



PENGUJIAN BLACK BOX PADA APLIKASI E-COMMERCE OPENCART DENGAN METODE EQUIVALENCE PARTITIONING DAN BOUNDARY VALUE ANALYSIS

Fairus In'am Pratama ¹⁾, Eka Mira Novita Subroto ²⁾, Radina Mutia Haira ³⁾, Muhammad Ainul Yaqin⁴⁾

¹ Program Studi Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang

² Program Studi Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang

³ Program Studi Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang

⁴ Program Studi Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang

email: ¹ 210605110095@student.uin-malang.ac.id, ² 210605110056@student.uin-malang.ac.id,

³ 210605110016@student.uin-malang.ac.id, ⁴ yaqinov@ti.uin-malang.ac.id

ARTICLE INFO

Article History:

Received : 19 May 2023

Accepted : 11 June 2023

Published : 30 June 2023

Keywords:

Black Box Testing

Equivalence Partitioning

Boundary Value Analysis

OpenCart

Checkout System

IEEE style in citing this article:

F. I. Pratama, E. M. N. Subroto, R. M. Haira, and M. A. Yaqin, "Pengujian Black Box pada Aplikasi E-Commerce OpenCart dengan Metode Equivalence Partitioning dan Boundary Value Analysis", *Jurnal.ilmiah.informatika*, vol. 8, no. 1, pp. 54-64, Jun. 2023.

Corresponding Author:

Muhammad Ainul Yaqin

Universitas Islam Negeri

Maulana Malik Ibrahim

Malang

ABSTRACT

E-commerce applications have become a significant trend in modern trade. One popular open-source e-commerce application is OpenCart. Testing the application is paramount to ensure that OpenCart functions appropriately and meets the required quality and performance. This study aims to conduct Black Box testing on the OpenCart e-commerce platform using the Equivalence Partitioning (EP) and Boundary Value Analysis (BVA) methods. The testing is carried out to ensure the application's functionality directly relates to user satisfaction and trust. EP divides input data into partitions based on valid and invalid values, while BVA tests input values based on specific boundaries. The focus of the testing is on the checkout feature, which is an essential component of any e-commerce application. The testing flow follows the Software Testing Life Cycle (STLC) approach. The test results show a 57.14% level of conformity using EP and 100% using BVA. Based on these two results, the overall conformity level is 78.57%. The testing results indicate that the system performs well within the tested boundary values but may require improvements in the combination of partition usage.

1. PENDAHULUAN

Aplikasi belanja *daring* membawa banyak perubahan pada bagaimana masyarakat berbelanja. Hal tersebut membuatnya menjadi bagian penting pada bisnis modern. Contoh aplikasi jual-beli *daring* yang cukup populer dan banyak digunakan oleh para pedagang adalah OpenCart. OpenCart merupakan aplikasi *e-commerce open source* berlisensi GNU GPL (*General Public Lisence*) yang dapat digunakan secara gratis. Para pedagang dapat membuat dan mengolah aplikasi secara fleksibel sesuai kebutuhan mereka [1]. Pedagang juga tidak perlu mahir dalam memahami atau menerapkan bahasa pemrograman dalam penggunaannya. Selain itu, SEO dan antarmuka yang mudah digunakan, serta banyaknya fitur yang dimiliki membuatnya menjadi pilihan populer untuk berbagai kategori bisnis [2].

Pengujian pada suatu aplikasi sangat penting dilakukan untuk memastikan kualitas dan kinerja dari aplikasi yang diuji. Aplikasi yang berfungsi dengan baik berguna dalam meningkatkan kepuasan serta kepercayaan pengguna. Selain itu, proses pengujian juga berguna untuk mengetahui kekurangan serta kesalahan pada aplikasi yang dibuktikan dari studi literatur berikut.

Pada pengujian aplikasi Seleksi Sales Terbaik ditemukan kesalahan dalam edit data di mana terjadi ketidaksesuaian antara notifikasi pesan yang menunjukkan berhasil disimpan dengan diikuti tidak adanya perubahan pada data [3]. Pada studi berikutnya menunjukkan bahwa pengujian perangkat lunak pada sistem informasi perpustakaan masih ada *form* yang hasil keluarannya belum sesuai dengan harapan dalam perancangan dikarenakan eror [4]. Kemudian, pada pengujian aplikasi Pengadaan Bahan Baku Batik ditemukan kesalahan yang dibuktikan dengan nilai DRE sebesar 0.45

atau 45% *test case* yang dibangun tidak lolos uji [5].

Berdasarkan studi literatur di atas, memastikan aplikasi dapat berfungsi dengan baik menjadi faktor penting dalam pengujian perangkat lunak yang harus diperhatikan bagi setiap pelaku bisnis. Salah satu teknik yang populer dan sering digunakan adalah teknik *Black Box* [6][7]. Teknik *Black Box* berguna untuk menguji fungsionalitas suatu sistem perangkat lunak dari perspektif pengguna tanpa perlu memahami bagaimana internal dari sistem tersebut bekerja [8]. *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) merupakan dua macam teknik dari pengujian *Black Box* yang digunakan untuk menguji ketepatan antara masukan dan keluaran dari sistem [9].

EP adalah metode pengujian perangkat lunak dengan cara membagi data masukan berdasarkan partisi-partisi dengan nilai valid dan tidak valid. Prinsip pengujian dengan teknik EP berdasar pada spesifikasi dan persyaratan dari perangkat lunak. Adanya pendekatan ini dapat membantu mengurangi waktu pengujian karena jumlah kasus uji yang lebih kecil [10].

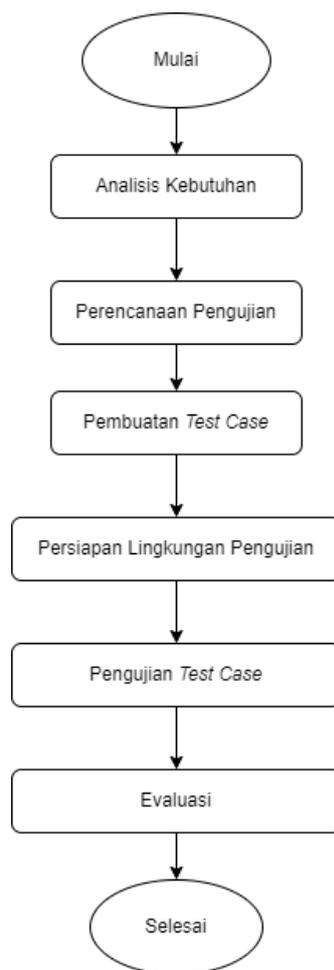
BVA adalah metode pengujian dengan menguji nilai batas dari masukan sistem. Metode BVA sering dipilih untuk menyempurnakan pengujian dari metode EP. Nilai batas uji masukan sistem, terdiri dari nilai di bawah batas minimum (BLB), nilai minimum (LB), nilai di atas batas minimum (ALB), nilai di bawah batas minimum (BUB), nilai maksimum (UP), dan nilai di atas batas maksimum (TUB). Menurut penelitian yang pernah dilakukan, teknik ini memiliki peluang lebih tinggi untuk mendeteksi kesalahan [11]. Dengan begitu, penggunaan metode EP dan BVA dapat memastikan apakah sistem berjalan sesuai ekspektasi atau memiliki kesalahan.

Penelitian ini bertujuan untuk melakukan pengujian secara *Black Box* pada aplikasi *e-commerce* OpenCart menggunakan metode *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA). Fokus utama dari pengujian terletak pada fitur *checkout* yang merupakan komponen penting pada setiap aplikasi *e-commerce*. Fitur *checkout* memiliki peranan penting dalam membantu dan mengarahkan pengguna melewati fitur pembelian. Penggabungan kedua metode

ini, diharapkan dapat secara efektif menguji OpenCart dan memastikan bahwa aplikasi tersebut berjalan dan berfungsi dengan baik.

2. METODE PENELITIAN

Dalam pengujian ini, peneliti menggunakan STLC (*Software Testing Life Cycle*) sebagai alur pengujian perangkat lunak. Alur dari STLC digambarkan pada Gambar 1 [12].

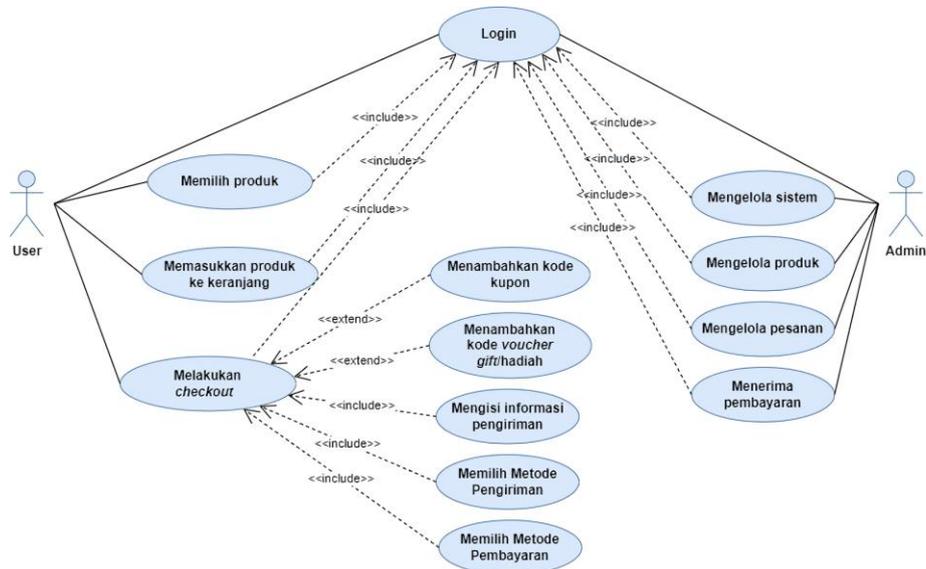


Gambar 1. Diagram Alir *Software Testing Life Cycle*

2.1 Analisis Kebutuhan

Tahapan pertama pengujian adalah menganalisis kebutuhan yang diperlukan, seperti perangkat lunak serta fitur yang diuji, *use case diagram*, dan persyaratan juga spesifikasi sistem.

Pada penelitian ini, perangkat lunak yang diuji adalah aplikasi OpenCart versi 4.0.1.1. Berikut *use case diagram* aplikasi OpenCart yang dapat dilihat pada Gambar 2.



Gambar 2. Use Case Diagram Aplikasi OpenCart

Dari Gambar 2, peneliti akan menguji salah satu *case*, yaitu aktivitas *checkout* atau fitur *checkout* dimulai dari menambah kode kupon dan *gift/hadiah*, mengisi informasi pembayaran, memilih metode

pengiriman, hingga memilih metode pembayaran. Antarmuka dari fitur *checkout* dapat dilihat pada Gambar 3. dan Gambar 4.

What would you like to do next?
Choose if you have a discount code or reward points you want to use or would like to estimate your delivery cost.

Use Coupon Code

Enter your coupon here

Apply Coupon

Use Gift Certificate

Continue Shopping

Checkout

Gambar 3. Antarmuka Fitur *Checkout* Menambah Kode Kupon dan Hadiah

Shopping Cart Checkout

Checkout

Shipping Address

I want to use an existing address
 I want to use a new address

* First Name: First Name
* Last Name: Last Name
Company: Company
Address 1: Address 1
Address 2: Address 2
City: City
Post Code: Post Code
Country: Please Select ---
Region / State: None ---

Flat Rate: Please Select ---

Payment Method: Please Select ---

Add Comments About Your Order

Product Name	Total
Zx iPhone	\$202.00
Sub-Total	\$202.00
Total	\$202.00

Continue Confirm Order

Gambar 4. Antarmuka Fitur *Checkout* Halaman Utama

Analisis kebutuhan lainnya adalah spesifikasi dan persyaratan sistem dari fitur yang diuji. Kedua hal tersebut berguna sebagai acuan atau aturan dalam

pembuatan *test case* di tahap selanjutnya. Spesifikasi dan persyaratan sistem fitur *checkout* digambarkan pada Tabel 1. dan Tabel 2.

Tabel 1. Persyaratan Sistem Pada Fitur *Checkout*

Kode	Bagian yang diuji	Kriteria
K01	Penambahan kode kupon	Menambahkan kode kupon yang sesuai dengan yang telah ditambahkan pada sistem
K02	Penambahan <i>voucher gift</i> /hadiah	Menambahkan <i>voucher gift</i> /hadiah yang telah ditambahkan pada sistem
K03	Informasi Pengiriman ⊙ Input Data Wajib: Kolom <i>first name</i> , <i>last name</i> , <i>address 1</i> , <i>city</i> , <i>country</i> , dan <i>region/state</i>	Mengisi <i>input</i> data yang wajib diisi sesuai dengan formatnya
	⊙ Input Data Optional: Kolom <i>company</i> , <i>address 2</i> , dan <i>postcode</i>	
K04	Metode Pengiriman	Memilih hanya satu metode pengiriman
K05	Metode Pembayaran	Memilih hanya satu metode pembayaran

Tabel 2. Spesifikasi Sistem Pada Fitur *Checkout*

Kode	Bagian yang diuji	Kriteria
K06	Penambahan Kode Kupon	Menambahkan kode kupon dengan syarat batas minimal harga belanja sudah terpenuhi (dalam penelitian ini nilai minimal harga belanja \$200)
K07	Informasi Pengiriman: Kolom <i>First Name</i>	Berisikan antara 1 sampai 32 karakter
K08	Informasi Pengiriman: Kolom <i>Last Name</i>	Berisikan antara 1 sampai 32 karakter
K09	Informasi Pengiriman: Kolom <i>Address 1</i>	Berisikan antara 3 sampai 128 karakter
K10	Informasi Pengiriman: Kolom <i>City</i>	Berisikan antara 2 sampai 128 karakter

2.2 Perencanaan Pengujian

Pada pengujian aplikasi OpenCart, peneliti menggunakan teknik pengujian *Black Box* dengan metode *Equivalence Partitioning* (EP) dan *Boundary Value*

Analysis (BVA) untuk menguji fungsionalitas dari fitur *checkout*.

Pengujian dengan metode EP akan membagi masukan menjadi partisi-partisi. Partisi-partisi tersebut kemudian dikelompokkan menjadi nilai valid (*true*)

dan invalid (*false*). Pada metode EP, penentuan partisi disesuaikan dari persyaratan sistem pada Tabel 1.

Metode BVA akan menguji bagian spesifikasi sistem berdasarkan batas nilai uji yang digambarkan pada Tabel 2. Terdapat dua variasi masukan yang membuat penentuan batas nilai uji berbeda yaitu:

- a) Pada penambahan kode kupon dan *gift*, *input* berupa suatu nilai $> n$ sehingga batas nilai ujinya adalah $(n-1)$ dan $(n+1)$ atau BLB dan AUB.
- b) Pada pengujian kolom *First Name*, *Last Name*, *Address 1*, dan *City*, *input* berupa *range* nilai x dan y sehingga batas nilai uji adalah $(x-1)$, x , y dan $(y+1)$ atau BLB, LB, UB, dan AUB.

2.3 Pembuatan Test Case

Tahap ini adalah tahap pembuatan kasus tes. Kasus tes dibuat berdasarkan perencanaan pengujian yang telah dilakukan di tahap sebelumnya. Kemudian, hasil dari pembuatan tes selanjutnya akan dijelaskan pada Tabel 3. dan Tabel 4. beserta hasil pengujianya.

2.4 Persiapan Lingkungan Pengujian

Perangkat yang digunakan untuk menguji aplikasi OpenCart versi 4.0.1.1

adalah perangkat keras berbasis Windows 11 dan Laragon versi 6.0 sebagai *localhost*. Pada bagian admin ditambahkan kode kupon dan hadiah/*gift* untuk digunakan dalam pengujian.

2.5 Pengujian Test Case

Pada tahap ini, peneliti akan menguji kasus uji yang telah dibuat. Kemudian, apabila hasil pengujian sesuai dengan hasil yang diharapkan maka tes berhasil, jika tidak maka hasil tidak sesuai.

2.6 Evaluasi

Tahap terakhir adalah evaluasi dengan menganalisis dan menyimpulkan hasil *test case* yang telah diuji sehingga dapat diketahui apakah aplikasi OpenCart sudah berfungsi dengan baik atau masih terdapat masalah yang ditemukan.

3. HASIL DAN PEMBAHASAN

Dari dari kegiatan yang dilakukan sebelumnya, yaitu pada tahap pembuatan *test case* didapatkan hasil 14 tes untuk metode *Equivalence Partitioning* dan 14 tes untuk metode *Boundary Value Analysis*. *Test case* yang telah dibuat tersebut kemudian akan diuji. *Test case* dan hasil pengujianya ditunjukkan pada Tabel 3. dan Tabel 4.

Tabel 3. Test Case dan Hasil Pengujian dengan Metode *Equivalence Partitioning*

No. Data Uji	Deskripsi Pengujian	Nilai	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan	
					Sesuai	Tidak Sesuai
D01	Menambahkan kode kupon yang sesuai	Valid	Sistem akan menerima <i>kode</i>	Kode berhasil ditambahkan	✓	
D02	Menambahkan kode kupon yang tidak sesuai	Invalid	Sistem akan memberikan pesan error	Kode gagal ditambahkan	✓	

D03	Menambahkan <i>voucher gift</i> /hadiah yang sesuai	Valid	Sistem akan menerima <i>voucher gift</i> /hadiah	<i>Gift</i> berhasil ditambahkan	✓
D04	Menambahkan <i>voucher gift</i> /hadiah yang tidak sesuai	Invalid	Sistem akan memberikan pesan error	<i>Gift</i> gagal ditambahkan	✓
D05	Mengisi semua <i>input</i> data	Valid	Sistem menyimpan informasi	Data masukan berhasil disimpan	✓
D06	Mengisi semua <i>input</i> data wajib dan mengosongkan semua <i>input</i> data yang opsional	Valid	Sistem menyimpan informasi	Data masukan berhasil disimpan	✓
D07	Mengisi semua <i>input</i> data <i>optional</i> dan mengosongkan semua <i>input</i> data yang wajib	Invalid	Sistem akan memberikan pesan error yang mengharuskan mengisi <i>first name, last name, address 1, city, country, dan region/state</i> .	Sistem menunjukkan penolakan dan mengharuskan pengisian kolom <i>first name, last name, address 1, city, dan country</i> , tetapi kolom <i>region/state</i> tidak mengalami penolakan.	✓
D08	Mengosongkan semua <i>input</i> data	Invalid	Sistem akan memberikan pesan error yang mengharuskan mengisi <i>first name, last name, address 1, city, country, dan region/state</i> .	Sistem menunjukkan penolakan dan mengharuskan mengisi kolom <i>first name, last name, address 1, city, dan country</i> , tetapi kolom <i>region/state</i> tidak mengalami penolakan.	✓

D09	Memilih metode pengiriman	Valid	Sistem menyimpan informasi	Sistem menyimpan informasi	✓
D10	Memilih metode pengiriman, lalu mengosongkannya kembali	Invalid	Sistem memberikan pesan eror untuk mengisi metode pengiriman	Sistem tetap menyimpan informasi dan menerima permintaan	✓
D11	Tidak memilih metode pengiriman	Invalid	Sistem memberikan pesan eror untuk mengisi metode pengiriman	Sistem tetap menyimpan informasi dan menerima permintaan	✓
D12	Memilih hanya satu metode pembayaran	Valid	Sistem menyimpan informasi	Sistem menyimpan informasi	✓
D13	Memilih metode pembayaran, lalu mengosongkannya kembali	Invalid	Sistem memberikan pesan eror untuk mengisi metode pembayaran	Sistem menyimpan informasi dan permintaan telah diterima	✓
D14	Tidak memilih metode pembayaran	Invalid	Sistem memberikan pesan eror untuk mengisi metode pembayaran	Sistem tetap menyimpan informasi dan menerima permintaan	✓

Tabel 4. *Test Case* dan Hasil Pengujian dengan Metode *Boundary Value Analysis*

No. Data Uji	Kolom	Deskripsi Pengujian	Nilai Uji	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan	
						Sesuai	Tidak Sesuai
D15	Kode Kupon	Harga pembelian produk \$199	BLB	Sistem akan menolak <i>kode</i> kupon	Kode tidak berhasil ditambahkan	✓	

D16	Kode Kupon	Harga pembelian produk \$201	AUB	Sistem akan menerima <i>kode</i> kupon	Kode berhasil ditambahkan	✓
D17	<i>First Name</i> dan <i>Last Name</i>	Memasukkan 0 karakter	BLB	Sistem akan memberikan pesan error	Terdapat pesan error	✓
D18	<i>First Name</i> dan <i>Last Name</i>	Memasukkan 1 karakter	LB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓
D19	<i>First Name</i> dan <i>Last Name</i>	Memasukkan 32 karakter	UB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓
D20	<i>First Name</i> dan <i>Last Name</i>	Memasukkan 33 karakter	AUB	Sistem akan memberikan pesan error	Terdapat pesan error	✓
D21	<i>Address 1</i>	Memasukkan 2 karakter	BLB	Sistem akan memberikan pesan error	Terdapat pesan error	✓
D22	<i>Address 1</i>	Memasukkan 3 karakter	LB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓
D23	<i>Address 1</i>	Memasukkan 128 karakter	UB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓
D24	<i>Address 1</i>	Memasukkan 129 karakter	AUB	Sistem akan memberikan pesan error	Terdapat pesan error	✓
D25	<i>City</i>	Memasukkan 1 karakter	BLB	Sistem akan memberikan pesan error	Terdapat pesan error	✓
D26	<i>City</i>	Memasukkan 2 karakter	LB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓
D27	<i>City</i>	Memasukkan 128 karakter	UB	Sistem akan menerima data	Data dapat diterima oleh sistem	✓

D28	City	Memasukkan 129 karakter	AUB Sistem akan memberikan pesan eror	Terdapat pesan eror	✓
-----	------	-------------------------	---------------------------------------	---------------------	---

Berdasarkan hasil pengujian menggunakan metode *Equivalence Partitioning* pada Tabel 3, dari total 14 tes yang dilakukan, sistem menunjukkan kesesuaian sebanyak 57,14% dengan total delapan tes dinyatakan sesuai, dan enam tes dinyatakan tidak sesuai.

Selama pengujian didapati kolom yang kurang konsisten dalam memberikan keterangan yaitu kolom *region/state* yang keluar notifikasi eror ketika *dropdown* dari kolomnya ditekan atau dipilih dan kemudian dikembalikan lagi pada mode *select*.

Pada kolom pembayaran dan kolom metode pengiriman, juga menunjukkan ketidakkonsistenan. Secara *default*, sistem mewajibkan pengisian kedua kolom tersebut, tetapi setelah dilakukan pengisian, dan mengosongkan kembali, sistem tetap menerima masukan tanpa ada larangan atau notifikasi eror.

Kemudian, pada Tabel 4. yang menunjukkan hasil pengujian menggunakan metode *Boundary Value Analysis* diperoleh hasil bahwa dari total 14 tes yang dilakukan tidak ditemukan kesalahan pada sistem yang diuji atau menunjukkan kesesuaian sebanyak 100%.

4. UCAPAN TERIMA KASIH

Segala puji serta syukur hanya milik Allah SWT yang telah memberikan karunia dan rahmat-Nya karena dengan kehendak-Nya penelitian ini bisa diselesaikan. Peneliti juga mengucapkan banyak terima kasih kepada Bapak Muhammad Ainul Yaqin, M. Kom selaku dosen pembimbing yang senantiasa membantu memberi arahan serta bimbingannya terhadap penelitian ini sehingga bisa diselesaikan tepat waktu.

Tak lupa juga peneliti ucapkan terima kasih kepada JIMI selaku media penyelenggara jurnal yang sudah menerima jurnal ini.

5. KESIMPULAN

Pengujian fitur *checkout* pada aplikasi OpenCart menggunakan metode *Equivalence Partitioning* dan *Boundary Value Analysis* menunjukkan tingkat kesesuaian sebanyak 78,57%. Hasil pengujian menunjukkan bahwa sistem dapat berjalan dengan baik pada nilai-nilai batas yang diuji menggunakan metode *Boundary Value Analysis* dengan tingkat kesesuaian 100%. Namun, masih perlu ditingkatkan dalam mengatasi kombinasi penggunaan antar partisi yang tingkat kesesuaiannya hanya 57,14% pada pengujian dengan metode *Equivalence Partitioning*. Oleh karena itu, peneliti menyarankan pengembangan lebih lanjut pada sistem untuk meningkatkan kemampuan sistem dalam mengatasi kombinasi penggunaan antar partisi.

6. REFERENSI

- [1] Z.-J. Lee *et al.*, "Design an Online Shopping Store Based on Opencart," *Artificial Intelligence Evolution*, pp. 1–7, Mar. 2020, doi: 10.37256/AIE.112020262.
- [2] N. Yuliani, A. Yuniarto, and M. Marnis, "Implementasi Penggunaan CMS OpenCart Dalam Upaya Toko Online Naik Kelas," *Tekinfor: Jurnal Bidang Teknik Industri dan Teknik Informatika*, vol. 23, no. 2, pp. 123–138, Oct. 2022, Accessed: Apr. 14, 2023. [Online]. Available: <https://journals.upi->

- yai.ac.id/index.php/TEKINFO/article/view/2604
- [3] F. C. Ningrum, D. Suherman, S. Aryanti, H. A. Prasetya, and A. Saifudin, "Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions," *Jurnal Informatika Universitas Pamulang*, vol. 4, no. 4, pp. 125–130, Dec. 2019, doi: 10.32493/INFORMATIKA.V4I4.3782.
- [4] M. Nur Ichsanudin, M. Yusuf, S. Jurusan Rekayasa Sistem Komputer, J. Teknik Industri, I. AKPRIND Yogyakarta, and R. Artikel, "Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula," *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 1, no. 2, pp. 1–8, May 2022, doi: 10.55123/STORAGE.V1I2.270.
- [5] I. Widia, I. D. M. Widia, S. Rosalin, S. R. Asriningtias, and E. Sonalitha, "Black Box Testing Menggunakan Boundary Value Analysis dan Equivalence Partitioning pada Aplikasi Pengadaan Bahan Baku Batik dengan Pendekatan Use Case," *JIMP (Jurnal Informatika Merdeka Pasuruan)*, vol. 6, no. 1, Mar. 2022, doi: 10.37438/jimp.v6i1.300.
- [6] S. R. Yulistina, T. Nurmala, R. M. A. T. Supriawan, S. H. I. Juni, and A. Saifudin, "Penerapan Teknik Boundary Value Analysis untuk Pengujian Aplikasi Penjualan Menggunakan Metode Black Box Testing," *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 2, pp. 129–135, Jun. 2020, doi: 10.32493/INFORMATIKA.V5I2.5366.
- [7] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and Y. Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis," *Jurnal Informatika Universitas Pamulang*, vol. 4, no. 4, pp. 143–148, Dec. 2019, doi: 10.32493/INFORMATIKA.V4I4.3841.
- [8] A. Rifqi, Y. Arfani, P. Kasih, and D. P. Pamungkas, "Pengujian Aplikasi Presensi dengan Black box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, vol. 6, no. 1, pp. 338–343, Aug. 2022, doi: 10.29407/INOTEK.V6I1.2539.
- [9] G. W. Sasmito and M. Nishom, "Testing the Population Administration Website Application Using the Black Box Testing Boundary Value Analysis Method," *2020 IEEE Conference on Open Systems, ICOS 2020*, pp. 48–52, Nov. 2020, doi: 10.1109/ICOS50156.2020.9293645.
- [10] D. Widhyaestoeti *et al.*, "Black Box Testing Equivalence Partitions For Front-End Testing On Academic Systems Sitoda," *Jurnal Ilmiah Teknologi Infomasi Terapan*, vol. 7, no. 3, pp. 211–216, Aug. 2021, doi: 10.33197/JITTER.VOL7.ISS3.2021.626.
- [11] S. Rani, "A Comparative Study of Different Software Testing Techniques," *Journal of Advances in Shell Programming*, vol. 5, no. 1, pp. 1–8, Jun. 2018, Accessed: Apr. 14, 2023. [Online]. Available: <https://computers.stmjournals.com/index.php?journal=JoASP&page=article&op=view&path%5B%5D=1559>
- [12] M. Divyani, S. Taley, and B. Pathak, "Comprehensive Study of Software Testing Techniques and Strategies: A Review," *International Journal of Engineering Research & Technology*, vol. 9, no. 8, Sep. 2020, doi: 10.17577/IJERTV9IS080373.