

Pengujian Black Box Testing Pada Website Hej Vet! dengan Pendekatan Boundary Value Analysis

Frizaski Al Fath¹, Muhammad Yermi Rachman², Muhammad Faris Maulana Suryadani³,
Farhan Hakim⁴, Gema Parasti Mindara^{5*}, Aditya Wicaksono⁶

^{1,2,3,4,6}Program Studi Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi, IPB University

⁵Program Studi Teknologi Rekayasa Komputer, Sekolah Vokasi, IPB University

frizaskialfath@apps.ipb.ac.id, yermi2413rachman@apps.ipb.ac.id, farissuryadani@apps.ipb.ac.id,
4401122005farhan@apps.ac.id, gemaparasti@apps.ipb.ac.id, adityawicaksono@apps.ipb.ac.id

*Penulis Korespondensi

(Gema Parasti Mindara, IPB University, gemaparasti@apps.ipb.ac.id)

ABSTRAK

Manajemen data pasien dan rekam medis di klinik hewan peliharaan seringkali masih dilakukan secara manual, sehingga dapat menyebabkan kehilangan data dan memperlambat pelayanan bagi klien dan hewan peliharaan mereka. Penelitian ini bertujuan untuk menguji sistem informasi manajemen klinik hewan bernama "Hej Vet!", sebuah sistem manajemen berbasis web yang dirancang untuk membantu klinik hewan berjalan lebih lancar. Tujuan utama dari aplikasi ini adalah untuk mendigitalisasi proses administrasi dan medis, sehingga klinik dapat menjaga rekam medis secara terorganisir dan dapat beroperasi lebih efisien. Perancangan sistem melibatkan beberapa aktor utama, yaitu Admin, Dokter, dan Pemilik Hewan. Dengan sistem digital ini, semua aktivitas klinik yang terkait dengan pengelolaan data menjadi lebih terstruktur. Hasil dari penelitian ini adalah pengujian terhadap fitur-fitur kunci pada aplikasi "Hej Vet!", dengan fokus khusus pada proses *login* dan fitur tambah data hewan. Kedua fitur ini dipilih karena berkaitan langsung dengan alur autentikasi pengguna serta proses pengelolaan informasi hewan peliharaan. Berdasarkan hasil pengujian menggunakan metode Black Box Testing dengan pendekatan Boundary Value Analysis (BVA), beberapa fungsionalitas sistem masih belum berjalan sebagaimana mestinya dan masih perlu dilakukan perbaikan agar sistem dapat berfungsi secara optimal.

Kata kunci: Hej Vet!, Black box, Boundary Value Analysis, Klinik Hewan, Dokter Hewan, Sistem Informasi

ABSTRACT

Patient data and medical record management in veterinary clinics are often still carried out manually, which can lead to data loss and slower service for clients and their pets. This study aims to evaluate a web-based veterinary clinic management information system called "Hej Vet!", designed to help veterinary clinics operate more smoothly. The main goal of this application is to digitize administrative and medical processes so that clinics can maintain organized medical records and operate more efficiently. The system design involves several key actors: Admin, Doctor, and Pet Owner. With this digital system, all clinic activities related to data management become more structured. The result of this study is a test of key features in the "Hej Vet!" application, with a specific focus on the login process and the add animal data feature. These two features were chosen because they directly relate to the user authentication process and the pet information management process. Based on the testing results using the Black Box Testing.

Keywords: Hej Vet!, Black box, Boundary Value Analysis, Veterinary Clinic, Veterinary Doctor, Information System

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah membawa perubahan besar dalam berbagai aspek kehidupan, termasuk di bidang kesehatan hewan yang mulai mengadopsi *Veterinary Information Management System* (VIMS). Sistem ini dirancang untuk mengefisienkan penjadwalan, digitalisasi rekam medis, serta meningkatkan komunikasi klien untuk mengatasi kendala manajemen manual yang rentan kesalahan [1]. Di Indonesia, fenomena ini didukung oleh penggunaan internet masif dengan lebih dari 229 juta pengguna aktif, yang mendorong pengadopsian platform digital terutama pada sektor kesehatan [2]. Salah satu platform yang dikembangkan dalam konteks ini adalah "Hej Vet!", sebuah *website* layanan kesehatan hewan yang dirancang untuk memudahkan pemilik hewan dalam mengakses layanan klinik dan menyimpan riwayat kesehatan secara terintegrasi.

Namun, kualitas sistem *website* sering kali diabaikan pada tahap pengujian, sehingga banyak aplikasi menghadapi masalah seperti kesalahan *input*, autentikasi yang tidak valid, hingga penyimpanan data yang tidak

konsisten. Padahal, validasi *input* sangat penting untuk mencegah kesalahan saat pemrosesan data [3], sedangkan autentikasi diperlukan untuk memvalidasi identitas pengguna sebelum diberi akses ke sistem [4]. Hal ini penting dalam bidang medis, karena kegagalan pada mekanisme autentikasi dapat membahayakan kerahasiaan data pemilik, sementara kekeliruan dalam validasi data pasien (hewan) dapat menimbulkan inkonsistensi pada rekam medis yang dibutuhkan oleh dokter [5]. Tanpa pengujian yang memadai, kondisi ini menunjukkan adanya risiko pada layanan digital yang bersifat sensitif terhadap data pengguna. Oleh karena itu, metode *Black Box Testing* menjadi relevan karena teknik ini memungkinkan pengujian fungsional berdasarkan *input* dan *output* tanpa perlu mengetahui struktur internal dari *source code* pada aplikasi [6].

Sejumlah penelitian sebelumnya telah membahas tantangan dalam pembuatan *website* klinik hewan, namun aspek pengujiannya masih terbatas. Seperti penelitian yang dilakukan oleh Hasanuddin, Arwan dan Amalia pada pengembangan sistem manajemen klinik dengan hasil valid, namun belum merinci pada rancangan pengujian *input*-nya [7]. Fitriana dan Kristania menekankan kebutuhan fitur lengkap untuk mengatasi duplikasi data, namun belum mencantumkan metrik efektivitas deteksi kesalahan [8]. Penelitian lain oleh Syahidah *et al* [9], Elvida *et al* [10], serta Handayani *et al* [11] berhasil membangun sistem informasi fungsional, namun pengujian yang dilakukan bersifat umum tanpa pendekatan *Boundary Value Analysis* (BVA) yang sistematis. Padahal studi yang dilakukan oleh Joosten *et al.* membuktikan bahwa BVA lebih andal dalam menangani *input* di luar batas, meskipun cakupan studi mereka masih terbatas pada satu form pendaftaran saja [12]. Sementara itu, Wulan *et al.* juga menyoroti pentingnya rekam medis *online* terlepas dari belum diterapkannya batasan *input* pada penelitian yang mereka lakukan [13].

Kebaruan pada penelitian ini terletak pada penerapan pendekatan *Boundary Value Analysis* (BVA) secara mendalam untuk mengisi celah dari penelitian terdahulu yang kurang berfokus pada pengujian nilai batas. Beberapa penelitian menunjukkan bahwa BVA mampu memberikan validasi efektif terhadap *input* yang melebihi batas dengan tingkat keberhasilan mencapai 96% [14]. Teknik ini berfokus pada pengujian nilai minimum, maksimum, serta nilai di sekitar batas untuk memastikan aplikasi tetap berfungsi baik pada kondisi ekstrem [15]. Pendekatan ini belum banyak diterapkan secara spesifik pada fitur inti aplikasi hewan dalam penelitian-penelitian sebelumnya.

Tujuan penelitian ini adalah mengimplementasikan pengujian validasi data pada modul *login* dan fitur tambah hewan peliharaan di *website* "Hej Vet!". Pemilihan kedua fitur ini didasarkan pada urgensi fungsionalitasnya, di mana *login* berfungsi sebagai keamanan awal sistem [16], sedangkan fitur tambah hewan menjadi dasar pembentukan data utama yang berpengaruh terhadap konsistensi informasi medis selanjutnya. Fokus utamanya adalah untuk mendeteksi *bugs* saat menerima *input* batas dan memvalidasi mekanisme penanganan *error*. Pengujian yang berbasis persyaratan ini diharapkan dapat membantu mengidentifikasi celah lebih awal, meningkatkan efektivitas deteksi cacat, serta berkontribusi langsung pada upaya jaminan kualitas perangkat lunak secara menyeluruh [17].

2. METODOLOGI PENELITIAN

Metodologi penelitian ini disusun untuk memberikan gambaran yang jelas mengenai langkah-langkah yang ditempuh dalam menguji fungsionalitas *website* "Hej Vet!". Pada bagian ini, dijelaskan secara rinci mengenai pendekatan eksperimental yang digunakan, prosedur pengujian berbasis *Boundary Value Analysis* (BVA), serta teknik pengumpulan data untuk mendeteksi kesalahan pada fitur-fitur krusial aplikasi.

2.1. Desain Penelitian

Penelitian ini menggunakan desain penelitian dengan penelitian eksperimental deskriptif dengan metode pendekatan pengujian perangkat lunak. Dengan menggunakan pendekatan deskriptif eksperimental ini, peneliti dapat menilai efektivitas metode *Black Box Testing* menggunakan pendekatan *Boundary Value Analysis* (BVA) terhadap fungsi-fungsi tertentu dalam sistem tanpa melihat kode sumber program. Ini memungkinkan peneliti untuk menilai kesesuaian antara keluaran sistem dan spesifikasi kebutuhan (*Software Requirement Specification*).

Pendekatan *Boundary Value Analysis* digunakan karena fokusnya pada pengujian nilai-nilai batas *input*, yaitu nilai minimum, tepat di batas dan maksimum karena kesalahan perangkat lunak sering kali muncul pada area batas tersebut [15]. Oleh karena itu, BVA dipilih dalam penelitian ini untuk menguji validasi data pada fitur *login* dan tambah hewan di *website* "Hej Vet!", di mana setiap *input field* memiliki batas nilai tertentu seperti panjang teks dan rentang numerik usia hewan.

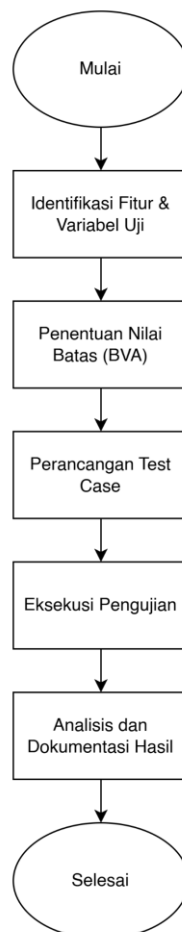
2.2. Prosedur Penelitian

Penelitian dilakukan melalui beberapa tahapan, yaitu:

1. Identifikasi Fitur dan Analisis Risiko. Tahap awal dimulai dengan menentukan ruang lingkup pengujian pada fitur *login* dan tambah data hewan. Pemilihan kedua fitur ini didasarkan pada analisis risiko sistem sebagai berikut:
 - Fitur *Login*: Memiliki risiko keamanan tinggi (*High Security Risk*). Kegagalan pada validasi *input* di fitur ini dapat membuka celah bagi akses tidak sah (*unauthorized access*) dan ancaman

- terhadap kerahasiaan data pemilik serta pasien [16].
- Fitur Tambah Data Hewan: Memiliki risiko integritas data (*Data Integrity Risk*) Kesalahan validasi pada tahap ini akan menyebabkan inkonsistensi informasi medis yang berdampak fatal pada ketepatan diagnosis dan pengambilan keputusan medis selanjutnya [5].
2. Penyusunan *test case* menggunakan pendekatan *Boundary Value Analysis* (BVA), dengan merancang *input* pada batas bawah, batas atas tepat pada batas, dan di luar rentang yang diperkenankan pada setiap *field*.
 3. Eksekusi pengujian dilakukan dengan memasukkan data ke sistem sesuai skenario *test case* yang telah disusun. Setiap *form* memiliki lebih dari satu *input field*, sehingga pengujian dilakukan dengan prinsip satu variabel diuji, sementara *field* lainnya diisi dengan data yang valid, kemudian melakukan observasi apakah sistem memproses *input* dengan benar dan menghasilkan keluaran sesuai harapan atau menampilkan pesan *error*.
 4. Hasil pengujian dicatat secara sistematis, meliputi keberhasilan proses, pesan *error* yang muncul, maupun penyimpangan dari *requirement* yang telah diharapkan.
 5. Analisis hasil dilakukan untuk mengevaluasi ketepatan respons sistem terhadap berbagai skenario *input*, mengidentifikasi *error* yang ditemukan, serta menyusun rekomendasi perbaikan apabila diperlukan.

Secara ringkas, alur kerja pengujian dengan metode *Boundary Value Analysis* (BVA) ini digambarkan dalam diagram alir pada Gambar 1.



Gambar 1. Alur Kerja Pengujian *Boundary Value Analysis* (BVA)

2.3. Sumber Data dan Lokasi Penelitian

Sumber data dalam penelitian ini merupakan data primer yang diperoleh secara langsung melalui proses pengujian fitur *login* dan tambah hewan pada *website* “Hej Vet!” Menggunakan teknik *Boundary Value Analysis* (BVA). Seluruh data *input* pada *form* dilakukan sendiri oleh tim penguji secara manual, dengan memasukkan skenario nilai batas untuk setiap *field* yang di uji, mulai dari batas bawah, atas, tepat di batas, hingga luar rentang valid. Lokasi penelitian ini bersifat daring, karena seluruh proses pengujian dilakukan secara langsung dengan mengakses *website* “Hej Vet!” melalui internet.

2.4. Lingkungan Pengujian

Pengujian website “Hej Vet!” dilakukan pada lingkungan pengujian tertentu yang mencakup sistem operasi, *browser*, bahasa pemrograman, *framework*, serta *database* yang digunakan. Aspek lain di luar lingkungan tersebut, seperti spesifikasi perangkat keras dan konfigurasi antarmuka (misalnya resolusi layar), tidak ditampilkan karena tidak memengaruhi hasil pengujian validasi batas input menggunakan metode Boundary Value Analysis.

Tabel 1. Lingkungan pengujian website “Hej Vet!”

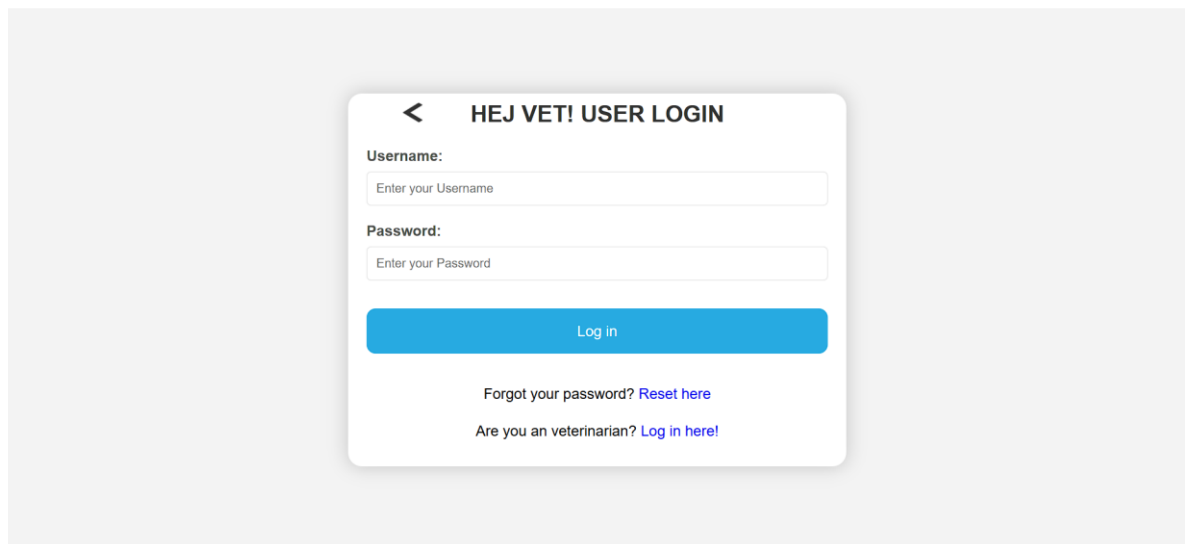
| Komponen Lingkungan | Spesifikasi |
|---------------------|--|
| Sistem operasi | vWindows 11 64-bit |
| <i>Browser</i> | Google Chrome 143.0.7499.109 (Official Build) (64-bit) |
| Bahasa Pemrograman | PHP |
| Framework | Laravel |
| <i>Database</i> | MySQL |

3. HASIL DAN PEMBAHASAN

Penelitian ini memfokuskan ruang lingkup pengujian pada dua fitur utama, yaitu fitur *login* dan fitur tambah data hewan, yang dipilih karena berhubungan langsung dengan autentikasi pengguna serta proses pencatatan informasi hewan peliharaan. Hasil pengujian disajikan secara sistematis dalam bentuk tabel *test case*. Setiap tabel memuat informasi penting yang mencakup *ID test case*, deskripsi pengujian, kondisi batas, hasil yang diharapkan, dan hasil pengujian yang telah dieksekusi.

3.1. Pengujian Fitur

A. Pengujian *Form Login*



Gambar 2. *Form Login Hej Vet!*

Pada *form login* yang ditunjukkan pada Gambar 1, terdiri atas dua bidang isian (*field*) wajib, yaitu *username* dan *password*, yang harus diisi untuk dapat mengakses sistem. Pengujian dengan metode *Boundary Value Analysis* (BVA) difokuskan pada kedua *field* tersebut karena memiliki batas *input* yang dapat diuji, sebagaimana dirinci pada Tabel 1.

Tabel 2. Analisis nilai batas *boundary value analysis field* pada halaman *login*

| <i>Input</i> | <i>Boundary Values</i> |
|-----------------|--|
| <i>Username</i> | <i>NULL</i> , a, aa, aaa...(sampai 254), aaa...(sampai 255), aaa...(sampai 256) |
| <i>Password</i> | <i>NULL</i> , 7 karakter, 8 karakter, 254 karakter, 255 karakter, 256 karakter |

Berdasarkan Tabel 1, analisis nilai batas (Boundary Value Analysis) pada *form login* dilakukan dengan menguji *input* pada titik-titik kritis panjang karakter. Untuk *username*, pengujian dimulai dari *NULL* sebagai batas bawah, kemudian “a” sebagai nilai tepat pada batas bawah, “aa” sebagai nilai tepat di atas batas bawah, “aaa...(255)” sebagai nilai tepat pada batas atas, dan “aaa...(256)” sebagai nilai yang berada di atas batas maksimum. Untuk *password*, Tabel 1 menunjukkan pola pengujian yang sama, yaitu *NULL* sebagai batas bawah, 7 karakter sebagai nilai di bawah batas minimum, 8 karakter sebagai batas bawah valid, 254–255 karakter sebagai panjang pada batas atas yang masih diterima, serta 256 karakter sebagai nilai yang melampaui batas atas.

Tabel 3. Hasil *Boundary Value Analysis field* pada halaman *login*

| ID | Deskripsi Pengujian | Hasil yang Diharapkan | Hasil |
|-----------------------|---|--|--------------|
| <i>Field Username</i> | | | |
| LOG-01 | Tepat di bawah batas bawah (1) → Tidak memasukkan nama (0 karakter) | Muncul <i>alert</i> “nama tidak boleh kosong”, <i>Form login</i> tidak dapat disubmit. (Ditolak) | <i>Pass</i> |
| LOG-02 | Tepat di batas bawah (1) → Memasukkan 1 karakter (<i>Input</i> = G) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-03 | Tepat di atas batas bawah (1) → Memasukkan 2 karakter (<i>Input</i> = Fi) | <i>Form login</i> dapat disubmit (Diterima) | <i>Pass</i> |
| LOG-04 | Tepat di bawah batas atas (255) → Memasukkan 254 karakter (<i>Input</i> =: aaa... (sebanyak 254 karakter)) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-05 | Tepat di batas atas (255) → Memasukkan 255 karakter (<i>Input</i> = aaa... (sebanyak 255 karakter)) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-06 | Tepat di batas atas (255) → Memasukkan 256 karakter (<i>Input</i> = aaa... (sebanyak 256 karakter)) | Menampilkan pesan <i>error</i> dan tidak bisa <i>login</i> . (Ditolak) | <i>Pass</i> |
| <i>Field Password</i> | | | |
| LOG-07 | Tepat di bawah batas bawah (1) → Tidak memasukkan nama (0 karakter) | Muncul <i>alert</i> “nama tidak boleh kosong”, <i>Form login</i> tidak dapat disubmit. (Ditolak) | <i>Pass</i> |
| LOG-08 | Tepat di bawah batas bawah (8) → Memasukkan 7 karakter (<i>Input</i> = Faris12) | Menampilkan pesan <i>error</i> dan tidak bisa <i>login</i> . (Ditolak) | <i>Pass</i> |

| | | | |
|--------|---|---|-------------|
| LOG-09 | Tepat di batas bawah (8) → Memasukkan 8 karakter (<i>Input</i> = Faris123) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-10 | Tepat di bawah batas atas (255) → Memasukkan 254 karakter (<i>Input</i> = 1Baaa...(sebanyak 254 karakter)) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-11 | Tepat di batas atas (255) → Memasukkan 255 karakter (<i>Input</i> = 1Baaa...(sebanyak 254 karakter)) | <i>Form login</i> dapat disubmit. (Diterima) | <i>Pass</i> |
| LOG-12 | Tepat di atas batas atas (255) → Memasukkan 256 karakter (<i>Input</i> : 1Baaa...(sebanyak 256 karakter)) | <i>Form login</i> tidak dapat disubmit. (Ditolak) | <i>FAIL</i> |

Berdasarkan hasil pengujian Tabel 2 pada *form login* menggunakan metode *Boundary Value Analysis* (BVA), dari total 12 skenario uji yang dilakukan, 11 di antaranya dinyatakan berhasil dan 1 mengalami kegagalan. Secara umum, hasil ini menunjukkan bahwa fungsionalitas *form login* telah berjalan dengan baik dan sesuai dengan batasan *input* yang ditentukan. Kasus kegagalan terjadi pada saat pengujian batas atas, yaitu ketika sistem tetap menerima *input* sebanyak 256 karakter pada *field username*, padahal batas maksimal yang telah ditetapkan adalah 255 karakter. Kondisi ini menunjukkan bahwa validasi panjang karakter pada *field* tersebut belum diterapkan secara optimal, sehingga diperlukan penyesuaian pada logika validasi *input* agar sistem dapat membatasi jumlah karakter sesuai dengan ketentuan yang berlaku.

B. Pengujian *Form* Tambah Data Hewan

The screenshot shows a web form titled "Tambahkan informasi hewan Anda" (Add your animal information). The form contains the following elements:

- Nama:** A text input field with the placeholder "Nama".
- Gender:** Radio buttons for "Jantan" (Male) and "Betina" (Female).
- Usia:** A text input field with the placeholder "Usia dalam bulan" (Age in months).
- Spesies:** A dropdown menu with the placeholder "Pilih spesies" (Select species).
- Ras:** A text input field with the placeholder "e.g Angora".
- Simpan Profil:** A button to save the profile.

Gambar 3. *Form* Tambah Data Hewan Website Hej Vet!

Form tambah data hewan pada Gambar 2 terdiri atas tiga *field* utama (Nama, Usia, dan Ras), satu *radio button* untuk *gender*, serta satu *dropdown* untuk spesies. Pengujian dengan metode *Boundary Value Analysis* (BVA) difokuskan pada ketiga *field* tersebut karena memiliki batas *input* yang dapat diuji, sebagaimana dirinci pada Tabel 3.

Tabel 4. Analisis nilai batas *Boundary Value Analysis* field pada halaman tambah hewan

| <i>Input</i> | <i>Boundary Values</i> |
|--------------|--|
| Nama | <i>NULL</i> , a, aa, aaa...(sampai 254), aaa...(sampai 255), aaa...(sampai 256) |
| Usia | -1, 0, 1, 719, 720, 721 |
| Ras | <i>NULL</i> , a, aa, aaa...(sampai 254), aaa...(sampai 255), aaa...(sampai 256) |

Berdasarkan Tabel 3, analisis nilai batas pada *form* tambah data hewan dilakukan untuk tiga field yang memiliki batas *input* yang jelas, yaitu Nama, Usia, dan Ras. Untuk *field* Nama dan Ras, batas bawah ditandai dengan *input NULL*, diikuti oleh “a” sebagai nilai tepat di batas bawah, “aa” sebagai nilai tepat di atas batas bawah, kemudian rentang panjang karakter hingga nilai ekstrem seperti 254 karakter (mendekati batas atas), 255 karakter (tepat di batas atas), dan 256 karakter (melewati batas atas). Sementara itu, untuk *field* Usia, pengujian dilakukan dengan nilai -1 sebagai batas bawah yang tidak valid, 0 sebagai nilai tepat di batas bawah, 1 sebagai nilai di atas batas bawah, serta 719, 720, dan 721 yang masing-masing merepresentasikan nilai mendekati batas atas, tepat di batas atas, dan melewati batas atas.

Tabel 5. Hasil *Boundary Value Analysis* field pada halaman tambah hewan

| ID | Deskripsi Pengujian | Hasil yang Diharapkan | Hasil |
|-------------------|---|---|-------------|
| Field Nama | | | |
| HWN-01 | Tepat di bawah batas bawah (1) → Tidak memasukkan nama (0 karakter) | <i>Form</i> tidak dapat disubmit. (Ditolak) | <i>Pass</i> |
| HWN-02 | Tepat di batas bawah (1) → Memasukkan 1 karakter (<i>Input</i> = A) | Data berhasil disimpan. (Diterima) | <i>Pass</i> |
| HWN-03 | Tepat di atas batas bawah (1) → Memasukkan 2 karakter (<i>Input</i> = Hi) | Data berhasil disimpan. (Diterima) | <i>Pass</i> |
| HWN-04 | Tepat di bawah batas atas (255) → Memasukkan 254 karakter (<i>Input</i> =: aaa... (sebanyak 254 karakter)) | Data berhasil disimpan. (Diterima) | <i>Pass</i> |
| HWN-05 | Tepat di batas atas (255) → Memasukkan 255 karakter (<i>Input</i> = aaa... (sebanyak 255 karakter)) | Data berhasil disimpan. (Diterima) | <i>Pass</i> |
| HWN-06 | Tepat di atas batas atas (255) → Memasukkan 256 karakter (<i>Input</i> = aaa... (sebanyak 256 karakter)) | Data tidak berhasil disimpan, muncul pesan <i>error</i> . (Ditolak) | <i>Pass</i> |
| Field Usia | | | |
| HWN-07 | Tepat di bawah batas bawah (1) → Memasukkan usia < 0 (<i>Input</i> = -1) | Data gagal disimpan. (Ditolak) | <i>Pass</i> |
| HWN | Tepat di batas bawah (1) → Memasukkan | Data berhasil disimpan. (Diterima) | <i>Pass</i> |

| | | | |
|------------------|--|--|------|
| -08 | usia (0, 720) (<i>Input</i> = 0) | | |
| HWN -09 | Tepat di atas batas bawah (1) → Memasukkan usia (0, 720) (<i>Input</i> = 1) | Data berhasil disimpan. (Diterima) | Pass |
| HWN -10 | Tepat di bawah batas atas (719) → Memasukan usia (0 ,720) (<i>Input</i> = 719) | Data berhasil disimpan. (Diterima) | Pass |
| HWN -11 | Tepat di batas atas (720) → Memasukan usia (0, 720) (<i>Input</i> = 720) | <i>Form login</i> dapat disubmit. (Diterima) | Pass |
| HWN -12 | Tepat di atas batas atas (721) → Memasukan usia > 720 (<i>Input</i> = 721) | Data tidak berhasil disimpan, muncul pesan <i>error</i> . (Ditolak) | FAIL |
| Field Ras | | | |
| HWN -13 | Tepat di bawah batas bawah (1) → Tidak memasukkan nama (0 karakter) | <i>Form</i> tidak dapat disubmit. (Ditolak) | Pass |
| HWN -14 | Tepat di batas bawah (1) → Memasukkan 1 karakter (<i>Input</i> = A) | Data berhasil disimpan.(Diterima) | Pass |
| HWN -15 | Tepat di atas batas bawah (1) → Memasukkan 2 karakter (<i>Input</i> = Hi) | Data berhasil disimpan.(Diterima) | Pass |
| HWN -16 | Tepat di bawah batas atas (255) → Memasukkan 254 karakter (<i>Input</i> =: aaa... (sebanyak 254 karakter) | Data berhasil disimpan.(Diterima) | Pass |
| HWN -17 | Tepat di batas atas (255) → Memasukkan 255 karakter (<i>Input</i> = aaa... (sebanyak 255 karakter) | Data berhasil disimpan.(Diterima) | Pass |
| HWN -18 | Tepat di atas batas atas (255) → Memasukkan 256 karakter (<i>Input</i> = aaa... (sebanyak 256 karakter)) | Data tidak berhasil disimpan, muncul pesan <i>error</i> . (Ditolak) | Pass |

Berdasarkan hasil pengujian Tabel 4 pada fitur tambah data hewan menggunakan metode *Boundary Value Analysis (BVA)*, dari total 18 skenario uji yang dilakukan, 17 di antaranya dinyatakan berhasil dan 1 mengalami kegagalan pada *field* usia di *form* tambah hewan. Secara umum, hasil ini menunjukkan bahwa fungsionalitas formulir telah berjalan sesuai dengan batasan *input* yang ditetapkan. Kasus kegagalan terjadi pada pengujian batas atas nilai Usia. Hal ini ditunjukkan oleh sistem yang masih menerima *input* sebesar 721 bulan, padahal batas maksimal yang telah ditentukan adalah 720 bulan. Kondisi ini mengindikasikan bahwa validasi terhadap batas nilai pada *field* Usia belum diterapkan secara optimal, sehingga perlu dilakukan penyesuaian untuk memastikan sistem dapat menolak *input* yang melebihi batas yang telah ditentukan.

3.2. Ringkasan dan Analisis Dampak Error

Tabel 5. Ringkasan Hasil Pengujian

| Fitur yang diuji | Total Test Case | Berhasil (Pass) | Gagal (FAIL) | Persentase Keberhasilan | Persentase Kegagalan |
|------------------|-----------------|-----------------|--------------|-------------------------|----------------------|
| Login | 12 | 11 | 1 | 91.67% | 8.33% |

| | | | | | |
|-------------------|----|----|---|--------|-------|
| Tambah Data Hewan | 18 | 17 | 1 | 94.44% | 5.56% |
| Total Keseluruhan | 30 | 28 | 2 | 93.33% | 6.67% |

Hasil pengujian *Black Box Testing* dengan metode *Boundary Value Analysis* (BVA) pada *website* "Hej Vet!" menunjukkan tingkat keberhasilan keseluruhan sebesar 93,33%, di mana 28 dari total 30 skenario uji dinyatakan valid. Secara spesifik, fitur *login* mencapai tingkat keberhasilan 91,67% (11 dari 12 kasus berhasil), sedangkan fitur tambah data hewan mencapai 94,44% (17 dari 18 kasus berhasil), dengan kegagalan yang terdeteksi hanya pada pengujian batas atas (*upper boundary*) di kedua modul tersebut. Meskipun persentase *error* tergolong kecil (6,67%), dampak yang ditimbulkan cukup signifikan terhadap kualitas dan keamanan data.

Kegagalan memvalidasi *input* usia yang melebihi batas (seperti 721 bulan) pada fitur tambah data hewan berisiko mendistorsi akurasi rekam medis yang vital bagi diagnosis dokter. Jika data usia yang tidak valid ini digunakan sebagai parameter otomatis untuk dosis obat atau saran perawatan, hal ini dapat membahayakan kesehatan hewan karena validasi input sangat krusial untuk mencegah kesalahan pemrosesan data. Selain itu, data yang melebihi spesifikasi yang diharapkan dapat menyebabkan anomali saat pembuatan laporan statistik klinik atau saat data diekspor ke sistem lain yang memiliki validasi lebih ketat.

Sementara itu, dari aspek keamanan, kegagalan validasi pada fitur *login* yang tetap menerima *input username* melebihi 255 karakter membawa risiko teknis yang signifikan. Risiko utama yang muncul adalah *Buffer Overflow* atau *Truncation*, yakni ketika penerimaan karakter yang melebihi kapasitas basis data tanpa penanganan *error* yang tepat dapat memicu kegagalan eksekusi *SQL query* yang tidak terduga atau memunculkan kesalahan sistem yang berpotensi mengekspos informasi sensitif terkait struktur *database* kepada pengguna. Selain itu, ketiadaan pembatasan panjang karakter yang ketat di sisi server membuka celah kerentanan terhadap serangan *Denial of Service* (DoS). Hal ini memungkinkan penyerang untuk membebani memori server dengan mengirimkan *string* jutaan karakter yang berpotensi melumpuhkan atau memperlambat pelayanan bagi klien lain. Lebih jauh lagi, kegagalan pada fitur autentikasi ini menjadi ancaman privasi yang serius, karena celah tersebut dapat dimanfaatkan sebagai pintu masuk eksploitasi lebih lanjut yang membahayakan keamanan data pribadi pemilik hewan.

4. KESIMPULAN

Berdasarkan hasil pengujian terhadap *website* manajemen klinik hewan "Hej Vet!", penerapan metode *Black Box Testing* dengan pendekatan *Boundary Value Analysis* (BVA) terbukti efektif dalam memverifikasi keandalan fungsi sistem, khususnya dalam menangani *input* pada nilai batas. Secara umum, fitur *login* dan tambah data hewan telah berjalan sesuai dengan kebutuhan pengguna. Sebagian besar skenario pada pengujian fitur *login* dan tambah hewan menunjukkan hasil yang valid.

Namun, pengujian menemukan kelemahan pada validasi batas atas (*upper boundary*) yang memiliki implikasi serius. Sistem masih gagal menolak *input* yang melebihi batas maksimum, seperti nilai numerik pada kolom usia dan panjang karakter *username*. Kegagalan validasi pada *input* usia berisiko mendistorsi akurasi rekam medis yang vital bagi diagnosis dokter dan penentuan dosis obat. Sementara itu, penerimaan *input username* yang melebihi kapasitas karakter membawa risiko keamanan teknis, mulai dari inkonsistensi data akibat pemotongan (*truncation*) hingga potensi kerentanan terhadap beban server atau *Denial of Service* (DoS). Temuan ini menegaskan bahwa celah validasi pada kondisi ekstrem tidak hanya memengaruhi fungsionalitas, tetapi juga mengancam integritas data pasien dan keamanan sistem secara keseluruhan.

Kontribusi penting penelitian ini adalah memberikan evaluasi empiris terhadap kualitas perangkat lunak layanan veteriner sekaligus menawarkan rekomendasi teknis bagi pengembang untuk menyempurnakan logika validasi *input* demi mencegah kesalahan pemasukan data dan menjaga integritas rekam medis digital. Meski demikian, ruang lingkup pengujian masih terbatas pada dua modul utama (*login* dan tambah hewan) serta belum mencakup aspek kode internal. Untuk pengembangan berikutnya, penulis menyarankan untuk memperluas pengujian ke fitur lain seperti rekam medis, penjadwalan, serta menambahkan pengujian performa dan keamanan guna memastikan kualitas sistem secara menyeluruh.

DAFTAR PUSTAKA

- [1] S. J. Cachuela, D. M. Matias, R. J. Jose, dan J. Altura, “Navigating Veterinary Practice in the Digital Age: Implementing a Web-Based Information Management System at Animals’ Choice Clinic,” *Psychol. Educ. Multidiscip. J.*, vol. 37, no. 5, hlm. 466–473, Mei 2025, doi: 10.70838/pemj.370506.
- [2] S. Shabrina, “APJII Rilis Data Terbaru 2025: Pengguna Internet di Indonesia Capai 229 Juta Jiwa,” *Teknologi.id*. Diakses: 13 November 2025. <https://teknologi.id/teknologi/apjii-rilis-data-terbaru-2025-pengguna-internet-di-indonesia-capai-229-juta-jiwa>
- [3] H. Liu dan H. B. K. Tan, “An approach for the maintenance of input validation,” *Inf. Softw. Technol.*, vol. 50, no. 5, hlm. 449–461, 2008.
- [4] J. M. Kizza, “Authentication,” dalam *Guide to Computer Network Security*, Springer, 2024, hlm. 215–238.
- [5] E. T. Ardianto, L. Nurjanah, dan others, “Analisis Aspek Keamanan Data Pasien dalam Implementasi Rekam Medis Elektronik di Rumah Sakit X,” *J. Rekam Med. Manaj. Inf. Kesehat.*, vol. 3, no. 2, hlm. 18–30, 2024.
- [6] M. E. Khan dan F. Khan, “A Comparative Study of White Box, Black Box and Grey Box Testing Techniques,” 2012. [Daring]. Tersedia pada: www.ijacsa.thesai.org
- [7] F. M. Hasanuddin, A. Arwan, dan F. Amalia, “Pengembangan Sistem Manajemen Klinik Hewan Berbasis Web (Studi Kasus: Rakki Pet Clinic & Klinik Hewan UB),” Jan 2021. [Daring]. Tersedia pada: <http://j-ptiik.ub.ac.id>
- [8] S. Fitriana dan Y. M. Kristania, “Perancangan Sistem Informasi Klinik Hewan Berbasis Android,” *Jurnal Sains Dan Manaj.*, vol. 9, no. 2, 2021.
- [9] K. Syahidah, S. Esabella, dan E. Mardinata, “Sistem Informasi Pelayanan Kesehatan Hewan Pada Klinik Dinas Peternakan Dan Kesehatan Hewan Kabupaten Sumbawa Berbasis Web,” *Manag. Inf. Syst. J.*, vol. 2, no. 3, hlm. 43–48, Jul 2024, doi: 10.47065/mis.v2i3.1424.
- [10] R. Elvida, N. W. Al-Hafiz, dan M. H. Siregar, “Sistem Informasi Rekam Medis Hewan Peliharaan Berbasis Web,” dalam *Prosiding Seminar Nasional Hasil Penelitian Dan Pengabdian Kepada Masyarakat*, 2021, hlm. 46–52.
- [11] S. Handayani, J. Prayoga, dan B. S. Hasugian, “Sistem Informasi Klinik Hewan Monsabel Pet’s Clinic,” *DEVICE J. Inf. Syst. Comput. Sci. Inf. Technol.*, vol. 4, no. 1, hlm. 33–43, Jun 2023.
- [12] Joosten, Handoko, Murdiaty, dan Elly, “Perbandingan Metode Black Box Testing dalam Pengujian Tampilan Pendaftaran Rumah Sakit,” vol. 4, no. 2, 2024, doi: 10.46880/tamika,vol 4, no 2 (SEMNASTIK). pp8-13.
- [13] T. D. Wulan, T. W. Bintarti, N. F. W. Salamy, F. A. Susanto, dan D. R. Hartika, “Pendampingan dan Sosialisasi Pemanfaatan Rekam Medis Elektronik Pada Klinik Hewan,” UNUSA, 2021. [Daring]. Tersedia pada: <https://snpm.unusa.ac.id>
- [14] F. K. Kartono *dkk.*, “Pengujian black box testing pada sistem website Osha Snack: Pendekatan teknik boundary value analysis,” *J. Kridatama Sains Dan Teknol.*, vol. 6, no. 02, hlm. 754–766, 2024.
- [15] A. Jailani dan M. A. Yaqin, “Pengujian aplikasi sistem informasi akademik menggunakan metode blackbox dengan teknik boundary value analysis,” *J. Autom. Comput. Inf. Syst.*, vol. 4, no. 2, hlm. 60–66, 2024.
- [16] D. M. Khairina, “Analisis Keamanan Sistem Login,” *J. Inform. Mulawarman*, vol. 6, no. 2, hlm. 64–67, 2011.
- [17] A. M. G. Esperón, F. M. Pérez, J. V. B. Martínez, M. D. D. Dapena, dan I. L. Fonseca, “Specifying requirements for modern software development: A test-oriented methodology,” *Int. J. Softw. Eng. Knowl. Eng.*, vol. 34, no. 01, hlm. 27–48, 2024.