

Product Automation Testing pada Kalcare.com Memanfaatkan Teknik Boundary Value Analysis dan Equivalence Partitioning

1st Hans Prabowo
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
hansprabowo@student.telkomunive
rsity.ac.id

2nd Rahmat Yasirandi
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
batanganhitam@telkomuniversity.a
c.id

3rd Bernadus Ricardo Saputra
Divisi IT Digital
PT Alto Network
Jakarta, Indonesia
bricardosaputra@gmail.com

Abstrak-Kalcare.com merupakan *website e-commerce* milik PT. Karsa Hasta Dinamika di bawah naungan PT. Karsa lintas buwana yang merupakan salah satu anak perusahaan dari Kalbe Group. *Website* ini digunakan untuk menjual produk-produk dari perusahaan Kalbe sendiri maupun perusahaan yang bekerja sama dengan Kalbe Group. Pada pengembangan *website* terdapat fase dimana perlu dilakukan pengujian untuk menentukan kelayakan *website*. Pengujian dapat dilakukan secara otomatis (*automation testing*) untuk mengurangi *human error* dan meningkatkan efisiensi pengujian tanpa memakan durasi yang cukup panjang. Terdapat berbagai macam kategori pengujian otomatis salah satunya *black box testing* yang dapat menguji *website* tanpa perlu melihat kode program secara detail. Pada penelitian ini dilakukan pengujian otomatis pada menu *login website* Kalcare.com dengan metode *black box testing*. Pengujian dilakukan dengan teknik *Boundary Value Analysis (BVA)* yang dapat menerima respon dari Kalcare.com ketika menerima input panjang yang *valid* serta teknik *Equivalence Partitioning (EP)* untuk melihat respon Kalcare.com saat input diterima dengan tipe data yang *valid* dan sebaliknya. Berdasarkan hasil pengujian diketahui bahwa kualitas *website* telah sesuai dengan fungsi dan didapatkan kesuksesan sebesar 100% pada fungsi *login website* Kalcare.com.

Kata Kunci- kalcare.com, pengujian otomatis, *black box testing*, *boundary value analysis*, *equivalence partitioning*.

Abstract-Kalcare.com is an *e-commerce website* owned by PT. Karsa Hasta Dinamika under the auspices of PT. Karsa Lintas Buwana which is a subsidiary of the Kalbe Group. This *website* is used to sell products from Kalbe's own companies or companies that work with Kalbe Group. In *website* development there is a phase where testing is necessary to determine the feasibility of the *website*. Testing can be done automatically (*automation testing*) to reduce *human error* and improve test efficiency without taking a long time. There are various categories of automated testing, one of which is *black box testing* which can test *websites* without the

need to look at the program code in detail. In this study, automatic testing was carried out on the Kalcare.com *website* login menu with the *black box testing* method. The test is carried out using the *Boundary Value Analysis (BVA)* technique which can receive a response from Kalcare.com when it receives a valid input length and the *Equivalent Partitioning (EP)* technique to see the response of Kalcare.com when the input is received with a valid data type and vice versa. Based on the test results, it is known that the quality of the *website* is in accordance with the function and 100% success is obtained on the Kalcare.com *website* login function.

Keywords- kalcare.com, *automation testing*, *black box testing*, *boundary value analysis*, *equivalence partitioning*.

I. PENDAHULUAN

A. Latar Belakang

Pada dasarnya suatu *software* memerlukan pengujian untuk mengetahui kelayakan *website* dalam penggunaan komersial [1]. Terdapat dua cara yang dapat digunakan untuk pengujian *software* termasuk *website* yaitu pengujian manual dan pengujian otomatis. Dalam pengujian manual waktu yang dibutuhkan relatif lebih panjang sehingga pengujian otomatis dijadikan solusi untuk memberikan efisiensi waktu pengerjaan [2]. Pengujian otomatis dapat menjalankan perintah yang telah dimasukkan secara otomatis [3]. Terdapat tiga kategori teknik pengujian perangkat lunak, pertama pengujian berdasarkan analisis tes dasar dokumentasi, termasuk aspek fungsional dan non-fungsional yang disebut *Black Box Testing*. Kedua teknik pengujian berdasarkan uji kasus langsung dari struktur komponen atau sistem yang dikenal sebagai *structure-based* atau *White Box Testing*. Ketiga teknik pengujian berdasarkan uji kasus dari pengalaman *tester* sistem serupa dan pengalaman umum dari pengujian yang dikenal sebagai *Experience-based Techniques* [4]. *Black Box Testing* memiliki kelebihan dibanding dengan kedua

teknik lainnya yaitu dapat menentukan spesifikasi program di awal, tidak perlu melihat kode program secara detail, dan dapat digunakan untuk menilai konsistensi program.

Kalcare adalah *website e-commerce* yang dibuat oleh PT. Karsa Hasta Dinamika di bawahaungan PT. Karsa lintas buwana yang merupakan salah satu anak perusahaan dari Kalbe Group¹. *Website* ini digunakan untuk menjual produk-produk dari perusahaan Kalbe sendiri maupun perusahaan yang bekerja sama dengan Kalbe Group. Dalam upaya pengembangan *website* perlu dilakukan pengujian kelayakan yang dapat dilakukan dengan *Black Box Testing*[5]. Terdapat beberapa teknik pengujian *Black Box Testing* salah satunya *Boundary Value Analysis* dan *Equivalence Partitioning* [6]. Teknik *Boundary Value Analysis* membahas *Black Box Testing* dalam keseluruhan aspek menu dan modul, sehingga dapat diketahui sisi kekeliruannya untuk dapat dilakukan pengembangan dan *maintenance*. *Equivalence Partitioning* membahas mengenai *testing* dalam aspek validasi input yang dilihat dari *valid class*, pengamatan isi inputan, dan akurasi inputan [7].

Pada penelitian ini dilakukan pengujian otomatis pada *website* Kalcare.com dengan pengujian *Black Box Testing* menggunakan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. Dengan adanya penelitian ini pengujian secara manual dengan durasi panjang dan banyaknya kendala seperti *human error* dan lainnya dapat diatasi dengan *automation testing*. Dengan demikian pengujian dapat dilakukan dengan cepat, akurat dan efisien.

B. Topik dan Batasan

Berdasarkan latar belakang penelitian diketahui Kalcare sedang dalam tahap pengembangan sehingga perlu dilakukan pengujian pada *website*. Pada pengujian yang dilakukan waktu yang panjang dan akurasi yang rendah memberikan kendala pada pengujian secara manual untuk itu dilakukan pengujian secara otomatis (*automation testing*) dengan *Black Box Testing* tanpa perlu melihat kode program secara detail. Teknik yang digunakan pada pengujian *Black Box Testing* adalah teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. *Boundary Value Analysis* digunakan untuk melakukan pengujian batas atas dan batas bawah nilai yang diinput pada sistem, sedangkan *Equivalence Partitioning* digunakan untuk pengujian berdasarkan input dari setiap menu yang termasuk dalam menu *login*. Setiap menu input diuji dengan klasifikasi dan pengelompokan berdasarkan fungsinya [8]. *Scope* pada penelitian ini berfokus pada halaman *login*

menggunakan robot *framework* dan selenium *library* dengan pengerjaan dalam *software* Katalon Studio. Pengembangan kerangka pengujian hanya dilakukan pada aspek fungsional dan *user interface* tanpa meninjau API dan validasi ke database *website*.

C. Tujuan

Penelitian ini bertujuan untuk mengurangi tingkat resiko ketidaksesuaian data pada pengujian manual di halaman *login website* Kalcare dengan menerapkan *automation testing* dengan menerapkan pengujian otomatis menggunakan sistem *black box testing* dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. Pengujian dilakukan untuk mengetahui performa *website* Kalcare.com pada laman *login* serta mengetahui teknik terbaik yang dapat digunakan untuk pengujian otomatis sehingga menghasilkan hasil yang akurat.

¹ Kalcare | <https://www.Kalcare.com/>

II. TINJAUAN PUSTAKA

A. Kalcare.com

Kalcare.com merupakan situs *e-commerce* produk-produk nutrisi, perlengkapan ibu dan anak, makanan dan minuman, perawatan pribadi, vitamin dan suplemen, obat-obatan. Kalcare, bernaung di bawah perusahaan PT Karsa Lintas Buwana yang merupakan salah satu anak perusahaan dari Kalbe Group².

B. Automation Testing

Pengujian otomatis adalah eksekusi otomatis kasus uji tanpa intervensi manual, cukup dengan merekam dan memutar ulang kasus uji. Hal ini memungkinkan penguji / pengujian untuk mengulang pekerjaan mereka di setiap siklus tes. Pengujian otomatis dapat mengurangi upaya pengujian dan memastikan pelaksanaan pengujian yang efisien. Ini dapat diimplementasikan menggunakan alat uji otomatis yang saat ini digunakan seperti robot *framework*, katalon studio dan selenium [9].

C. Functional Testing

Tes fungsional adalah jenis tes yang bertujuan untuk menentukan apakah item tes atau contoh persyaratan fungsional terpenuhi, termasuk kemampuan untuk menentukan apakah suatu fungsi telah dilakukan sesuai dengan persyaratan yang ditentukan. Dapat dilakukan dengan menggunakan teknik desain pengujian berbasis spesifikasi dan berbasis struktur [10].

D. Katalon Studio

Katalon Studio adalah perangkat lunak pengujian otomatisasi yang dikembangkan oleh Katalon, Inc. Perangkat lunak ini dibangun

di atas *open-source automation frameworks*

² Kalbe | <https://www.kalbe.co.id/id/produk-dan-jasa/Kalcare>

Selenium, *Appium* dengan antarmuka IDE khusus untuk pengujian aplikasi web, API, seluler, dan desktop [11].

1. Test Case

Test case adalah serangkaian tindakan yang diambil sistem untuk menentukan apakah memenuhi persyaratan perangkat lunak dan berfungsi dengan baik. Tujuan dari *test case* adalah untuk menentukan apakah fitur sistem berfungsi seperti yang diharapkan dan untuk memastikan bahwa sistem memenuhi semua standar, pedoman, dan persyaratan pengguna yang relevan. *Test case* juga harus membantu mengungkapkan kesalahan dan kekurangan sistem. Studi sebelumnya juga menunjukkan bahwa teknik prioritas kasus uji dapat meningkatkan efektivitas *test case* untuk berbagai tujuan. Pendekatan prioritas ini memungkinkan untuk mengatur dan menjadwalkan kasus pengujian sehingga dapat menjalankan *test case* prioritas tinggi lebih cepat daripada *test case* prioritas rendah berdasarkan beberapa kriteria [12].

2. Test Suite

Untuk memfasilitasi pengujian perangkat lunak yang dikembangkan, *test suite* biasanya dikembangkan untuk pengujian perangkat lunak versi pertama dan digunakan kembali untuk menguji setiap versi perangkat lunak berikutnya [13]. Saat *test case* baru ditambahkan ke *test suite* untuk menguji persyaratan baru atau yang diubah dan menjaga *validitas test case*, ukuran *test suite* biasanya bertambah seiring dengan perubahan. *Test suite* itu sendiri adalah wadah yang berisi sejumlah tes yang dapat dijalankan oleh pengujian untuk melaporkan status uji coba [14].

C. Robot Framework

Robot framework adalah otomatisasi pengujian berbasis kata kunci untuk pengujian penerimaan ujung ke ujung dan pengembangan berbasis pengujian penerimaan (ATDD). Pustaka kata kunci ditulis dengan *python* atau *java*. Kerangka kerja ini gratis dan dapat digunakan untuk menjalankan pengujian sistem, pengujian fungsional, dan pengujian regresi. Perangkat lunak ini mendukung struktur pengujian tingkat tinggi dan menyediakan beberapa *editor* pengujian seperti *plugin RIDE* dan *Eclipse*, sehingga pengguna dapat dengan mudah mengelola dan menskalakan pengujian mereka [3]. Saat mulai menjalankan pengujian, dilakukan analisis data pengujian terlebih dahulu. Kemudian gunakan *keywords* yang

disediakan oleh pustaka pengujian untuk berinteraksi dengan sistem yang diuji. Beberapa pustaka pengujian memerlukan *driver* seperti *selenium library* dan REST (untuk pengujian API) untuk terhubung ke sistem yang sedang diuji. *Robot framework* bekerja dengan beberapa *browser* seperti *Internet Explorer*, *Firefox* dan *Chrome*. Hasilnya, dapat menerima laporan berformat HTML dan keluaran XML [15].

D. Keyword-Driven Testing

Pengujian berbasis kata kunci adalah pendekatan spesifikasi kasus uji yang biasa digunakan untuk membangun otomatisasi pengujian dan mengembangkan kerangka kerja pengujian otomatis. Pengujian berbasis kata kunci adalah cara untuk menggambarkan kasus pengujian menggunakan serangkaian kata kunci yang telah ditentukan sebelumnya. Kata kunci untuk pengujian ini terkadang disebut fungsi atau pemrograman, yaitu fungsi yang melakukan tindakan. Kata kunci kadang-kadang disebut sebagai kombinasi dari satu atau lebih tindakan pada objek uji. Gunakan kata kunci untuk menjelaskan prosedur pengujian dan membuat pengujian lebih mudah dipahami, dipelihara, dan otomatisasi. Manfaat pengujian kata kunci termasuk kemudahan penggunaan, kemudahan pemahaman bagi mereka yang tidak memiliki pengetahuan pemrograman, dan kemudahan pemeliharaan [16]. Tujuan utama pengujian berbasis kata kunci adalah untuk menyediakan kumpulan kata kunci dasar yang cukup komprehensif sehingga sebagian besar kasus pengujian dapat mencakup semua kata kunci ini. Pengujian otomatisasi mengharuskan semua kata kunci diimplementasikan dalam perangkat lunak [16].

E. Selenium Library

Selenium Library adalah salah satu *library* pengujian *eksternal Robot Framework* yang digunakan untuk menguji aplikasi berbasis *website*. *Keywords* umum yang didefinisikan di *library* ini digunakan untuk mengontrol elemen *input* pada aplikasi berbasis *website*. elemen *keywords Selenium library* identik dengan *keywords driven testing*, yang juga terdiri dari *keyword*, *locator*, dan parameter. Selain itu, *Selenium Library* menyediakan dokumentasi bagus yang menguraikan cara menggunakan dengan *keyword* [17].

F. Boundary Value Analysis

Boundary value analysis (BVA) berfokus pada batas-batas ruang input untuk mengidentifikasi pengujian [17]. Batas input ini memiliki nilai input di batas, tepat di batas bawah (di atas) dan tepat di batas atas (di bawah). Algoritma *black box testing* dengan teknik *boundary value analysis* adalah sebagai berikut:

1. Jika kondisi input berada pada *range* nilai x , maka perlu dibuat *test case* dengan menggunakan data sampel $x-1$, x , dan $x+1$.
2. Jika kondisi input menggunakan beberapa nilai, *test case* harus terdiri dari minimal -1, minimum, maksimum, dan maksimum +1 data sampel.
3. Ikuti langkah 1 dan 2 untuk proses keluaran. Jika data sudah memiliki batas input (misalnya, jika *array* diatur ke maksimum 10), kasus uji untuk batas ini akan dibuat [12].

G. Equivalence Partitioning

Teknik *Equivalence partitioning* dalam *black box testing* adalah dengan membagi input ke dalam kelas ekuivalensi [11]. Definisi lain dari *equivalence partitioning* adalah teknik yang didasarkan pada input dan *output* dari suatu komponen. Ini dibagi menjadi beberapa kelas sesuai dengan spesifikasi komponen dan diperlakukan sama oleh komponen. *test case* biasanya digunakan untuk menguji kesetaraan partisipasi. Seperti halnya BVA, hal penting dalam *test case partitioning equivalence* adalah *expected conclusion* yang berisi *success* atau *failed*-nya pengujian tersebut [18].

H. Penelitian Sebelumnya

Pada penelitian yang menggunakan *black box testing* untuk pengujian sistem dengan berjudul “Komparasi Dua Teknik *Black Box Testing: Equivalence Partitioning* dan *Boundary Value Analysis* (Studi Kasus: Aplikasi *Postcrossing*)” oleh Sofiyah tahun 2019. Perbandingan antara kedua metode tersebut adalah teknik *equivalence partitioning* dapat menguji pada tipe data *range* dan bebas, sedangkan *boundary value analysis* hanya dapat menguji tipe data *range*. Teknik *equivalence partitioning* cocok untuk mengeksplorasi semua kemungkinan berdasarkan kriteria, sedangkan

teknik *boundary value analysis* cocok untuk sistem yang kritis dan mengekspos masalah masukan *user*. Teknik *equivalence partitioning*

sulit dalam menentukan nilai representatif karena jangkauannya luas, sedangkan *boundaryvalue analysis* memiliki nilai yang lebih jelas [6].

III. METODE

A. Observasi Partisipasi

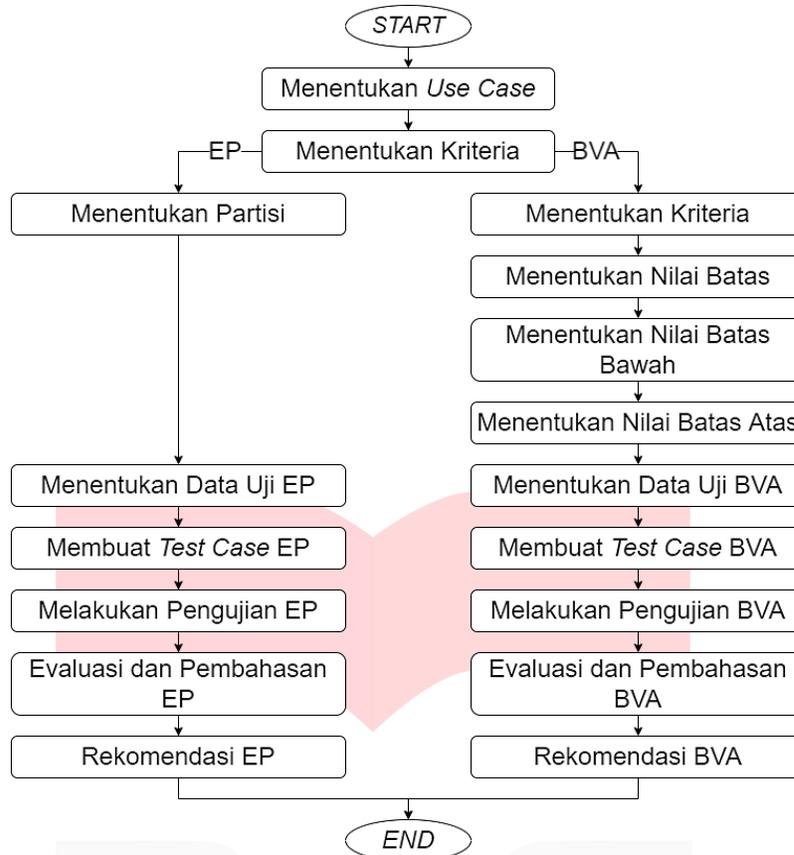
Partisipasi observasi adalah metode yang peneliti gunakan untuk mengumpulkan data, dan mereka bergabung dengan tim IT di PT Karsa Lintas Buwana (KLB) untuk mengamati dan menjalankan tes yang dilakukan dalam pengembangan *website* Kalcare.

B. Black Box Testing

Pengujian dijalankan dengan menggunakan metode pengujian *black box*. Pengujian *blackbox* dilakukan dari sudut pandang pengguna akhir. Pentingnya tes ini dalam mengidentifikasi input yang *valid* dan tidak *valid* dari perspektif pengguna akhir [19]. Teknik yang digunakan untuk pengujian ini adalah *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) dengan diagram alir penelitian yang ditunjukkan pada Gambar 1.

C. Boundary Value Analysis (BVA)

Boundary Value Analysis akan digunakan untuk mengetahui apakah Kalcare.com masih dapat berfungsi dengan baik ketika menerima inputan berupa panjang data yang tidak diharapkan. Langkah-langkah pada teknik BVA diantaranya:



GAMBAR 1.
DIAGRAM ALIR PENELITIAN

1. Menentukan nilai batas, nilai batas yang ditentukan adalah nilai batas bawah dan nilai batas atas yang kemudian disesuaikan berdasarkan level ekspresinya.
2. Menentukan nilai batas bawah, level ekspresi ini memiliki batas -1.
3. Menentukan nilai batas atas, level ekspresi ini memiliki nilai +1 yang valid.
4. Menentukan data uji *Boundary Value Analysis*, data uji yang dibuat adalah dasar untuk membuat *test case* [20].
5. Membuat *test case Boundary Value Analysis*, kasus uji berisi langkah-langkah pengujian yang ingin dijalankan. Langkah yang dilakukan harus berisi urutan fungsi awal ke akhir yang disebut dalam tes *Boundary Value Analysis*.
6. Melakukan pengujian *Boundary Value Analysis*, tes ini berjalan secara otomatis di Katalon Studio. Pada saat pengujian diperoleh hasil *valid* dan *invalid*, serta hasil *script test* berupa *fail/pass* [21].
7. Evaluasi dan pembahasan *Boundary Value Analysis*,
8. Rekomendasi *Boundary Value Analysis*

D. *Equivalence Partitioning* (EP)

Salah satu teknik pengujian *black box* yang

digunakan dalam penelitian ini adalah *Equivalence Partitioning* (EP). *Equivalence Partitioning* digunakan untuk mengetahui apakah pembacaan Kalcare.com masih berfungsi dengan baik saat menerima *input* berupa tipe data yang tidak diharapkan. *Test case* biasanya digunakan untuk menguji kesetaraan partisi. Seperti halnya *Boundary Value Analysis*, kunci untuk menyamakan pemisahan *test case* adalah kesimpulan yang mencakup keberhasilan atau kegagalan pengujian [22]. Langkah langkah pada teknik EP diantaranya:

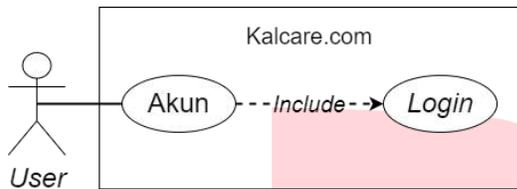
1. Menentukan partisi, setelah kriteria data diketahui, maka akan ditentukan partisi sesuai apa yang telah ditentukan pada bagian kriteria untuk *Equivalence Partitioning*.
2. Menentukan data uji *Equivalence Partitioning*, data uji adalah dasar untuk membuat *test case Equivalence Partitioning* [23].
3. Membuat *test case Equivalence Partitioning*, berisi langkah-langkah pengujian yang ingin dijalankan. Langkah yang dibuat harus berisi urutan fungsi awal ke akhir yang disebut dalam pengujian *Equivalence Partitioning* [20].
4. Melakukan pengujian *Equivalence*

Partitioning, tes ini berjalan secara otomatis di Katalon Studio. Pada saat tes, akan melihat hasil dengan hasil *valid* dan tidak *valid*, dan hasil *test script* berupa *fail* dan *pass*.

5. Evaluasi dan pembahasan *Equivalence Partitioning*
6. Rekomendasi *Equivalence Partitioning*

IV. HASIL DAN PEMBAHASAN

A. Use Case Diagram



TABEL 1.

USE CASE DESCRIPTION LOGIN

Nama Use Case	:	Login
Brief Description	:	Fitur ini digunakan oleh <i>user</i> untuk masuk ke dalam <i>web</i> Kalcare.com. <i>User</i> diharuskan <i>login</i> dengan memasukkan nomor telepon/ <i>Email</i> dan <i>password</i> yang telah terdaftar sebelumnya.
Primary Actor	:	<i>User</i>
Supporting Actor	:	<i>None</i>
Basic Flow	:	<ol style="list-style-type: none"> 1. <i>User</i> membuka <i>website</i> Kalcare.com 2. <i>User</i> klik <i>button</i> masuk 3. Kalcare.com menampilkan antarmuka <i>Login</i> 4. <i>User</i> memasukan <i>Email</i> atau nomor telepon yang sesuai <ul style="list-style-type: none"> A-1 <i>User</i> memasukan <i>Email</i> yang tidak sesuai A-2 <i>User</i> memasukan nomor telepon yang tidak sesuai 5. <i>User</i> mengklik <i>button</i> Selanjutnya 6. <i>User</i> memasukan <i>password</i> yang sesuai <ul style="list-style-type: none"> A-3 <i>User</i> memasukan <i>password</i> yang tidak sesuai 7. <i>User</i> mengklik <i>button</i> masuk
Alternative Flow	:	<p>A-1 <i>User</i> memasukan <i>Email</i> yang tidak sesuai</p> <ol style="list-style-type: none"> 1. Kalcare.com menampilkan pesan error 2. Kembali ke <i>basic flow</i> no.3 <p>A-2 <i>User</i> memasukan nomor telepon yang tidak sesuai</p> <ol style="list-style-type: none"> 1. Kalcare.com menampilkan pesan error 2. Kembali ke <i>basic flow</i> no.3 <p>A-3 <i>User</i> memasukan <i>password</i> yang tidak sesuai</p> <ol style="list-style-type: none"> 1. Kalcare.com menampilkan pesan error 2. Kembali ke <i>basic flow</i> no.3
Error Flow	:	<i>None</i>
Pre-Conditions	:	<i>User</i> membuka <i>website</i> Kalcare.com
Supporting Actor	:	<i>User</i> berhasil <i>Login</i>

GAMBAR 2.
USE CASE DIAGRAM

Pada Gambar 2 terdapat menu *akun* dan *login* yang dapat menerima masukan baik berupa angka maupun huruf pada *website*. Menu *login* merupakan menu pertama yang dapat diakses sebelum memasuki menu lainnya. Proses *login* dapat dilakukan dengan memasukkan nomor telepon/*Email* dan *password* yang telah terdaftar sebelumnya.

B. Use Case Description Login

Use case description untuk *use case login* ditunjukkan pada Tabel 1.

C. Kriteria Login

Kriteria *user login* ditunjukkan pada Tabel 2 dengan tiga buah *field* yaitu *email* dengan

kode K1, nomor telepon (K2) dan *password* (K3).

TABEL 2.
KRITERIA LOGIN

Kode	Nama Field	Kriteria
K1	Email	[Email] max 250
K2	Nomor telepon	[0-9] max 20
K3	Password	[Text] no max length

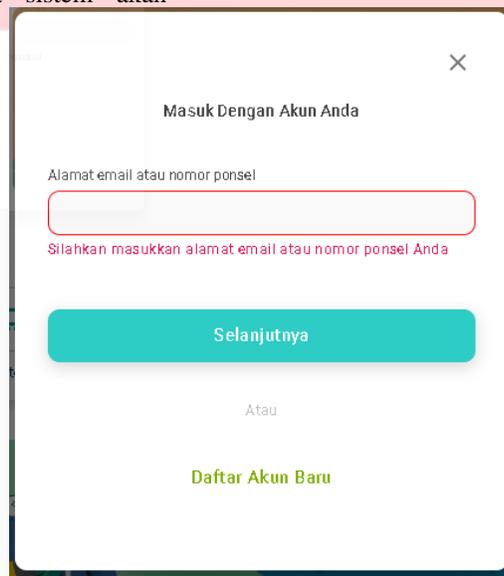
D. Boundary Value Analysis (BVA)

Pada BVA digunakan batas bawah dan batas atas dengan level ekspresi -1 pada batas bawah dan +1 pada batas atas.

memberitahu bahwa No HP/Email perlu diisi sehingga *user* tidak bisa memasuki tahap selanjutnya. Hasil pengujian batas bawah dengan teknik *Boundary Value Analysis* (BVA) pada *webiste* Kalcare.com ditunjukkan pada Gambar 3 dengan hasil yang positif dimana sistem tidak bisa berlanjut dan memberikan peringatan pada *user* untuk mengisi data terlebih dahulu.

1. Batas Bawah

Batas bawah dengan teknik *Boundary Value Analysis* (BVA) dengan level ekspresi -1 terjadi ketika *user* memberikan input pada laman *login* dengan No HP/Email kosong. Bila hasil pengujian positif maka sistem akan



GAMBAR 3.
PENGUJIAN BATAS BAWAH BVA PADA LAMAN LOGIN KALCARE.COM

2. Batas Atas

Pada pengujian batas atas, level ekspresi yang digunakan adalah +1 dimana menu *login* akan beralih pada laman beranda bila *user*

memasukkan No HP/Email yang *valid* pada laman *login* disertai dengan *password* yang *valid*.



GAMBAR 4.
HASIL PENGUJIAN BATAS ATAS BVA DENGAN NO HP/EMAIL SERTA PASSWORD VALID.

Berdasarkan hasil pengujian batas atas yang ditunjukkan pada Gambar 4, diketahui sistem memasuki beranda setelah memasukkan No HP/Email serta password yang valid, dengan demikian hasil pengujian dengan Boundary Value Analysis (BVA) dinyatakan 100% berhasil dan sistem login pada Kalcare.com bekerja dengan baik.

E. Equivalence Partitioning (EP)

Berbeda dengan sistem pengujian pada Boundary Value Analysis (BVA), pengujian dengan teknik Equivalence Partitioning (EP) dilakukan pada beberapa kondisi valid dan invalid seperti pada Tabel 3.

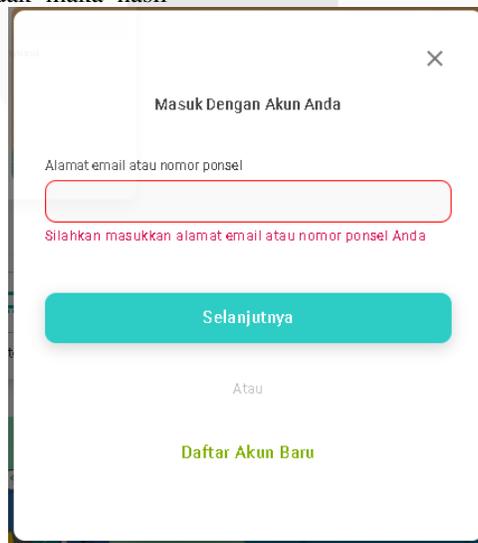
TABEL 3.
KONDISI VALID INVALID PADA PENGUJIAN EQUIVALENCE PARTITIONING (EP)

Pengujian	User Input		Output yang diharapkan
	No HP/Email	Password	
1	Kosong	Kosong	Peringatan
2	Invalid	Invalid	Peringatan
3	Valid	Invalid	Peringatan
4	Valid	Valid	Masuk Beranda

1. Pengujian 1

Pada pengujian 1 user memberikan input pada menu login dengan kondisi No HP/Email serta password kosong. Bila output yang dihasilkan memberikan peringatan untuk memasukkan No HP/Email serta password maka hasil pengujian positif, bila tidak maka hasil

pengujian negatif. Hasil pengujian 1 dengan teknik Equivalence Partitioning (EP) ditunjukkan pada Gambar 5 yang memberikan peringatan pada user untuk memasukkan No HP/Email. Dengan demikian pengujian 1 menggunakan teknik EP menunjukkan hasil yang positif.

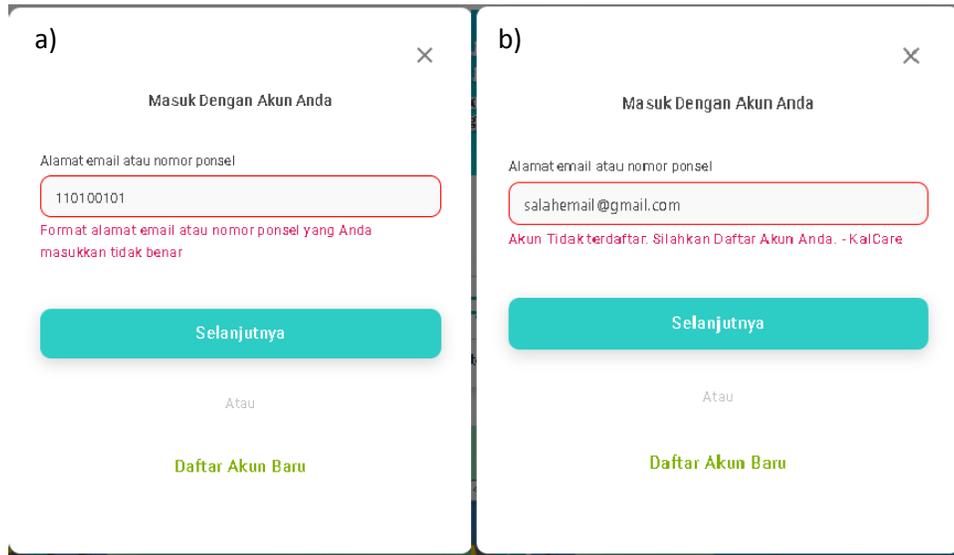


GAMBAR 5.
HASIL PENGUJIAN 1 EP PADA LAMAN LOGIN KALCARE.COM

1. Pengujian 2

Pada pengujian 2 *user* memberikan *input* pada menu *login* dengan kondisi No HP/*Email* serta *password invalid*. Bila *output* yang dihasilkan memberikan peringatan untuk *user* mengisi No HP/*Email* serta *password* dengan benar maka hasil pengujian positif, bila tidak

maka hasil pengujian negatif. Hasil pengujian 2 dengan teknik *Equivalence Partitioning* (EP) ditunjukkan pada Gambar 6 yang memberikan peringatan pada *user* untuk mengisi No HP/*Email* dengan benar. Dengan demikian pengujian 2 menggunakan teknik EP menunjukkan hasil yang positif.



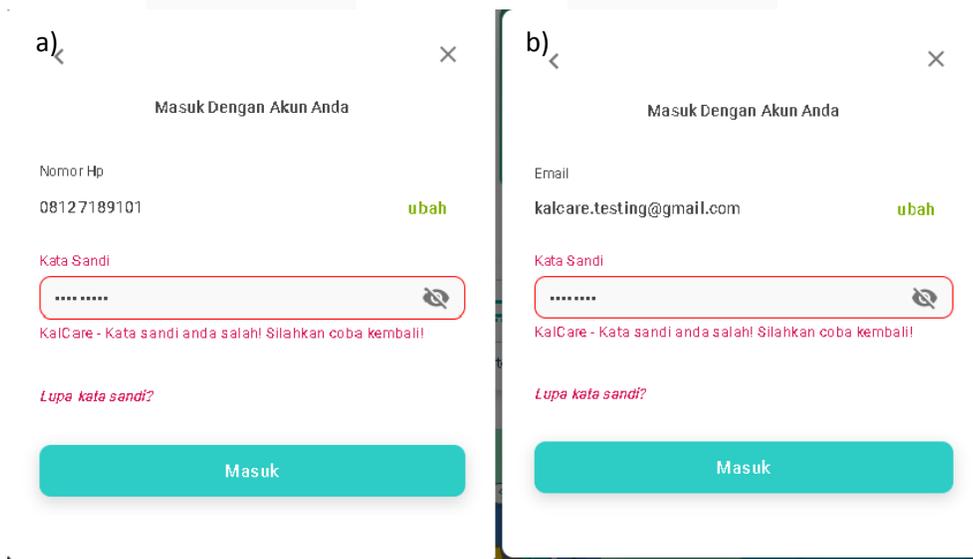
GAMBAR 6. HASIL PENGUJIAN 2 EP DENGAN A) NO HP INVALID, DAN B) EMAIL INVALID.

Berdasarkan hasil pengujian otomatis pada Gambar 6, diketahui bahwa sistem memberikan peringatan kepada *user* untuk mengisi No HP/*Email* dengan benar sehingga dapat memasuki laman pengisian *password*, dengan demikian pengujian 2 dengan metode EP

memberikan hasil yang positif.

2. Pengujian 3

Hasil pengujian 3 dengan teknik EP ditunjukkan pada Gambar 7.



GAMBAR 7. HASIL PENGUJIAN 3 EP DENGAN A) NO HP VALID PASSWORD INVALID, DAN B) EMAILVALID PASSWORD INVALID.

Pada pengujian 3 *user* memasukkan No HP/Email dengan benar sehingga *user* diharapkan memasuki laman pengisian *password*. Pada laman pengisian *password* *user* memasukkan *password* yang salah sehingga diharapkan sistem akan memberikan peringatan agar *user* memasukkan *password* dengan benar. Berdasarkan hasil pengujian pada Gambar 7, diketahui bahwa sistem memberikan peringatan agar *user* memasukkan *password* dengan benar, dengan demikian hasil pengujian

dapat dikatakan positif.

3. Pengujian 4

Pada pengujian 4 *user* memasukkan No HP/Email serta *password* dengan benar sehingga diharapkan sistem dapat mengarahkan *user* ke laman beranda. Hasil pengujian 4 dengan teknik EP ditunjukkan pada Gambar 8 dan dihasilkan bahwa *user* dapat memasuki laman beranda, dengan demikian hasil pengujian bersifat positif.



GAMBAR 8. HASIL PENGUJIAN 4 EP DENGAN NO HP/EMAIL SERTA *PASSWORD* VALID.

F. Pembahasan

Berdasarkan hasil pengujian

otomatis pada *website* Kalcare diperoleh hasil seperti pada Tabel 4.

TABEL 4. HASIL PENGUJIAN OTOMATIS DENGAN TEKNIK *BOUNDARY VALUE ANALYSIS* (BVA) DAN *EQUIVALENCE PARTITIONING* (EP)

Teknik	Pengujian	Hasil
<i>Boundary Value Analysis</i> (BVA)	Batas Bawah	Positif
	Batas Atas	Positif
<i>Equivalence Partitioning</i> (EP)	Pengujian 1	Positif
	Pengujian 2	Positif
	Pengujian 3	Positif
	Pengujian 4	Positif

Berdasarkan Tabel 4, diketahui bahwa seluruh hasil pengujian positif yang menunjukkan bahwa laman *login* pada *website* Kalcare.com 100% layak untuk dapat digunakan secara komersial. Dengan demikian hasil pengujian otomatis dengan metode *black box testing* dinyatakan berhasil.

V. KESIMPULAN

Pengujian otomatis pada *website* Kalcare.com dilakukan dengan *black box testing* menggunakan metode *Boundary Value Analysis* dan *Equivalence Partitioning* yang menunjukkan efisiensi kerja lebih baik

pada pengujian otomatis tanpa adanya kendala seperti human error dan durasi pengujian yang panjang dibandingkan dengan pengujian manual. Pengujian dengan teknik *Boundary Value Analysis* dilakukan pada dua kondisi yang menunjukkan bahwa laman *login* memberikan peringatan ketika *user* memasukkan input akun salah dan memasuki laman beranda ketika *user* mengisi akun dengan benar. Pada teknik *Equivalence Partitioning* dilakukan 4 pengujian dengan variasi kondisi valid dan invalid pada pengisian No HP/Email serta *password*. Berdasarkan hasil pengujian diketahui bahwa laman *login* pada *website*

Kalcare.com 100% bekerja dengan baik dan layak untuk digunakan secara komersial. Teknik terbaik yang direkomendasikan adalah dengan Equivalence Partitioning karena dapat meninjau lebih banyak kondisi sehingga menghasilkan hasil yang akurat.

REFERENSI

- [1] W. Wibisono and F. Baskoro, "Penguujian perangkat lunak dengan menggunakan model behaviour uml," *JUTI J. Ilm. Teknol. Inf*, 2002.
- [2] I. Dobles, A. Martinez, and C. Quesada-Lopez, "Comparing the effort and effectiveness of automated and manual tests: An industrial case study," *Iberian Conference on Information Systems and Technologies (CISTI)*, 2019.
- [3] V. Garousi and F. Elberzhager, "Test Automation: Not Just for Test Execution," *IEEE Software*, pp. 90–96, 2017.
- [4] M. A. Jamil, M. Arif, S. A. Abubakar, and A. Ahmad, "Software testing techniques: A literature review," in *Proceedings - 6th International Conference on Information and Communication Technology for the Muslim World, ICT4M*, 2017.
- [5] 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, p. 129, Jun. 2020, doi: 10.32493/informatika.v5i2.5366.
- [6] S. Ikhlaashi and H. P. Putro, "Komparasi Dua Teknik Black Box Testing: Equivalence Partitioning dan Boundary Value Analysis," *Prosiding Annual Research Seminar (ARS)*, vol. 5, no. 1, pp. 210–220, 2019.
- [7] T. Hidayat and H. D. Putri, "Penguujian Portal Mahasiswa pada Sistem Informasi Akademik (SINA) menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *JUTIS*, vol. 7, no. 1, 2019, [Online]. Available: www.ccsenet.org/cis
- [8] H. Albert, S. Hutapea, Y. Priyadi, and E. Darwiyanto, "Analisis dan Pengujian dengan Menggunakan Metode Boundary Value Analysis dan Metode Equivalence Partitioning (Studi Kasus: Aplikasi Homelab)," *e-Proceeding of Engineering*, vol. 9, no. 2, p. 746, 2022.
- [9] S. N. Yutia, "Automated Functional Testing pada API menggunakan Keyword Driven Framework," *Journal of Informatics and Communications Technology (JICT)*, vol. 3, no. 2, 2019.
- [10] C. H. Geneva, "Software and systems engineering –Software testing- Part 4: Test Techniques," *Standard, International Organization for Standardization*, 2015.
- [11] I. Jovanovic, "Software Testing Methods and Techniques," *IPSI BgD Trans. Internet Res.*, 2009.
- [12] B. Qu, C. Nie, B. Xu, and X. Zhang, "Test case prioritization for black box testing," in *Proceedings - International Computer Software and Applications Conference*, 2007.
- [13] A. Y. Aleryani, "Comparative Study between Data Flow Diagram and Use Case Diagram," *Int. J. Sci. Res. Publ.*, 2016.
- [14] S. Xu, L. Chen, C. Wang, and O. Rud, "A comparative study on black-box testing with open source applications," *IEEE/ACIS 17th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPDC*, 2016.
- [15] T. Suwannasart, "Generating Test Scripts for Web-Based Applications," *International Conference on Sensors, Signal and Image Processing*, vol. 2, 2020.
- [16] J. E. Bentley, "Software testing fundamentals—concepts, roles, and terminology," *Proc. SAS Conf*, 2005.
- [17] K. Vij and W. Feng, "Boundary value analysis using divide-and-rule approach," *Proceedings - International Conference on Information Technology: New Generations, ITNG*, 2008.
- [18] M. S. Mustaqbal, R. F. Firdaus, and

H. Rahmadi, "Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis," J. Ilm. Teknol. Inf. Terap., 2015.

[19] S. Nidhra, "Black Box and White Box Testing Techniques - A Literature Review," Int. J. Embed. Syst. Appl., 2012.

[20] Rully Pramudita, "Pengujian Black Box pada Aplikasi Ecampus Menggunakan Metode Equivalence Partitioning," INFORMATICS Educ. Prof., vol. 4, 2020.

[21] T. S. Jaya, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)," J. Inform. Pengemb. IT, 2018.

[22] M. S. A. Kesuma Jaya, P. Gumilang, T. Wati, Y. P. Andersen, and T. Desyani, "Pengujian Black Box pada Aplikasi Sistem Penunjang Keputusan Seleksi Calon Pegawai Negeri Sipil Menggunakan Teknik Equivalence Partitions," J. Inform. Univ. Pamulang, 2019.

[23] G. K. S. Tresnavitane, P. Mudjihartono, and Y. Harjoseputro, "Pengujian Aplikasi Mobile untuk Lelang Mobil dengan Metode Black Box menggunakan Automation Testing Tool," J. Inform. Atma Jogja, vol. 1, 2020.