

Evaluasi Fungsionalitas Website Yuma Laundry melalui Pengujian Black Box dengan Teknik Boundary Value Analysis

Sharfina Andzani Minhalina¹, Denty Nirwana Bintang², Ari Dian Prastyo³, Surya Agung⁴,
Muhammad Yordi Septian⁵, Aditya Wicaksono^{*6}, Muhammad Nasir⁷

¹⁻⁷Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi, IPB University, Indonesia
Email: ¹zsharfina@apps.ipb.ac.id, ²1502denty@apps.ipb.ac.id, ³aridprastyo@apps.ipb.ac.id,
⁴pesatsurya@apps.ipb.ac.id, ⁵yordiseptian@apps.ipb.ac.id, ⁶adityawicaksono@apps.ipb.ac.id,
⁷m_nasir@apps.ipb.ac.id

Abstrak

Studi ini mengevaluasi fungsionalitas situs web Yuma Laundry menggunakan metode *Blackbox Testing* dengan teknik *Boundary Value Analysis* (BVA). Situs web ini dikembangkan untuk meningkatkan aksesibilitas layanan dan efisiensi operasional dalam pengelolaan data pelanggan untuk bisnis laundry. Penelitian ini bertujuan untuk menguji kemampuan sistem dalam menangani berbagai skenario masukan kritis dengan berfokus pada batasan masukan di beberapa bidang. Proses pengujian melibatkan evaluasi bidang-bidang penting seperti nama pelanggan, jumlah barang, dan jenis layanan, memastikan semuanya sesuai dengan batas masukan yang telah ditentukan sebelumnya. Hasilnya menunjukkan tingkat keberhasilan 60% dari 20 kasus uji, dengan validasi efektif di bidang terkait tanggal, sementara kekurangan diamati dalam penanganan masukan untuk nama pelanggan, jumlah barang, dan jenis layanan. Masalah-masalah ini terutama terkait dengan pesan kesalahan yang tidak jelas atau hilang ketika masukan dibiarkan kosong atau melampaui batas yang ditentukan. Temuan tersebut mengungkapkan area-area di mana validasi masukan sistem perlu ditingkatkan. Rekomendasi mencakup penyempurnaan mekanisme validasi dan penyediaan pesan kesalahan yang lebih jelas untuk meningkatkan keandalan sistem dan pengalaman pengguna secara keseluruhan. Studi ini berkontribusi pada pengembangan sistem informasi berbasis web yang andal bagi penyedia layanan lokal, memastikan akurasi yang lebih tinggi dan interaksi pengguna yang lebih baik bagi administrator dan pelanggan.

Kata kunci: Blackbox Testing, Boundary Value Analysis, Evaluasi Fungsionalitas, Laundry, Website

Abstract

This study evaluates the functionality of the Yuma Laundry website using black-box testing with the Boundary Value Analysis (BVA) technique. The website was developed to improve service accessibility and operational efficiency in customer data management for laundry businesses. This research aims to test the system's ability to handle various critical input scenarios by focusing on input boundaries across several fields. The testing process involves evaluating essential fields such as customer name, item quantity, and service type, ensuring they conform to predetermined input limits. The results indicate a 60% success rate from 20 test cases, with effective validation in date-related fields, while deficiencies were observed in handling inputs for customer names, item quantities, and service types. These issues were primarily related to unclear or missing error messages when the inputs were left empty or exceeded defined limits. The findings reveal areas where the system's input validation needs improvement. Recommendations include refining validation mechanisms and providing clearer error messages to enhance the system's reliability and overall user experience. This study contributes to the development of reliable, web-based information systems for local service providers, ensuring higher accuracy and improved user interaction for both administrators and customers.

Keywords: Blackbox Testing, Boundary Value Analysis, Functionality Evaluation, Laundry, Website

This work is an open access article and licensed under a Creative Commons Attribution-NonCommercial ShareAlike 4.0 International (CC BY-NC-SA 4.0)



1. PENDAHULUAN

Kemajuan teknologi informasi telah mendorong berbagai sektor, termasuk industri jasa yang juga mulai mengadopsi digitalisasi[1]. Layanan laundry adalah salah satu sektor yang beralih ke digitalisasi[2]. Layanan berbasis daring menjadi solusi untuk meningkatkan aksesibilitas dan efisiensi

dalam pengelolaan data pelanggan serta promosi layanan[3]. Salah satu contohnya adalah Website Yuma Laundry, yang dirancang untuk memudahkan konsumen memperoleh informasi tentang daftar layanan, biaya jasa, status antrian, dan pengelolaan data secara lebih terstruktur.

Pengujian perangkat lunak menjadi langkah penting untuk memastikan sistem berjalan sesuai dengan spesifikasi yang telah ditetapkan[4]. Salah satu metode pengujian yang sering digunakan adalah *blackbox testing*, pengujian yang menguji fungsionalitas sistem tanpa perlu mengetahui struktur internalnya[5]. *Blackbox testing* cocok diterapkan pada aplikasi berbasis website yang berorientasi pada pengguna akhir karena dapat mengevaluasi pengalaman pengguna secara langsung[6].

Boundary value analysis dalam *blackbox testing* adalah metode yang berfokus untuk menguji nilai-nilai batas dari input yang masuk ke dalam sistem[7]. Teknik ini berlandaskan prinsip bahwa kesalahan sering muncul di sekitar nilai batas, baik itu batas minimum, maksimum, atau nilai ekstrem[8]. Dengan menguji nilai-nilai batas ini memungkinkan deteksi kesalahan lebih cepat dan akurat, karena banyak sistem yang gagal dalam menangani input di titik ekstrem ini[9]. *Boundary value analysis* sering digunakan dalam pengujian *blackbox* karena dapat mempersempit ruang lingkup pengujian dan memastikan sistem dapat berfungsi dengan baik dalam situasi yang paling kritis[10].

Beberapa penelitian sebelumnya telah menggunakan teknik *boundary value analysis testing* untuk mengevaluasi fungsionalitas sistem berbasis web. Penelitian oleh Dobsław et al.[11] menunjukkan bahwa teknik ini dapat menemukan anomali pada batas input yang sering terlewat oleh metode pengujian lainnya. Ijudin & Saifudin[12] juga mengungkapkan bahwa teknik ini efektif untuk menemukan kesalahan yang sering muncul di sekitar batas minimum dan maksimum dari parameter input. Dengan menguji nilai-nilai batas, teknik ini memastikan bahwa sistem dapat berfungsi dengan baik dalam menangani skenario input yang paling kritis[13]. Penelitian oleh Kartono et al.[14] mengevaluasi sistem pemesanan daring menggunakan *boundary value analysis* dan menyimpulkan bahwa metode ini efektif dalam menguji skenario input ekstrem yang sering diabaikan oleh pendekatan pengujian lainnya, serta mampu meningkatkan keandalan dan efisiensi sistem secara keseluruhan. Selain itu, penggunaan *boundary value analysis* membantu mengurangi waktu pengujian tanpa mengorbankan cakupan pengujian yang diperlukan untuk memastikan kualitas sistem[15].

Namun, studi yang memfokuskan pada sistem informatif berbasis web untuk layanan laundry masih jarang dilakukan, sehingga memberikan peluang untuk penelitian lebih lanjut. Sebagai layanan lokal, Yuma Laundry mengembangkan website untuk memberikan informasi layanan seperti daftar layanan, biaya jasa, dan pengelolaan data pelanggan. Dalam penelitian ini, pengujian dilakukan menggunakan metode *blackbox testing* dengan teknik *boundary value analysis* untuk memastikan semua kombinasi masukan dan keluaran pada fitur website berjalan sesuai spesifikasi. Penelitian ini bertujuan untuk mengevaluasi fungsionalitas Website Yuma Laundry serta memberikan rekomendasi perbaikan berdasarkan hasil pengujian. Dengan demikian, hasil penelitian ini diharapkan dapat berkontribusi pada pengembangan sistem informasi berbasis web untuk layanan lokal serupa.

2. METODE PENELITIAN

Objek penelitian ini adalah website Yuma Laundry. Pada penelitian ini, pengujian website Yuma Laundry dilakukan menggunakan metode *blackbox testing* dengan teknik *boundary value analysis* untuk memeriksa batas nilai input pada sistem[16].

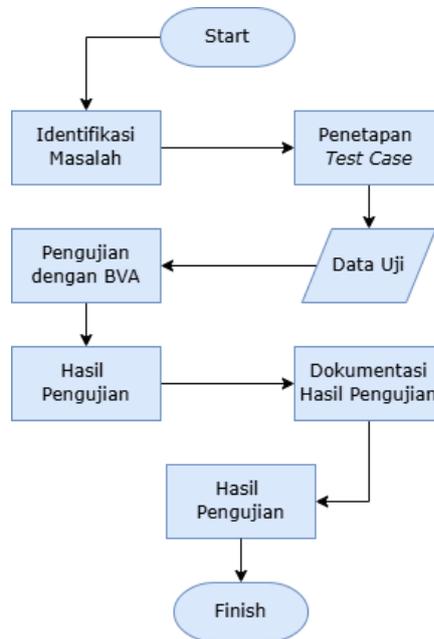
2.1. Blackbox Testing

Blackbox testing adalah metode pengujian perangkat lunak yang berfokus pada antarmuka aplikasi, fungsionalitas yang tersedia, dan kesesuaian alur kerja dengan kebutuhan pengguna tanpa melibatkan pemeriksaan kode sumber program[17]. Tujuan dari pengujian ini adalah mengidentifikasi kesalahan dalam fungsi, antarmuka, struktur data, kinerja, dan inisialisasi dan terminasi[18].

2.2. Boundary Value Analysis

Boundary value analysis dipilih sebagai teknik utama dalam penelitian ini karena efektivitasnya dalam mendeteksi kesalahan pada batas nilai input[19]. Kesalahan ini sering kali terlewat oleh metode pengujian lain seperti *equivalence partitioning*. Teknik *boundary value analysis* fokus pada pengujian nilai batas minimum, maksimum, nilai kosong, dan nilai ekstrem lainnya yang seringkali menjadi

sumber kesalahan sistem. Pada pengujian ini, tahapan-tahapan pengujian dilakukan berdasarkan studi sebelumnya oleh Yulistina et al.[16] yang menunjukkan bahwa teknik *boundary value analysis* efektif dalam menemukan kesalahan sistem terkait validasi input.



Gambar 1. Tahapan Pengujian *Blackbox* dengan Teknik *Boundary Value Analysis*

Gambar 1 menggambarkan proses pengujian yang dimulai dengan tahap identifikasi masalah hingga hasil pengujian. Pada tahap identifikasi masalah, ditentukan fitur-fitur yang akan diuji serta jenis permasalahan yang mungkin muncul dari input pengguna. Setelah masalah ditentukan, langkah berikutnya adalah penetapan *test case* yaitu menyusun kasus uji berdasarkan teknik *boundary value analysis* yang berfokus pada pengujian nilai batas minimum, maksimum, nilai kosong, dan nilai ekstrim lainnya. Data uji kemudian disiapkan sesuai dengan *test case* tersebut. Proses berlanjut ke tahap pengujian dengan *boundary value analysis*, di mana sistem diuji dengan data yang telah disiapkan untuk melihat apakah nilai batas input menyebabkan kesalahan. Hasil dari pengujian kemudian dicatat untuk menunjukkan apakah sistem berfungsi dengan baik atau ada kegagalan pada input tertentu. Hasil pengujian didokumentasikan untuk referensi dan analisis lebih lanjut.. Proses pengujian berakhir dengan menentukan persentase hasil pengujian sekaligus dokumentasi untuk referensi dan analisis lebih lanjut [20].

Untuk memperdalam tahapan pada Gambar 1, proses dimulai dengan identifikasi fitur-fitur sistem yang berpotensi memiliki risiko kesalahan input, di mana pemilihan difokuskan pada kolom-kolom yang bersifat wajib diisi dan memiliki batas nilai eksplisit (misalnya numerik dan waktu). Contohnya, kolom “Kuantitas Barang” dan “Tanggal Mulai” diprioritaskan karena kesalahan pada input ini berisiko besar terhadap logika sistem. Setelah fitur ditentukan, penyusunan test case dilakukan dengan mempertimbangkan skenario input ekstrem seperti nilai minimum (0 atau kosong), maksimum (misal 101), dan di luar batas (negatif atau terlalu panjang). Data uji yang disiapkan mencakup input valid maupun invalid, lalu diuji untuk melihat apakah sistem mampu memberikan validasi dan umpan balik yang sesuai. Semua hasil dicatat secara sistematis dan dianalisis lebih lanjut untuk menentukan efektivitas sistem dalam menangani batas input.

Tambah Data Waiting

Nama Pelanggan:

Nama Barang:

Kuantitas Barang:

Status:

Jenis Layanan:

Tanggal Mulai:

Tanggal Selesai:

Simpan

Gambar 2. *Form* Membuat Antrian

Formulir pada Gambar 2 memiliki beberapa kolom yang harus diisi untuk menambahkan data antrian dengan benar, yaitu “Nama Pelanggan”, “Nama Barang”, “Kuantitas Barang”, “Status”, “Jenis Layanan”, “Tanggal Mulai”, dan “Tanggal Selesai”. Tanda panah pada gambar menunjukkan kolom-kolom kritis yang diprioritaskan dalam pengujian karena memiliki batas nilai eksplisit dan berpotensi menimbulkan kesalahan sistem, seperti “Kuantitas Barang” (input numerik) dan “Tanggal Mulai” (input waktu).

Berdasarkan form input yang ditampilkan, setiap kolom diuji untuk memastikan bahwa input sesuai dengan batas nilai yang diperbolehkan. Dalam pengujian, teknik *boundary value analysis* digunakan untuk mengidentifikasi batas input pada setiap kolom. Metode ini memastikan bahwa sistem dapat menangani input pada nilai minimum dan maksimum yang diizinkan, serta mendeteksi input yang berada di luar rentang valid.

Berikut adalah kasus uji untuk setiap kolom yang disusun berdasarkan teknik *boundary value analysis*, seperti yang dapat dilihat pada tabel 1 berikut:

Tabel 1. *Test Case Form* Membuat Antrian

ID	Deskripsi Pengujian	Hasil yang Diharapkan
TC01	Isi <i>field</i> “Nama Pelanggan” dengan minimum 1 karakter, misalnya “A”	Sistem menerima input dan menyimpan data
TC02	Kosongkan <i>field</i> “Nama Pelanggan”	Menampilkan pesan kesalahan dan tidak menyimpan data
TC03	Isi <i>field</i> “Nama Pelanggan” dengan maksimum 50 karakter	Sistem menerima input dan menyimpan data
TC04	Isi <i>field</i> “Nama Pelanggan” dengan lebih dari 50 karakter	Menampilkan pesan kesalahan dan tidak menyimpan data
TC05	Isi <i>field</i> “Nama Barang” dengan minimum 1 karakter, misalnya “B”	Sistem menerima input dan menyimpan data
TC06	Kosongkan <i>field</i> “Nama Barang”	Menampilkan pesan kesalahan dan tidak menyimpan data
TC07	Isi <i>field</i> “Nama Barang” dengan maksimum 50 karakter	Sistem menerima input dan menyimpan data

TC08	Isi <i>field</i> “Nama Barang” dengan lebih dari 50 karakter	Menampilkan pesan kesalahan dan tidak menyimpan data
TC09	Isi <i>field</i> “Kuantitas Barang” dengan minimum valid (1)	Sistem menerima input dan menyimpan data
TC10	Isi <i>field</i> “Kuantitas Barang” dengan di bawah minimum valid (0)	Menampilkan pesan kesalahan dan tidak menyimpan data
TC11	Isi <i>field</i> “Kuantitas Barang” dengan maksimal valid (100)	Sistem menerima input dan menyimpan data
TC12	Isi <i>field</i> “Kuantitas Barang” dengan di atas maksimal valid (101)	Menampilkan pesan kesalahan dan tidak menyimpan data
TC13	Isi <i>field</i> “Status” dengan status valid, misalnya “Dalam Antrian”	Sistem menerima input dan menyimpan data
TC14	Kosongkan <i>field</i> “Status”	Menampilkan pesan kesalahan dan tidak menyimpan data
TC15	Isi <i>field</i> “Jenis Layanan” dengan status valid, misalnya “Reguler”	Sistem menerima input dan menyimpan data
TC16	Kosongkan <i>field</i> “Jenis Layanan”	Menampilkan pesan kesalahan dan tidak menyimpan data
TC17	Isi <i>field</i> “Tanggal mulai” valid, misalnya 11/15/2024	Sistem menerima input dan menyimpan data
TC18	Isi <i>field</i> “Tanggal mulai” tidak valid, misalnya 2024/11/15	Menampilkan pesan kesalahan dan tidak menyimpan data
TC19	Isi <i>field</i> “Tanggal selesai” valid, misalnya 11/16/2024	Sistem menerima input dan menyimpan data
TC20	Isi <i>field</i> “Tanggal mulai” dengan tanggal sebelum tanggal mulai, misalnya 11/10/2024	Menampilkan pesan kesalahan dan tidak menyimpan data

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pengujian

Pengujian *blackbox* dengan teknik *boundary value analysis* pada Website Yuma Laundry dilakukan dengan 20 skenario test case yang berfokus pada validasi input di kolom-kolom penting seperti “Nama Pelanggan”, “Kuantitas Barang”, dan “Tanggal”.

Tabel 2. Hasil Uji Test Case Form Membuat Antrian

ID	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil
TC01	Isi <i>field</i> “Nama Pelanggan” dengan minimum 1 karakter, misalnya “A”	Sistem menerima input dan menyimpan data	Berhasil
TC02	Kosongkan <i>field</i> “Nama	Menampilkan pesan kesalahan	Gagal

	Pelanggan”	dan tidak menyimpan data	
TC03	Isi <i>field</i> “Nama Pelanggan” dengan maksimum 50 karakter	Sistem menerima input dan menyimpan data	Berhasil
TC04	Isi <i>field</i> “Nama Pelanggan” dengan lebih dari 50 karakter	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC05	Isi <i>field</i> “Nama Barang” dengan minimum 1 karakter, misalnya “B”	Sistem menerima input dan menyimpan data	Berhasil
TC06	Kosongkan <i>field</i> “Nama Barang”	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC07	Isi <i>field</i> “Nama Barang” dengan maksimum 50 karakter	Sistem menerima input dan menyimpan data	Berhasil
TC08	Isi <i>field</i> “Nama Barang” dengan lebih dari 50 karakter	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC09	Isi <i>field</i> “Kuantitas Barang” dengan minimum valid (1)	Sistem menerima input dan menyimpan data	Berhasil
TC10	Isi <i>field</i> “Kuantitas Barang” dengan di bawah minimum valid (0)	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC11	Isi <i>field</i> “Kuantitas Barang” dengan maksimal valid (100)	Sistem menerima input dan menyimpan data	Berhasil
TC12	Isi <i>field</i> “Kuantitas Barang” dengan di atas maksimal valid (101)	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC13	Isi <i>field</i> “Status” dengan status valid, misalnya “Dalam Antrian”	Sistem menerima input dan menyimpan data	Berhasil
TC14	Kosongkan <i>field</i> “Status”	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC15	Isi <i>field</i> “Jenis Layanan” dengan status valid, misalnya “Reguler”	Sistem menerima input dan menyimpan data	Berhasil
TC16	Kosongkan <i>field</i> “Jenis Layanan”	Menampilkan pesan kesalahan dan tidak menyimpan data	Gagal
TC17	Isi <i>field</i> “Tanggal mulai” valid, misalnya 11/15/2024	Sistem menerima input dan menyimpan data	Berhasil
TC18	Isi <i>field</i> “Tanggal mulai” tidak valid, misalnya 2024/11/15	Tidak bisa menerima input tanggal tidak sesuai format	Berhasil
TC19	Isi <i>field</i> “Tanggal selesai” valid, misalnya 11/16/2024	Sistem menerima input dan menyimpan data	Berhasil
TC20	Isi <i>field</i> “Tanggal mulai” dengan	Tidak bisa menerima input	Berhasil

	tanggal sebelum tanggal mulai, misalnya 11/10/2024	tanggal sebelum tanggal mulai	
--	---	-------------------------------	--

Pengujian *blackbox* dilakukan pada website Yuma Laundry untuk memvalidasi berbagai jenis input yang diterima oleh sistem. Tabel 2 di atas menunjukkan respons sistem terhadap berbagai skenario berdasarkan nilai batas yang ditentukan, apakah sistem berhasil menerima dan menyimpan data, atau menolak data yang tidak valid dengan memberikan pesan kesalahan yang sesuai. Keberhasilan validasi pada beberapa kolom seperti Tanggal Mulai dan Tanggal Selesai menunjukkan bahwa sistem telah mampu menangani format tanggal dengan baik. Namun, masih terdapat beberapa kekurangan pada kolom lainnya.

3.2. Analisis dan Pembahasan

Tingkat keberhasilan dari hasil pengujian sistem pada website Yuma Laundry adalah sebesar 60% dari total 20 test case. Pengujian menunjukkan bahwa sistem berhasil menangani validasi input pada beberapa kolom seperti “Tanggal Mulai” dan “Tanggal Selesai”, di mana semua skenario berjalan sesuai dengan spesifikasi. Keberhasilan ini terjadi karena penggunaan komponen *datepicker* yang secara otomatis membatasi input pada format tanggal yang valid, sehingga mencegah kesalahan input manual. Hal ini menunjukkan bahwa mekanisme validasi pada kolom-kolom tersebut telah berfungsi dengan optimal dan sesuai dengan kebutuhan sistem. Selain itu, adanya validasi ganda di sisi frontend dan backend memperkuat keamanan serta menjaga integritas data.

Namun, meskipun beberapa kolom telah berfungsi dengan baik, terdapat beberapa kekurangan dalam validasi input pada kolom “Nama Pelanggan”, “Nama Barang”, “Kuantitas Barang”, “Status”, dan “Jenis Layanan”. Berikut adalah analisis penyebab kegagalan pada kolom-kolom yang diuji:

1. Kolom “Nama Pelanggan” dan “Nama Barang”
Kedua kolom ini tidak dapat memvalidasi input kosong atau yang melebihi batas karakter maksimum (50 karakter). Hal ini diduga karena validasi yang hanya mengandalkan atribut HTML seperti *maxlength* dan *required* di sisi *frontend* mudah dilewati melalui alat pengembang atau permintaan langsung ke server. Akibatnya, sistem berisiko menyimpan data yang tidak sah tanpa disadari pengguna. Selain itu, kurangnya pesan kesalahan yang eksplisit menunjukkan sistem tidak memiliki mekanisme *error handling* yang terstruktur untuk input karakter alfanumerik.
2. Kolom “Kuantitas Barang”
Kegagalan pada kolom ini terjadi saat input berada di bawah batas minimum (0) atau di atas maksimum (101). Hal ini menunjukkan bahwa proses validasi hanya memeriksa tipe data angka (*integer*) tanpa menerapkan batasan nilai yang diperbolehkan. Tidak adanya atribut *min* dan *max* di *frontend* dan validasi rentang di *backend* menyebabkan kasus-kasus ekstrim (*boundary cases*) dapat masuk tanpa terdeteksi. Dampaknya, data tidak valid tersimpan dalam sistem dan berpotensi mengganggu keakuratan perhitungan maupun laporan.
3. Kolom “Status” dan “Jenis Layanan”
Kedua kolom ini tidak memberikan pesan kesalahan ketika kolom dibiarkan kosong. Hal ini menunjukkan bahwa sistem tidak memiliki validasi yang memaksa pengguna memilih nilai yang sah dari *dropdown*. Ketika nilai kosong tetap dapat diproses, sistem berisiko menyimpan data yang tidak lengkap atau tidak valid. Kondisi ini dapat menimbulkan masalah dalam proses bisnis seperti pelacakan status pesanan atau pengkategorian jenis layanan. Selain itu, tidak adanya umpan balik yang jelas saat terjadi kesalahan juga berdampak negatif terhadap pengalaman pengguna, karena mereka tidak mendapat petunjuk yang memadai untuk memperbaiki input yang salah.

Kegagalan dalam validasi input di beberapa kolom menunjukkan bahwa sistem belum sepenuhnya tangguh dalam menangani semua jenis input dengan baik. Tanpa mekanisme yang efektif untuk mendeteksi dan memberikan umpan balik terhadap input yang salah, pengguna bisa melanjutkan tanpa menyadari kesalahan mereka. Hal ini bisa membuat sistem kurang dapat diandalkan dan membingungkan pengguna tentang cara memperbaiki kesalahan.

Pentingnya memperketat validasi input di semua kolom yang terlibat tidak bisa diabaikan. Setiap kolom harus memiliki mekanisme validasi yang memastikan data sesuai dengan batas yang diizinkan dan semua input wajib (seperti nama pelanggan dan jenis layanan) telah diisi dengan benar. Selain itu, sistem perlu memberikan pesan kesalahan yang jelas dan mudah dipahami. Pesan ini harus memberikan informasi yang cukup untuk membantu pengguna memperbaiki kesalahan mereka tanpa kebingungan. Sebagai contoh, jika kolom Nama Pelanggan kosong, pesan kesalahan yang jelas seperti "Nama pelanggan tidak boleh kosong. Harap isi dengan nama yang valid" dapat membantu menghindari kesalahan yang sama berulang kali.

4. KESIMPULAN

Penelitian ini mengevaluasi fungsionalitas website Yuma Laundry menggunakan metode *blackbox testing* dengan teknik *boundary value analysis*. Hasil pengujian menunjukkan bahwa sistem memiliki tingkat keberhasilan sebesar 60% dari 20 test case yang diuji. Keberhasilan paling signifikan terdapat pada kolom tanggal yang menggunakan komponen *datepicker*, yang berhasil membatasi format input dan memastikan logika kronologis antara tanggal mulai dan selesai. Namun, terdapat kekurangan dalam validasi input untuk kolom "Nama Pelanggan", "Nama Barang", "Kuantitas Barang", "Status", dan "Jenis Layanan", di mana sistem gagal memberikan pesan kesalahan ketika input kosong, melebihi batas karakter, atau berada di luar rentang nilai yang diperbolehkan. Hal ini menunjukkan bahwa sistem belum siap menghadapi berbagai skenario input ekstrem.

Sebagai rekomendasi, pengembang perlu memperkuat mekanisme validasi di sisi backend serta memastikan setiap kolom memiliki validasi yang sesuai dengan jenis dan batas data yang diperbolehkan. Selain itu, penyediaan pesan kesalahan yang informatif dan kontekstual sangat penting untuk membantu pengguna mengenali dan memperbaiki kesalahan input secara mandiri. Implementasi perbaikan ini diharapkan dapat meningkatkan keandalan sistem secara signifikan, mengurangi potensi kerentanan keamanan, serta meningkatkan pengalaman pengguna yang lebih baik dalam berinteraksi dengan sistem Yuma Laundry.

DAFTAR PUSTAKA

- [1] S. Idayanti and S. Siswanto, "Impact of Industry Digitalization on Recruitment of Employment Opportunities," *J. Din. Huk.*, vol. 22, no. 2, p. 406, 2022, doi: 10.20884/1.jdh.2022.22.2.3426.
- [2] F. Hamdani, U. Noviana, and F. Idifitriani, "PENGEMBANGAN APLIKASI MANAJEMEN LAUNDRY BERBASIS WEBVIEW PADA PLATFORM ANDROID (STUDI KASUS: LAUNDRY DJENGAH)," *JATI J. Mhs. Tek. Inform.*, vol. 8, no. 3, 2024, doi: <https://doi.org/10.36040/jati.v8i3.9906>.
- [3] M. Shahbandi, "Unleashing Competitive Advantage through Digital Marketing: Expanding Customer Access and Social Platform Development," *Int. Bus. Econ. Stud.*, vol. 6, no. 2, 2024, doi: 10.22158/ibes.v6n2p52.
- [4] H. Yenigun, N. Yevtushenko, and A. R. Cavalli, "Guest Editorial: Special issue on Testing Software and Systems," *Softw. Qual. J.*, vol. 27, no. 2, pp. 497–499, 2019, doi: 10.1007/s11219-019-09447-4.
- [5] M. H. S. Fedianto, F. P. Aditiawan, and M. M. A. Haromainy, "Pengujian Sistem Jaringan Dokumentasi Dan Informasi Menggunakan Black Box Testing Dan White Box Testing," *J. Publ. Sist. Inf. Dan Manaj. Bisnis*, vol. 3, no. 1, pp. 213–221, 2023, doi: 10.55606/jupsim.v3i1.2447.
- [6] Zulkarnaini, A. Firdhayanti, T. Taufik, and B. Bachry, "User Acceptance Testing through Blackbox Evaluation for Corn Distribution Information System," *Bit-Tech*, vol. 6, no. 2, pp. 208–215, 2023, doi: 10.32877/bt.v6i2.1065.
- [7] I. M. S. Ardana, "PENGUJIAN SOFTWARE MENGGUNAKAN METODE BOUNDARY VALUE ANALYSIS DAN DECISION TABLE TESTING," vol. 14, pp. 40–47, 2019.
- [8] X. Guo, H. Okamura, and T. Dohi, "Optimal test case generation for boundary value analysis," *Softw. Qual. J.*, vol. 32, no. 2, pp. 543–566, 2024, doi: 10.1007/s11219-023-09659-9.
- [9] N. Schick, "Exploring the Boundaries: Thorough Software Testing for Safety-Critical Driving Scenarios Based on Kinematics in the Context of Autonomous Driving," 2023, *arXiv*. doi: 10.48550/arXiv.2306.01898.
- [10] C. E. E. S. Putri and A. Susanto, "Feasibility Analysis of Bengkel Koding Website Using Black

- Box Testing and Boundary Value Analysis,” *Sinkron*, vol. 8, no. 2, pp. 764–776, 2024, doi: 10.33395/sinkron.v8i2.13589.
- [11] F. Dobslaw, F. G. de O. Neto, and R. Feldt, “Boundary Value Exploration for Software Analysis,” in *2020 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, IEEE, 2020, pp. 346–353. doi: 10.1109/ICSTW50294.2020.00062.
- [12] A. Ijudin and A. Saifudin, “Penguujian Black Box pada Aplikasi Berita Online dengan Menggunakan Metode Boundary Value Analysis,” *J. Inform. Univ. Pamulang*, vol. 5, no. 1, p. 8, 2020, doi: 10.32493/informatika.v5i1.3717.
- [13] C. D. Megawati, N. D. Miwa, and B. R. P. D. Palevi, “Black Box Testing of the ‘Hybrid Engine’ Application Using Boundary Value Analysis Technique,” *Sinkron*, vol. 8, no. 2, pp. 923–938, 2023, doi: 10.33395/sinkron.v8i2.12278.
- [14] F. K. Kartono *et al.*, “Penguujian Black Box Testing Pada Sistem Website Osha Snack: Pendekatan Teknik Boundary Value Analysis,” *J. KRIDATAMA SAINS DAN Teknol.*, vol. 6, no. 02, pp. 754–766, 2024, doi: 10.53863/kst.v6i02.1407.
- [15] X. Guo, H. Okamura, and T. Dohi, “Optimal test case generation for boundary value analysis,” *Softw. Qual. J.*, vol. 32, no. 2, pp. 543–566, 2024, doi: 10.1007/s11219-023-09659-9.
- [16] S. R. Yulistina, T. Nurmalia, R. M. A. T. Supriawan, S. H. I. Juni, and A. Saifudin, “Penerapan Teknik Boundary Value Analysis untuk Penguujian Aplikasi Penjualan Menggunakan Metode Black Box Testing,” *J. Inform. Univ. Pamulang*, vol. 5, no. 2, p. 129, Jun. 2020, doi: 10.32493/informatika.v5i2.5366.
- [17] A. Jailani and M. A. Yaqin, “Penguujian Aplikasi Sistem Informasi Akademik menggunakan Metode Blackbox dengan Teknik Boundary Value Analysis,” *JACIS J. Automation Comput. Inf. Syst.*, vol. 4, no. 2, pp. 60–66, 2024, doi: <https://doi.org/10.47134/jacis.v4i2.78>.
- [18] M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, “PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS,” *JITTER J. Ilm. Teknol. Inf. Terap.*, vol. 1, no. 3, pp. 31–36, 2015.
- [19] B. A. Maulana, E. Mawarni, M. Y. Hidayattuloh, V. Suryawijaya, and A. Saifudin, “Penguujian Black Box pada Sistem Informasi Barang Berbasis Web Menggunakan Metode Boundary Value Analysis,” *OKTAL J. Ilmu Komput. Dan Sci.*, vol. 2, no. 6, pp. 1747–1753, 2023.
- [20] S. D. S. Saian, N. L. Kakihary, and T. Wahyono, “PENGUJIAN CONTENT MANAGEMENT SYSTEM (CMS) SEKOLAHKU MENGGUNAKAN METODE BLACK BOX TESTING DENGAN TEKNIK BOUNDARY VALUE ANALYSIS,” *IT-Explore J. Penerapan Teknol. Inf. Dan Komun.*, vol. 1, no. 2, pp. 100–113, 2022, doi: 10.24246/itexplore.v1i2.2022.pp100-113.