com,

⁴mieng12@gmail.com, ⁵aries.saifudin@unpam.ac.id

(* : coresponding author)

Abstrak – Sistem informasi yang dirancang sebelum digunakan secara formal harus melewati pengujian terhadap perangkat lunaknya terlebih dahulu. Pada penelitian ini, metode Black Box Testing dipilih karena tidak memerlukan pengetahuan tentang Bahasa pemrograman tertentu. Teknik pengujian yang digunakan adalah Equivalen Partitions Testing, yang bertujuan untuk menguji apakah ada kesalahan pada fungsi sistem, interface, struktur data atau akses data, performansi, dan inisialisasi. Obyek yang diuji adalah Sistem Informasi Perpustakaan melalui web, dengan jumlah form yang diuji sebanyak 3 buah, yaitu Form Inisialisasi, Form Peminjaman, dan Form Pengembalian. Hasil penelitian menunjukkan bahwa pengujian perangkat lunak menggunakan metode Black Box Testing dengan Teknik Equivalence Partitions pada sistem informasi perpustakaan masih terdapat form yang hasil outputnya belum sesuai dengan yang diharapkan dalam perancangan karena terjadi error. Dari hasil perhitungan validitas sistem diperoleh nilai 75% pada sistem informasi perpustakaan buku, menunjukkan bahwa sistem masih memiliki beberapa kesalahan yang perlu diperbaiki. Kemungkinan error yang terjadi dikarenakan adanya kesalahan pada baris kode (syntax) pada sistem informasi perpustakaan buku. Oleh karena itu, perlu dilakukan update pada syntax untuk memperbaiki bug tersebut dan meningkatkan kualitas sistem.

Kata Kunci: Pengujian Fungsional; Perangkat Lunak; Sistem Informasi Perpustakaan; Metode *Black Box Testing*; *Equivalence Partitions*

Abstract – Information systems that are designed before being formally used must first pass testing of the software. In this research, the Black Box Testing method was chosen because it does not require knowledge of a particular programming language. The testing technique used is Equivalent Partitions Testing, which aims to test whether there are errors in system functions, interfaces, data structures or data access, performance and initialization. The object tested was the Library Information System via the web, with 3 forms tested, namely the Initialization Form, Loan Form and Return Form. The research results show that software testing using the Black Box Testing method with the Equivalence Partitions technique in the library information system still contains forms whose output results are not in accordance with what was expected in the design because errors occur. From the results of system validity calculations, the book library information system obtained a score of 75%, indicating that the system still has several errors that need to be corrected. It is possible that the error that occurred was caused by an error in the line of code (syntax) in the book library information system. Therefore, it is necessary to update the syntax to fix these bugs and improve system quality.

Keywords: Functional Testing; Software; Library Information Systems; Black Box Testing Methods; Equivalence Partitions

1. PENDAHULUAN

Pengujian perangkat lunak merupakan proses pelaksanaan suatu program dengan tujuan menemukan suatu masalah, sebuah tes dikatakan berhasil apabila dapat mengungkap sebuah Error yang akan muncul pada perangkat lunak. Oleh karena itu, pengujian perangkat lunak sangat penting dalam mengurangi resiko kesalahan dan meningkatkan kualitas software yang dikembangkan (Ahrizal et al., 2020). Aplikasi Sistem Informasi Perpustakaan Buku adalah sistem untuk pendataan pelanggan yang ingin menyewa atau meminjam buku di perpustakaan.

Lamanya waktu yang diperlukan untuk melakukan pengujian menunjukkan betapa pentingnya sebuah perangkat lunak diuji sebelum produk perangkat lunak tersebut diserahkan kepada pengguna. Dengan demikian, pengujian perangkat lunak dapat membantu mengurangi resiko kesalahan dan meningkatkan kualitas perangkat lunak yang dikembangkan, sehingga dapat memberikan hasil yang lebih efektif dan efisien kepada pengguna (Kurniawan, 2019). Dalam



pengujian black-box, tidak diperlukan pengetahuan tentang struktur internal software, namun hanya memeriksa apakah aplikasi dapat berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Dengan demikian, pengujian black-box dapat membantu mengidentifikasi kesalahan pada aplikasi dan meningkatkan kualitas software yang dikembangkan (Febrian et al., 2020).

Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem validasi data yang lebih efektif dan efisien untuk mengurangi resiko kesalahan dan meningkatkan kinerja system (Agustian et al., 2020). Oleh karena itu, pengujian ini harus dilakukan secara sistematis dan metode yang tepat untuk memastikan bahwa sistem aplikasi dapat berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Dengan demikian, hasil pengujian dapat membantu meningkatkan kualitas sistem aplikasi dan memastikan bahwa sistem dapat digunakan dengan efektif dan efisien (Cholifah et al., 2018). Proses perkembangan. Kegagalan saat pengujian perangkat lunak dapat menyebabkan produk yang dihasilkan tidak berjalan dengan baik dan dapat menimbulkan suatu kerugian yang besar, sehingga sangat penting untuk melakukan pengujian yang teliti dan sistematis untuk mengurangi resiko kesalahan dan meningkatkan kualitas perangkat lunak yang dikembangkan (Aziz et al., 2020).

Sejumlah peneliti melakukan penelitian pada pengujian perangkat lunak secara Black-Box dalam penelitiannya, merekamembahas tentang pengujian reduksi dengan menggunakan input-output perangkat lunak sebagai kategori yang akan dicari kesalahannya dengan metode Black-Box. Dalam penelitian ini, peneliti menggunakan teknik pengujian reduksi untuk mengidentifikasi kesalahan pada perangkat lunak, dengan fokus pada input dan output yang dapat mempengaruhi kinerja perangkat lunak. Dengan demikian, penelitian ini dapat membantu meningkatkan kualitas perangkat lunak dan mengurangi resiko kesalahan yang dapat terjadi ketika digunakan oleh pengguna (Komarudin, 2019).

Dengan demikian, pengujian perangkat lunak dapat membantu mengurangi resiko kesalahan dan meningkatkan keamanan serta kualitas perangkat lunak yang dikembangkan (Sinulingga et al., 2020).

Sedangkan untuk *Black Box Testing*, melakukan pengujian berdasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi- fungsi yang ada pada aplikasi dan kesesuaian alur fungsi dengan sistem kerja yang diinginkan perancangannya.

2. METODE

Pengujian sistem ini bertujuan untuk melihat apakah sistem informasi perpustakaan buku yang telah dibuat sudah layak untuk dipergunakan dan sesuai dengan tujuan awal pembuatan sistem tersebut. Sehingga sangat perlu untuk melakukan pengujian sistem informasi perpustakaan buku untuk mengurangi terjadinya kesalahan yang merugikan dan mengganggu fungsi sistem tersebut. Pengujian sistem informasi perpustakaan buku juga untuk mencari kesalahan pada tiap fungsi yang berfokus pada masukan sistem data, tampilan, pemakaian memori, dan kecepatan eksekusi data untuk memastikan kualitas sistem yang optimal. Dengan demikian, pengujian dapat membantu mengidentifikasi masalah-masalah yang mungkin terjadi dan mengurangi resiko kesalahan (Sinulingga et al., 2020).

Metode *Blackbox Testing* adalah sebuah metode yang dipakai untuk menguji sistem informasi perpustakaan buku tanpa harus memperhatikan detail sistem, hanya dengan memeriksa hasil output yang diberikan oleh sistem. Pengujian *Blackbox Testing* ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing pada sistem informasi perpustakaan buku untuk memastikan bahwa sistem dapat bekerja dengan benar dan sesuai dengan kebutuhan pengguna. Tidak ada upaya untuk mengetahui kode program apa yang dipakai untuk menghasilkan output. Dengan demikian, metode *Blackbox Testing* dapat membantu mengidentifikasi kesalahan pada software tanpa memerlukan pengetahuan tentang struktur internal software, sehingga dapat meningkatkan efisiensi dan efektifitas pengujian (Komarudin, 2019).

Pengujian *Blackbox Testing* yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari sistem informasi perpustakaan buku untuk memastikan bahwa sistem

dapat bekerja dengan benar dan sesuai dengan kebutuhan pengguna. Dengan demikian, pengujian ini dapat membantu mengidentifikasi kesalahan pada perangkat lunak dan meningkatkan kualitas software yang dikembangkan. Oleh karena itu, pengujian ini sangat penting untuk memastikan bahwa perangkat lunak dapat berfungsi dengan baik dan sesuai dengan kebutuhan pengguna (Astuti, 2018).

Sehingga dalam melakukan pengujian sistem informasi perpustakaan buku harus memilih teknik yang tepat, yaitu teknik *Blackbox* Testing yang dapat menemukan kesalahan yang belum terdeteksi sehingga dapat meningkatkan kualitas sistem dan memastikan keandalannya. Oleh karena itu, pengembang software harus memilih teknik pengujian yang sesuai dengan tujuan pengujian dan sistem yang dikembangkan, sehingga dapat mengurangi resiko kesalahan dan meningkatkan keamanan serta kualitas software yang dikembangkan (Debiyanti et al., 2020).

Metode Black Box Testing adalah pendekatan pengujian perangkat lunak yang memperlakukan sistem sebagai kotak hitam, di mana tester tidak memperhatikan struktur internal perangkat lunak. Pengujian ini berfokus pada input dan output yang dihasilkan oleh sistem tanpa memperhatikan bagaimana prosesnya dilakukan di dalamnya. Berikut adalah langkah-langkah yang dapat diikuti oleh pemula dalam melakukan pengujian fungsional perangkat lunak sistem informasi perpustakaan dengan metode Black Box Testing:

1. Analisis Requirement: Mulailah dengan memahami kebutuhan fungsional sistem informasi perpustakaan. Identifikasi fungsi-fungsi utama yang harus diuji.
2. Pembuatan Kasus Uji: Berdasarkan requirement, buatlah kasus uji yang mencakup skenario penggunaan yang berbeda. Pastikan untuk mencakup berbagai kemungkinan input dan situasi yang mungkin terjadi.
3. Identifikasi Boundary Conditions: Tentukan batasan-batasan input yang sah dan tidak sah untuk setiap fungsi yang diuji. Periksa apakah sistem berperilaku sesuai dengan yang diharapkan dalam situasi batas tersebut.
4. Pelaksanaan Pengujian: Lakukan pengujian sesuai dengan kasus uji yang telah disiapkan. Masukkan input yang sesuai dan periksa apakah output yang dihasilkan konsisten dengan yang diharapkan.
5. Pemantauan dan Pemecahan Masalah: Catat hasil pengujian dan identifikasi masalah yang ditemukan. Lakukan debugging jika diperlukan dan pastikan untuk memperbaiki kesalahan yang terdeteksi.
6. Pelaporan: Buat laporan yang mencakup hasil pengujian, masalah yang ditemukan, dan langkah-langkah yang diambil untuk memperbaikinya. Laporan ini dapat digunakan sebagai referensi untuk perbaikan di masa mendatang.

3. ANALISA DAN PEMBAHASAN

Pengujian dilakukan pada sistem informasi perpustakaan dengan tujuan untuk mengetahui kekurangan yang ada pada sistem sebelum sistem digunakan oleh pengguna. Hasil dari pengujian *Blackbox* Testing jika ditemukan kelemahan pada bagian sistem maka pengguna bisa mengembangkan dan memperbaiki pada bagian sistem tersebut yang dianggap lemah pada sistem informasi perpustakaan buku untuk meningkatkan kualitas dan keandalan sistem. Dengan demikian, pengujian dapat membantu meningkatkan kualitas sistem dan mengurangi resiko kesalahan, sehingga pengguna dapat menggunakan sistem dengan lebih efektif dan efisien (Sinulingga et al., 2020). Beberapa hasil pengujian yang umum meliputi penemuan bug pada fungsi pencarian buku, kesalahan dalam proses peminjaman, dan masalah dengan sistem pengembalian buku. Dengan demikian, hasil pengujian ini dapat membantu mengurangi resiko kesalahan dan meningkatkan kualitas sistem perpustakaan:

- a. Pengujian Fungsi-Fungsi Utama: Metode Black Box Testing memungkinkan pengujian terhadap fungsi-fungsi utama perangkat lunak sistem informasi perpustakaan, seperti pencarian buku, peminjaman dan pengembalian buku, dan manajemen anggota perpustakaan.

- b. Pengujian Boundary Conditions: Dengan memperhatikan batasan-batasan input yang telah ditentukan, pengujian ini berhasil mengidentifikasi bagaimana sistem merespons dalam situasi batas, seperti saat pengguna memasukkan data yang tidak valid atau mencoba memanipulasi sistem.
- c. Identifikasi Masalah: Melalui pengujian, beberapa masalah atau bug dalam perangkat lunak dapat diidentifikasi, termasuk kesalahan dalam logika bisnis, antarmuka pengguna yang tidak responsif, atau kinerja sistem yang buruk saat memproses permintaan.
- d. Verifikasi Konsistensi Output: Pengujian fungsional juga memastikan bahwa output yang dihasilkan oleh sistem konsisten dengan yang diharapkan, baik dalam format maupun konten.

Pembahasan mengenai hasil pengujian ini memberikan pemahaman yang mendalam tentang keandalan dan kualitas perangkat lunak sistem informasi perpustakaan. Dengan menggunakan metode Black Box Testing, pengembang dapat lebih percaya diri dalam melepaskan versi perangkat lunak yang stabil dan berkinerja baik kepada pengguna.

4. IMPLEMENTASI

4.1 Tampilan Data Sebelum Login



No	Judul	Author	Tahun Terbit	Deskripsi	Penyimpanan	Harga	Detail
1	Batman	Bat	2002	Buku yang berlatar mengenai kehidupan Batman	Buk C	Rp 400.000	Detail
2	Spiderman	John	2018	Buku yang berlatar mengenai kehidupan Spiderman	Buk A	Rp 11.000	Detail
3	Thor	Bat	2000	Buku yang berlatar mengenai kehidupan Thor	Buk A	Rp 22.000	Detail
4	Superman	John	2018	Buku yang berlatar mengenai kehidupan Superman	Buk A	Rp 11.000	Detail
5	Warrior	John	2019	Buku yang berlatar mengenai kehidupan Warrior	Buk B	Rp 10.000	Detail

Gambar 1. Perancangan Halaman Data Buku Bagi Pengguna/User

Pada gambar diatas menjelaskan perancangan halaman data buku bagi pengguna/user, halaman ini hanya di akses oleh user. Halaman ini dirancang khusus untuk diakses oleh pengguna/user. Halaman ini memungkinkan pengguna untuk melihat informasi lengkap mengenai buku-buku yang tersedia di database, termasuk judul, penulis, tahun terbit, deskripsi, lokasi penyimpanan, dan harga. Dengan adanya halaman ini, pengguna dapat dengan mudah mencari dan memperoleh informasi mengenai buku yang mereka butuhkan.

4.2 Tampilan Login



Login

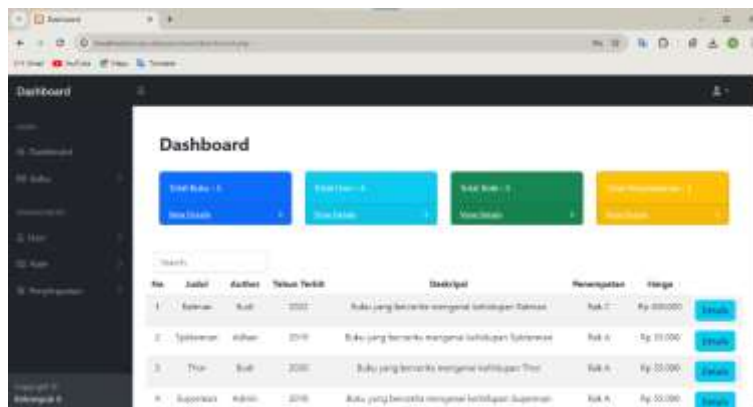
Daftar
Login

Masuk ke dashboard > >>>

Gambar 2. Perancangan Halaman Login

Pada gambar diatas menjelaskan perancangan halaman login, dimana user apabila ingin mengakses system harus memasukan username dan password lalu tekan tombol login. Jika username dan password benar maka akan di arahkan ke halaman utama.

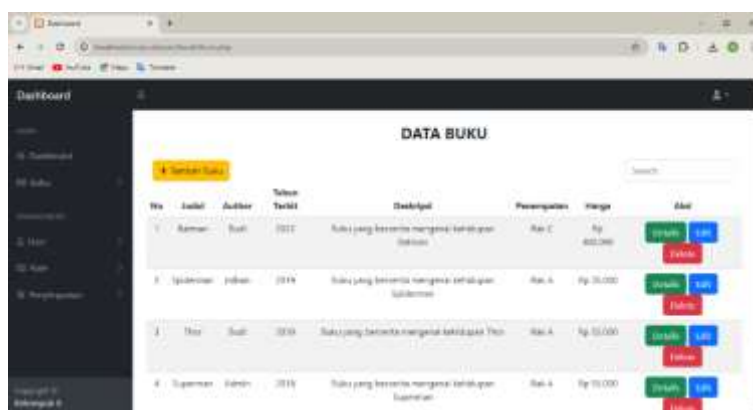
4.3 Tampilan Dashboard



Gambar 3. Perancangan Halaman Dashboard

Pada gambar diatas menjelaskan perancangan halaman dashboard, dimana user dapat melihat data pada masing-masing icon dan beberapa menu disebelah kiri.

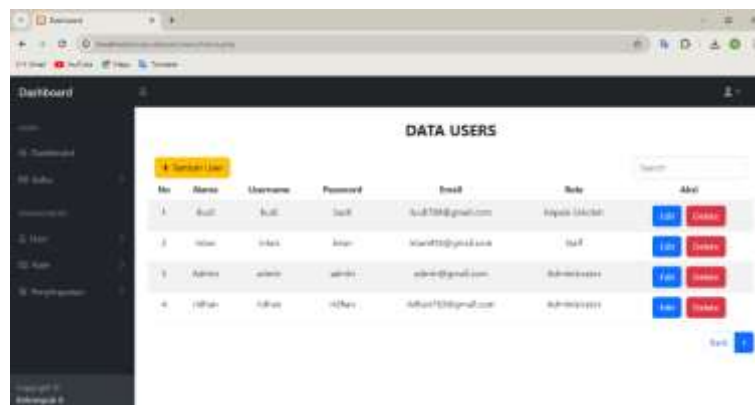
4.4 Tampilan Buku



Gambar 4. Perancangan Halaman Menampilkan Data Buku

Pada gambar diatas menjelaskan perancangan halaman menampilkan data buku yang berfungsi untuk menampilkan data data buku yang tersimpan dalam database, halaman ini hanya dapat di akses oleh admin untuk melakukan proses pengelolaan data seperti tambah data, ubah data dan hapus data buku.

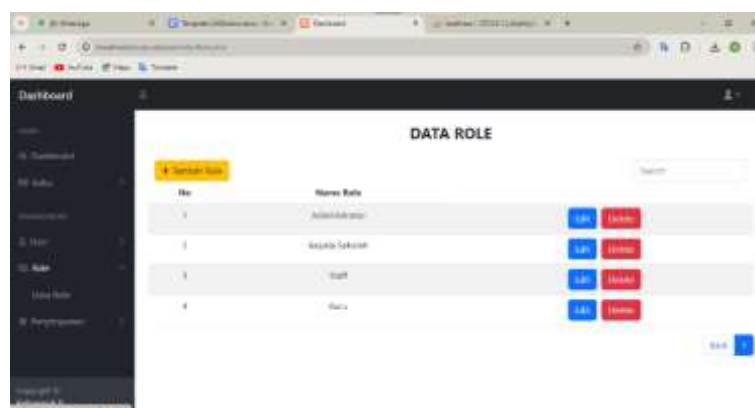
4.5 Tampilan User



Gambar 5. Halaman Menampilkan Data Dari Menu User Management

Pada gambar di atas menjelaskan perancangan halaman menampilkan data dari menu User Management berfungsi untuk menampilkan data User Management yang tersimpan dalam Database, halaman ini dapat diakses oleh admin pada sistem informasi perpustakaan buku. untuk melakukan proses tambah User, edit User, hapus User.

4.6 Tampilan Role/Peran



Gambar 6. Halaman Menampilkan Data Dari Menu User Management

Pada gambar di atas menjelaskan perancangan halaman role/peran. Halaman ini berfungsi untuk mengetahui dan mengelola peran-peran yang ada di database. Setiap peran memiliki hak akses dan tanggung jawab tertentu dalam sistem, dan halaman ini memungkinkan administrator untuk melihat, menambah, mengedit, atau menghapus peran sesuai kebutuhan. Dengan demikian, pengelolaan peran menjadi lebih terstruktur dan memudahkan dalam pengaturan hak akses pengguna.

4.7 Tampilan Data Tempat Penyimpanan



Gambar 7. Halaman Perancangan Halaman Data Tempat Penyimpanan Buku

Pada gambar diatas menjelaskan perancangan halaman data tempat penyimpanan buku. Halaman ini berfungsi untuk mengetahui dan mengelola lokasi-lokasi penyimpanan buku yang ada di dalam database. Setiap tempat penyimpanan memiliki informasi detail mengenai lokasi dan kapasitasnya, serta dapat digunakan untuk mengatur penempatan buku dengan lebih efektif. Dengan adanya halaman ini, pengelolaan tempat penyimpanan buku menjadi lebih terstruktur dan terorganisir, memudahkan administrator dalam pengaturan dan pemantauan lokasi penyimpanan buku.

4.8 Black Box Testing

No	Skenario Pengujian	Hasil yang diharapkan	Hasil pengujian	Status
1	Menekan Tombol Login	Menampilkan halaman Dashboard	Sistem menampilkan halaman Dashboard	<i>Valid</i>
2	Menekan Navigasi Buku	Dapat menampilkan halaman buku	Sistem menampilkan halaman buku	<i>Valid</i>
3	Menekan Navigasi User	Dapat menampilkan halaman user	Sistem menampilkan halaman user	<i>Valid</i>
4	Menekan Navigasi Role	Dapat menampilkan halaman role	Sistem menampilkan halaman role/peran	<i>Valid</i>
5	Menekan Navigasi Penyimpanan	Dapat menampilkan halaman penyimpanan	Sistem menampilkan halaman rak penyimpanan	<i>Valid</i>
6	Menekan tombol tambah data buku	Dapat menambah data buku	Sistem berhasil menambah data buku	<i>Valid</i>
7	Menekan tombol edit buku	Dapat mengubah data buku	Sistem berhasil mengubah data buku	<i>Valid</i>
8	Menekan tombol hapus	Dapat menghapus data buku	Sistem berhasil menghapus data buku	<i>Valid</i>
9	Menekan tombol tambah user	Dapat menambah data user	Sistem berhasil menambah data user	<i>Valid</i>

10	Menekan tombol edit user	Dapat mengubah data user	Sistem berhasil mengubah data user	<i>Valid</i>
11	Menekan tombol hapus user	Dapat menghapus data user	Sistem berhasil menghapus data user	<i>Valid</i>
12	Menekan tombol tambah data penyimpanan	Dapat menambah data penyimpanan	Sistem berhasil menambah data penyimpanan	<i>Valid</i>
13	Menekan tombol edit penyimpanan	Dapat mengubah data penyimpanan	Sistem berhasil mengubah data penyimpanan	<i>Valid</i>
14	Menekan tombol hapus penyimpanan	Dapat menghapus data penyimpanan	Sistem berhasil menghapus data penyimpanan	<i>Valid</i>

5. KESIMPULAN

Pengujian fungsional perangkat lunak sistem informasi perpustakaan dengan metode Black Box Testing membuktikan diri sebagai pendekatan yang efektif dalam memvalidasi fungsi-fungsi sistem dan mengidentifikasi potensi masalah. Kesimpulan yang dapat diambil dari artikel ini adalah bahwa penggunaan metode *Blackbox* Testing pada sistem informasi perpustakaan buku dapat membantu meningkatkan kualitas dan keandalan sistem, serta memastikan bahwa sistem dapat bekerja dengan benar dan sesuai dengan kebutuhan pengguna.

- Metode Black Box Testing memungkinkan tester untuk fokus pada input dan output sistem tanpa perlu memperhatikan struktur internal perangkat lunak.
- Pengujian fungsional ini memastikan bahwa sistem informasi perpustakaan berperilaku sesuai dengan kebutuhan fungsional yang telah ditentukan.

Identifikasi masalah melalui pengujian memungkinkan pengembang untuk melakukan perbaikan yang diperlukan untuk meningkatkan kualitas dan keandalan perangkat lunak.

REFERENCES

- Agustian, A., Andryani, I., Khoerunisa, S., Pangestu, A., & Saifudin, A. (2020). Implementasi Teknik Equivalence Partitioning pada Pengujian Aplikasi E-learning Berbasis Web. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(3), 178. <https://doi.org/10.32493/jtsi.v3i3.5371>
- Ahrizal, D., Miftah, M. K., Kurniawan, R., Zaelani, T., & Yulianti, Y. (2020). Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing. *Jurnal Informatika Universitas Pamulang*, 5(1), 73. <https://doi.org/10.32493/informatika.v5i1.4338>
- Astuti, P. (2018). Penggunaan Metode Black Box Testing (Boundary Value Analysis) Pada Sistem Akademik (Sma/Smk). *Faktor Exacta*, 11(2), 186. <https://doi.org/10.30998/faktorexacta.v11i2.2510>
- Aziz, I. A., Setiawan, B., Khanh, R., Nurdiyansyah, G., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Sistem Kasir Berbasis Website Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(2), 82. <https://doi.org/10.32493/jtsi.v3i2.4693>
- Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2018). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 3(2), 206. <https://doi.org/10.30998/string.v3i2.3048>
- Debiyanti, D., Sutrisna, S., Budrio, B., Kamal, A. K., & Yulianti, Y. (2020). Pengujian Black Box pada Perangkat Lunak Sistem Penilaian Mahasiswa Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(2), 162. <https://doi.org/10.32493/informatika.v5i2.5446>
- Febrian, V., Ramadhan, M. R., Faisal, M., & Saifudin, A. (2020). Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode *Blackbox*. *Jurnal Informatika Universitas Pamulang*, 5(1), 61. <https://doi.org/10.32493/informatika.v5i1.4340>



- Komarudin, M. (2019). Pengujian Perangkat Lunak Metode Black-Box Berbasis Equivalence Partitions pada Aplikasi Sistem Informasi di Sekolah. *Jurnal Mikrotik*, 06(3), 02–16.
- Kurniawan, T. A. (2019). Pengujian Struktur Program Dengan Pengujian Jalur Dasar (Basis Path Testing) : Teori Dan Aplikasi. *EECCIS*, 1, 29–32.
- Sinulingga, A. R., Zuhri, M., Mukti, R. B., Syifa, Z., & Saifudin, A. (2020). Pengujian Black Box pada Sistem Aplikasi Informasi Data Kinerja Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(1), 9. <https://doi.org/10.32493/jtsi.v3i1.4303>