



Pengujian *Blackbox* pada Sistem Informasi Komunitas Pecinta Kucing di Bandar Lampung

Ruki Rizal Nul Fikri¹, Indera Indera*², Agus Rahardi³ Isnandar Agus⁴

^{1,2,3,4}Institut Informatika dan Bisnis Darmajaya, Jl. Z.A Pagar Alam No. 93 Bandar Lampung 35142 Indonesia, Telp. (0721) 787214/Fax. (0721) 700261

*Email Penulis Korespondensi: indera@darmajaya.ac.id

Abstrak

Penelitian ini merespon dengan pertumbuhan minat komunitas pecinta kucing di Bandar Lampung dengan menelusuri dan mengevaluasi keandalan Sistem Informasi Forum Komunitas Pecinta Kucing melalui pendekatan blackbox testing. Fokus penelitian ditujukan untuk mengamati fungsionalitas sistem, untuk menemukan kekurangan dari sistem yang sebelumnya tidak diketahui oleh komunitas. Melalui penerapan metode blackbox testing, kinerja sistem dievaluasi dengan skenario penggunaan uji fungsional. Hasil pengujian akan mengungkap sejumlah aspek terkait performa sistem, serta menemukan bagian masalah yang dapat disarankan untuk diperbaiki. Dari 12 skenario pengujian yang digunakan, 75% pengujian menghasilkan hasil yang valid atau lulus uji, sementara 25% menghasilkan hasil yang tidak valid atau tidak valid, hal ini menandakan perlunya perbaikan dan tindak lanjut untuk meningkatkan pengalaman pengguna sistem. Dari penelitian ini dapat disimpulkan bahwa pengujian blackbox pada Sistem Informasi Komunitas Pecinta Kucing di Bandar Lampung memiliki peran yang sangat penting dalam menemukan masalah pada sistem tersebut, sehingga akan meningkatkan kewaspadaan pada komunitas pada sistem yang telah dibangun serta untuk pengembangan sistem pada masa yang akan datang.

Kata kunci—Uji Blackbox, Uji Sistem Informasi, Uji Fungsional

Abstract

This research responds to the growing interest of the cat lover community in Bandar Lampung by investigating and evaluating the reliability of the Cat Lover Community Forum Information System through blackbox testing. The research focuses on examining the system's functionality to identify deficiencies previously unknown to the community. The application of blackbox testing methodology allows the evaluation of system performance through usage scenarios and functional tests. The test results reveal various aspects related to system performance and identify problematic areas that can be recommended for improvement. Out of the 12 testing scenarios employed, 75% yielded valid results, while 25% produced invalid results, indicating the need for corrective measures and follow-up actions to enhance the user experience of the system. In conclusion, this study underscores the critical role of blackbox testing in uncovering system issues, thereby increasing awareness within the community about the system in place and facilitating future system development.

Keywords—Blackbox Testing, System Information Evaluation, Functional Evaluation

1. PENDAHULUAN

Pertumbuhan minat yang signifikan terhadap komunitas pecinta kucing di Bandar Lampung mendorong perhatian terhadap kehandalan Sistem Informasi Forum Komunitas Pecinta Kucing. Kenyamanan dan keamanan pada sistem tersebut menjadi faktor penting dalam menjaga kehandalan dan keamanan sistem. Oleh karena itu, penelitian ini bertujuan untuk menyelidiki dan meningkatkan keamanan serta fungsionalitas sistem melalui pendekatan *blackbox testing*.

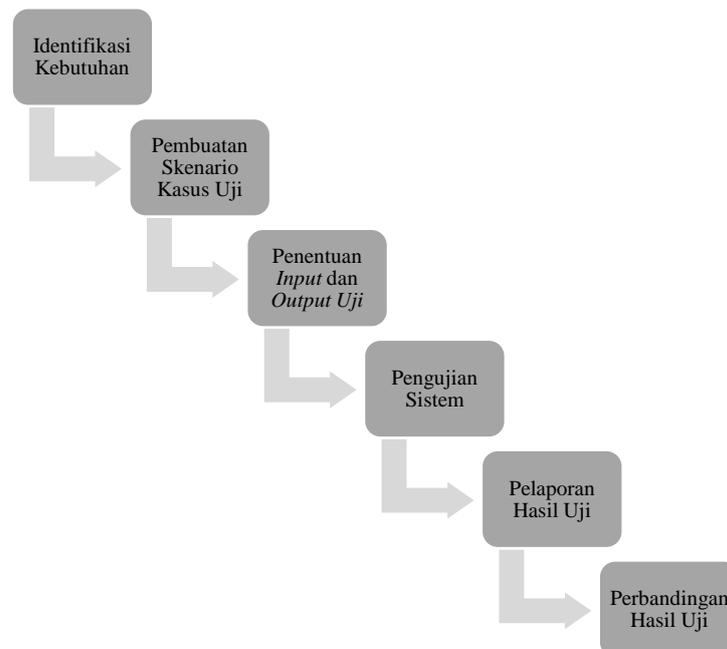
Blackbox testing adalah metode pengujian perangkat lunak yang berfokus pada evaluasi fungsionalitas suatu sistem tanpa memerhatikan struktur internalnya. Pengujian ini dilakukan dengan melihat *input* dan *output* yang dihasilkan oleh sistem tanpa mengetahui struktur kode dan algoritma didalamnya [1]. Pendekatan metode ini dengan melakukan pengujian dari sudut pandang pengguna, Sehingga akan didapat kelemahan dan kekurangan yang bisa terdeteksi untuk segera diperbaiki dan ditingkatkan [2]. Kelebihan dari metode ini adalah kemampuannya untuk menguji sistem tanpa harus memahami alur yang ada didalam sistem, sehingga pengujian dapat dilakukan tanpa mengganggu inti kode yang ada didalam sistem tersebut, selain itu penguji juga tidak memerlukan pengetahuan pengkodean komputasi karena penguji dapat mengidentifikasi masalah dari perspektif pengguna akhir [3]. Namun, metode ini juga memiliki kekurangan, seperti kurangnya detail tentang struktur internal sistem, keterbatasan cakupan dalam menemukan semua masalah potensial, dan ketergantungan pada pengujian fungsionalitas yang mungkin kurang efektif dalam menangani masalah kinerja atau keamanan yang mendasar [4].

Selain itu, *blackbox testing* memiliki keunggulan yang membuatnya lebih baik dari beberapa metode evaluasi testing lainnya. Kelebihan utamanya adalah dalam kemampuannya menciptakan pandangan yang objektif terhadap fungsionalitas suatu sistem, pendekatan ini memberikan keuntungan signifikan dalam beberapa aspek, yaitu, pertama, karena *blackbox testing* tidak melakukan pengujian pada *internal* sistem, maka penguji memiliki objektivitas yang lebih baik, karena penguji hanya berinteraksi dengan sistem [5]. Objektivitas ini memastikan bahwa hasil pengujian tidak dipengaruhi oleh pengetahuan atau persepsi penguji terhadap cara sistem seharusnya beroperasi. Kedua, kelebihan *blackbox testing* terletak pada pemahaman pengguna. Dengan menempatkan diri dalam posisi pengguna akhir, pengujian ini dapat mengidentifikasi masalah atau kekurangan sistem dari perspektif yang paling relevan [6]. Ini memungkinkan penguji untuk menggali masalah yang mungkin tidak terdeteksi jika diuji dengan fokus pada struktur internal saja. Dengan memahami bagaimana pengguna berinteraksi dengan sistem, *blackbox testing* dapat menilai sejauh mana sistem dapat memenuhi kebutuhan dan harapan pengguna [7].

Penelitian sebelumnya menunjukkan efektivitas *blackbox testing* dalam menguji sistem keamanan, pada sistem bernama *fuzzing blackbox* dengan *Rest API*. Dimana pengujian *blackbox* ini melakukan evaluasi di area deteksi *non-authorized request* pada sistem dan juga kesalahan yang terjadi pada *Rest API* yang dapat menyebabkan celah keamanan pada sistem [8]. Dalam penelitian lain juga *blackbox testing* digunakan untuk menguji aplikasi penilaian kinerja karyawan yang digunakan oleh PT. Cakra Jasa Pasific, dari 7 skenario yang diuji dihasilkan nilai presisi sebesar 100%. Keberhasilan implementasi metode pencocokan profil menunjukkan potensi untuk meningkatkan akurasi dan efisiensi proses penilaian, pada akhirnya berkontribusi pada sistem manajemen sumber daya manusia yang lebih efektif untuk organisasi tersebut [9]. Selain itu *blackbox testing* juga digunakan untuk menguji pada perangkat seluler, dimana Hasil uji *blackbox* dari aplikasi ini menunjukkan bahwa semua fungsi dan menu berfungsi dengan baik seperti yang diharapkan. Hasil uji menunjukkan skor rata-rata sebesar 91%, yang menunjukkan bahwa *blackbox testing* merupakan metode yang ideal untuk melakukan evaluasi pada sistem berbasis *mobile* [10]. Berdasarkan hal tersebut, penggunaan *blackbox testing* dapat dianggap sebagai pendekatan yang efektif untuk memastikan keamanan dan fungsionalitas yang optimal.

2. METODE PENELITIAN

Blackbox testing adalah sebuah metode pengujian perangkat lunak yang terfokus pada evaluasi fungsionalitas suatu sistem tanpa memerhatikan struktur internalnya, sehingga pengujian ini dapat digunakan untuk memastikan keamanan dan kinerja optimal pada sistem tanpa memerlukan pemahaman teknis mendalam terhadap implementasi *internal* sistem. Pada penelitian ini kami menggunakan 6 langkah metode dalam pengujian *blackbox testing* seperti yang ditunjukkan pada Gambar 1.



Gambar 1 Tahapan *Blackbox Testing* [11]

2.1 Identifikasi Kebutuhan

Identifikasi kebutuhan melibatkan langkah-langkah seperti memeriksa dokumen spesifikasi sistem, berkomunikasi dengan pihak terkait, dan menetapkan prioritas kebutuhan berdasarkan urgensi dan dampaknya. Fokus diberikan pada fungsi-fungsi krusial, dan setiap kebutuhan dijabarkan menjadi skenario kasus uji spesifik. Dokumentasi yang terperinci ini akan berperan sebagai panduan utama untuk memastikan bahwa pengujian dilakukan sesuai dengan kebutuhan dan tujuan yang telah ditentukan [12].

2.2 Pembuatan Kasus Uji

Pembuatan kasus uji dalam *blackbox testing* melibatkan analisis kebutuhan, identifikasi skenario penggunaan, dan penentuan langkah-langkah serta output yang diinginkan [13]. Setiap kasus uji didokumentasikan dengan judul, deskripsi skenario, langkah-langkah, *input*, dan *output* yang diharapkan, sambil mempertimbangkan batasan sistem. Dengan demikian, proses ini memastikan evaluasi menyeluruh terhadap fungsionalitas sistem sesuai dengan skenario penggunaan yang mungkin terjadi.

2.3 Penentuan Input dan Output

Penentuan *input* dan *output* dalam *blackbox testing* melibatkan proses spesifikasi parameter-parameter yang akan diuji dan hasil yang diharapkan dari sistem. *Input* ini dapat berupa data, perintah, atau kondisi tertentu yang dimasukkan ke dalam sistem [14]. *Output* yang diinginkan adalah respons atau hasil yang seharusnya dihasilkan oleh sistem berdasarkan *input*

yang diberikan. Hasil yang diharapkan yang telah ditentukan sebelumnya, akan menjadi perbandingan antara keluaran aktual dan yang diinginkan untuk mengevaluasi keberhasilan sistem. Dengan cara ini, maka mendapatkan hasil uji sistem secara menyeluruh.

2.4 Pengujian Sistem

Pengeksekusian uji dalam *blackbox testing* adalah langkah di mana tim pengujian menjalankan kasus uji yang telah disusun sebelumnya [15]. Proses ini melibatkan pemberian *input* yang telah ditentukan ke dalam sistem dan pengamatan terhadap keluaran atau respons yang dihasilkan oleh sistem. Selama eksekusi uji, hasil aktual dari sistem dicatat dan dibandingkan dengan hasil yang diharapkan, sehingga dapat diidentifikasi apakah ada kesalahan atau perbedaan yang perlu diperbaiki.

2.5 Pelaporan Hasil Uji

Setelah eksekusi kasus uji, tim pengujian membandingkan output yang dihasilkan oleh sistem dengan keluaran yang seharusnya terjadi sesuai dengan skenario uji. Dengan membandingkan hasil aktual dengan yang diharapkan, tim dapat mengidentifikasi kesalahan atau ketidaksesuaian yang mungkin terjadi [16].

2.6 Perbandingan Hasil Uji

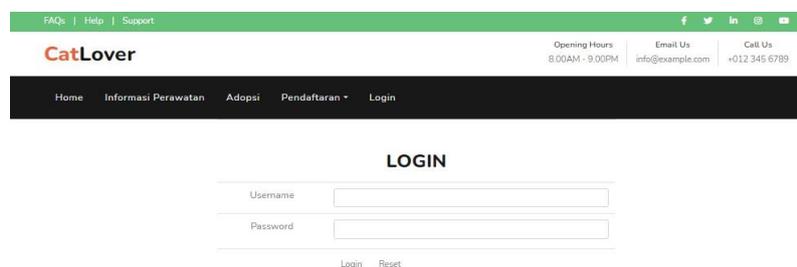
Pelaporan hasil dalam *blackbox testing* melibatkan penyusunan laporan yang mencakup informasi tentang hasil pengujian, temuan kesalahan, dan rekomendasi perbaikan. Laporan ini disiapkan setelah pengeksekusian uji dan perbandingan hasil aktual dengan yang diharapkan [17]. Isi laporan meliputi ringkasan hasil pengujian, deskripsi kesalahan yang ditemukan, langkah-langkah untuk mereproduksi kesalahan, serta rekomendasi untuk perbaikan.

3. HASIL DAN PEMBAHASAN

Dalam pembahasan hasil uji sistem ini, kami akan merinci temuan dan evaluasi dari pengujian *blackbox testing* yang telah dilakukan pada Sistem Informasi Forum Komunitas Pecinta Kucing. Pengujian ini bertujuan untuk menilai kehandalan, keamanan, dan fungsionalitas sistem berbasis *web* yang telah dirancang dan dikembangkan di Bandar Lampung. Dengan pendekatan *blackbox testing*, kami dapat mengidentifikasi potensi risiko keamanan, menguji fungsionalitas, serta mengevaluasi respons sistem dari sudut pandang pengguna.

3.1 Halaman Login Admin

Halaman Login Admin bertujuan untuk mengamankan akses ke sistem dengan memvalidasi email dan password admin. Pengguna akan diarahkan ke Halaman Beranda jika kredensial yang dimasukkan valid, sedangkan peringatan akan muncul jika terdapat kesalahan. Hasil evaluasi ditunjukkan pada Gambar 2.



The image shows a web browser window displaying the 'CatLover' Admin Login page. The page has a green header with 'FAQs | Help | Support' and social media icons. Below the header is a dark navigation bar with links for 'Home', 'Informasi Perawatan', 'Adopsi', 'Pendaftaran', and 'Login'. The main content area is titled 'LOGIN' and contains two input fields: 'Username' and 'Password'. Below the fields are 'Login' and 'Reset' buttons. The page also includes contact information: 'Opening Hours: 8:00AM - 9:00PM', 'Email Us: info@example.com', and 'Call Us: +012 345 6789'.

Gambar 2 Halaman Login Admin

Adapun hasil uji dijabarkan pada Tabel 1.

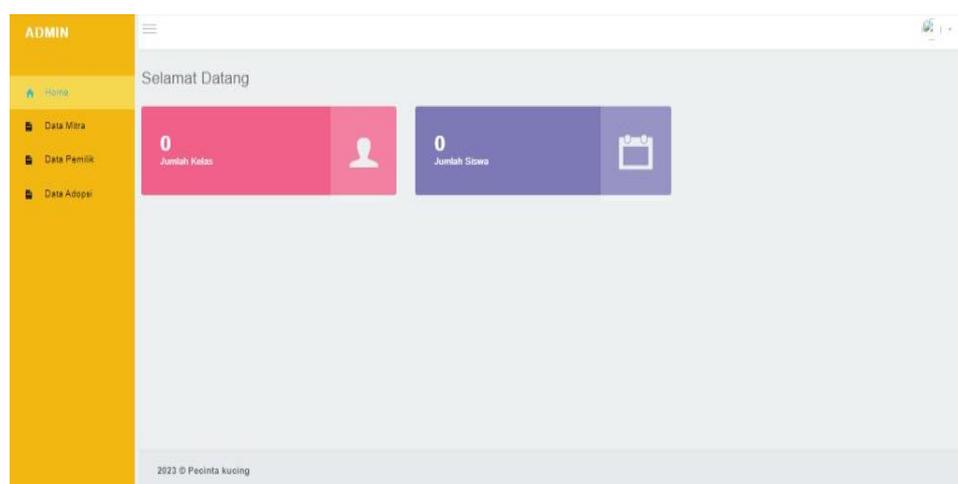
Tabel 1 Hasil Pengujian Pada Halaman Login Admin

No	Skenario Uji	Input	Output yang Diinginkan	Hasil Uji
1	Memasukkan email dan password admin yang valid	Email: admin@example.com, Password: admin123	Redirect ke Halaman Beranda	<i>Valid</i>
2	Memasukkan email yang tidak terdaftar	Email: salah@email.com, Password: admin123	Peringatan email/password salah	<i>Valid</i>
3	Memasukkan password yang salah	Email: admin@example.com, Password: salah123	Peringatan email/password salah	<i>Valid</i>
4	Mencoba login tanpa mengisi email dan password	Email: , Password:	Peringatan mengisi email dan password	<i>Valid</i>
5	Mencoba login hanya dengan mengisi email	Email: admin@example.com, Password:	Peringatan mengisi password	<i>Valid</i>
6	Mencoba login hanya dengan mengisi password	Email: , Password: admin123	Peringatan mengisi email	<i>Valid</i>

Dari hasil pengujian, Halaman *Login Admin* berfungsi dengan baik untuk validasi kredensial. Meskipun demikian, perlu diperhatikan pesan peringatan yang lebih informatif dan pengelolaan kasus-kasus yang tidak terduga. Kami menyarankan untuk meningkatkan keamanan dan pengalaman pengguna adalah mempertimbangkan penerapan kebijakan retensi kata sandi yang kuat dan menyediakan panduan login yang lebih jelas.

3.2 Halaman Beranda Admin

Halaman ini merupakan halaman utama pada sistem, dimana halaman ini hanya bisa diakses oleh pengguna dengan level halaman *admin*. Kami ingi menguji apakah halaman ini dapat diakses pengguna dan non pengguna pada sistem. Hasil evaluasi ditunjukkan pada Gambar 3.



Gambar 3. Halaman Beranda *admin*

Skenario evaluasi ditunjukkan pada Tabel 2.

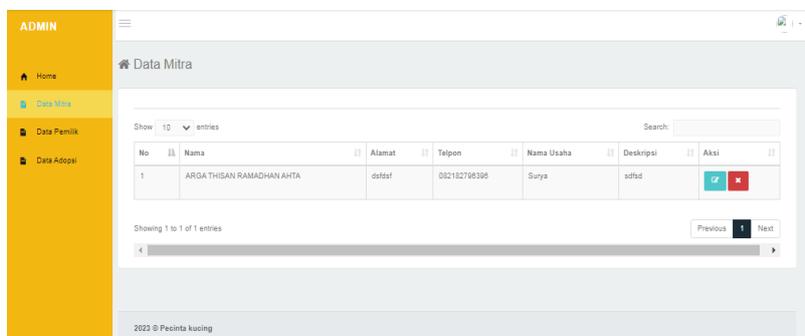
Tabel 2 Hasil Pengujian Pada Halaman Beranda *Admin*

No	Skenario Uji	Input	Output yang Diinginkan	Hasil Uji
1	Menampilkan halaman beranda setelah login	-	Halaman Beranda Dapat Diakses Admin	<i>Valid</i>
2	Menampilkan halaman beranda setelah login dengan email non-Admin	-	Halaman Beranda Tidak Dapat Diakses	<i>Valid</i>
3	Menyalin link halaman admin ke tab <i>browser</i> baru untuk melewati pemeriksaan login	-	Halaman Beranda Tidak Dapat Diakses dan dikembalikan ke Halaman login	<i>Valid</i>

Dari hasil pengujian, halaman beranda telah terlindungi dengan baik dan dapat mencegah akses masuk yang tidak diinginkan dengan penggunaan *user-validation* yang baik. Namun sistem perlu memiliki *pop-up warning* jika terjadi percobaan terhadap percobaan masuk ke sistem *admin* yang tidak sah.

3.3 Halaman Input Data Mitra

Halaman ini merupakan fitur untuk menambahkan data mitra yang bekerjasama dengan komunitas pecinta kucing. Pada pengujian ini difokuskan kepada pengujian *input*, *update*, dan *delete* data. Sistem ditunjukkan pada Gambar 4.



Gambar 4. Halaman *Input Data Mitra*

Hasil skenario evaluasi ditunjukkan pada Tabel 3.

Tabel 3 Hasil Pengujian Pada Fitur Mitra

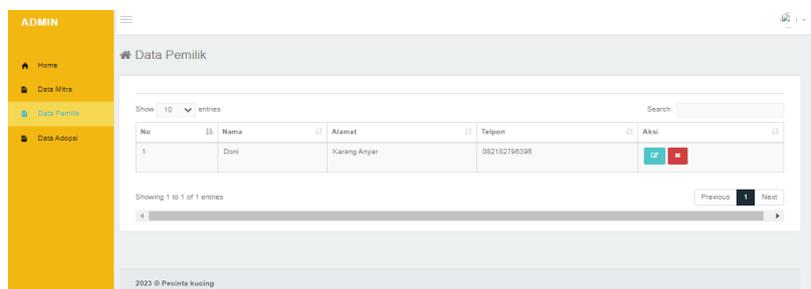
No	Skenario Uji	Input	Output yang Diinginkan	Hasil Uji
1	Melakukan <i>input data</i>	Klik masukkan <i>data</i> mitra	Data masuk kedalam <i>Database</i>	Tidak <i>Valid</i>
2	Melakukan <i>delete data</i>	Hapus <i>data</i> mitra	Data Terhapus dari <i>Database</i>	<i>Valid</i>
3	Melakukan <i>update data</i>	Perbaharui <i>data</i> mitra	Data teperbaharui dari <i>dataabse</i>	<i>Valid</i>

4	Kolom Pencarian	Masukkan pencarian <i>data</i> mitra	Pencarian berhasil	<i>valid</i>
---	-----------------	--------------------------------------	--------------------	--------------

Pada hasil uji ini terdapat kesalahan fitur pada *data* mitra, dimana pada fitur ini tidak terdapat tombol untuk melakukan *input data* untuk admin melakukan penambahan data. Kami menyarankan agar fitur ini dapat ditambahkan agar memudahkan admin melakukan manajemen mitra yang bekerjasama dengan komunitasnya.

3.4 Halaman Data Anggota

Halaman ini merupakan fitur untuk menambahkan data anggota yang yang tergabung pada komunitas pencinta kucing. Pada pengujian ini kami melakukan pengujian pada *input*, *update*, dan *delete* data anggota. Gambaran sistem uji ditunjukkan pada Gambar 5.



Gambar 5. Halaman *Input* Data Anggota

Hasil pengujian ditunjukkan pada Tabel 4.

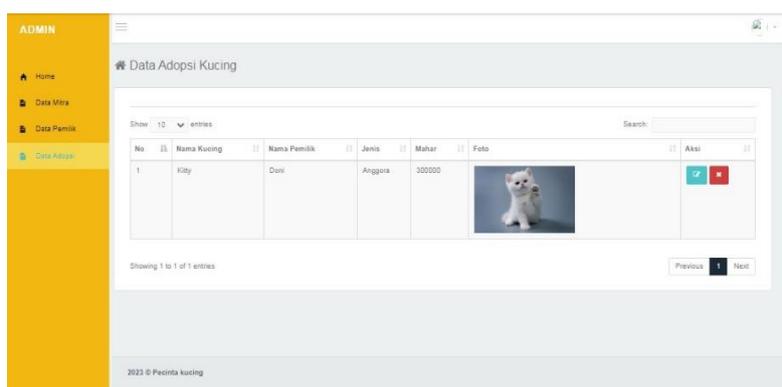
Tabel 4 Hasil Pengujian Pada Fitur Anggota

No	Skenario Uji	Input	Output yang Diinginkan	Hasil Uji
1	Melakukan <i>input data</i> Anggota	Klik masukkan <i>data</i> anggota	Data masuk kedalam <i>Database</i>	Tidak <i>Valid</i>
2	Melakukan <i>delete data</i> Anggota	Hapus <i>data</i> anggota	Data Terhapus dari <i>Database</i>	<i>Valid</i>
3	Melakukan <i>update data</i> Anggota	Perbaharui <i>data</i> anggota	Data teperbaharui dari <i>dataabase</i>	<i>Valid</i>
4	Kolom Pencarian Anggota	Masukkan pencarian <i>data</i> anggota	Pencarian berhasil	<i>valid</i>

Pada hasil uji ini tidak ditemukan tombol tau fitur tambah data anggota dari halaman *admin*, sehingga *admin* tidak dapat menambahkan data anggota baru secara manual. Pendaftaran anggota baru di sisi admin adalah hal yang penting, jika suatu saat terjadi kesalahan sistem pendaftaran anggota baru di bagian halaman utama sistem, sehingga kami menyarankan untuk menambahkan fitur tambahan fitur anggota.

3.5 Halaman Data Adopsi

Halaman ini merupakan halaman untuk melakukan manajemen data anggota yang melakukan adopsi terhadap kucing-kucing yang tersedia. Pada pengujian ini kami melakukan uji terhadap *input*, *delete*, dan *update* data, serta kolom pencarian nama-nama adaptor yang terekam dalam sistem seperti pada Gambar 6.



Gambar 6 Data Adopsi

Skenario hasil evaluasi ditunjukkan pada Tabel 5.

Tabel 5 Hasil Pengujian Pada Fitur Adopsi

No	Skenario Uji	Input	Output yang Diinginkan	Hasil Uji
1	Melakukan <i>input data adopter</i>	Klik masukkan <i>data adopter</i>	Data masuk kedalam <i>Database</i>	Tidak <i>Valid</i>
2	Melakukan <i>delete data adopter</i>	Hapus <i>data adopter</i>	Data Terhapus dari <i>Database</i>	<i>Valid</i>
3	Melakukan <i>update data adopter</i>	Perbaharui <i>data adopter</i>	Data teperbaharui dari <i>dataabase</i>	<i>Valid</i>
4	Kolom Pencarian <i>adopter</i>	Masukkan pencarian <i>adopter</i>	Pencarian berhasil	<i>valid</i>

Sama dengan masalah sebelumnya, pengembang sistem tidak memasukan tombol untuk menambah data *adopter* dari sisi admin, sehingga *admin* tidak dapat menambahkan data *adopter* baru jika dibutuhkan, sehingga penambahan data *adopter* murni dari halaman depan *website*. Kami menyarankan agar ditambahkan fungsi menambahkan data *adopter* dari halaman admin.

3.5 Hasil Pengujian

Untuk menghitung persentase validitas dan non-validitas yang kami temukan pada sistem ini, kami menggunakan rumus sebagai berikut:

$$\text{persentasi valid} = \left(\frac{\text{jumlah skenario valid}}{\text{total skenario uji}} \right) \times 100. \quad (1)$$

$$\text{persentasi tidak valid} = \left(\frac{\text{jumlah skenario tidak valid}}{\text{total skenario uji}} \right) \times 100. \quad (2)$$

Sehingga berdasarkan hasil uji sistem pada sistem informasi forum komunitas pecinta kucing, kami dapat menyimpulkan bahwa sebanyak 75% dari total 12 skenario uji berhasil melewati pengujian dengan hasil *valid*. Namun, sebesar 25% skenario menghasilkan ketidakvalidan dalam respons sistem.

4. KESIMPULAN

Berdasarkan penelitian ini hasil uji pada sistem informasi forum komunitas pecinta kucing, didapat kesimpulan sebagai berikut:

- Dari 12 skenario uji menggunakan blackbox testing, 75% berhasil melewati pengujian dan menghasilkan respons sistem yang valid
- Sebanyak 25% skenario menghasilkan ketidakvalidan dalam respons sistem.
- Keberhasilan 75% skenario valid menunjukkan sebagian besar fungsi dan keamanan sistem beroperasi sesuai harapan.
- Temuan ketidakvalidan pada 25% skenario menyoroati potensi perbaikan dan peningkatan yang perlu diperhatikan.

5. SARAN

Selanjutnya kami menyarankan beberapa perbaikan dan penelitian kedepan terkait pengujian sistem informasi.

- Pengujian masih bisa dilakukan dengan menguji bagian *internal* pada sistem
- Pengujian ini masih terbatas pada pengalaman pengguna terhadap sistem, pengujian bisa berlanjut kepada pengujian ketahanan sistem terhadap serangan injeksi untuk dapat masuk kedalam sistem secara illegal.
- Pengujian masih bisa dilanjutkan pada pengujian di bagian halaman depan sistem.
- Pengujian juga bisa mencakup internal sistem dengan *whitebox testing*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Ananda Muhammad Rifki Qasthalani Mahasiswa Sistem Informasi Institut Informatika dan Bisnis Darmajaya yang telah membangun sistem informasi komunitas secara baik dan mengizinkan tim untuk melakukan pengujian pada sistem yang telah dikembangkan, penulis juga mengucapkan terimakasih kepada Tim Redaksi Jurnal Teknik Politeknik Negeri Sriwijaya yang telah memberi kesempatan, sehingga artikel ilmiah ini dapat diterbitkan.

DAFTAR PUSTAKA

- [1] D. Yuliawati, A. Andriyadi, and N. Nursiyanto, "Pengujian Sistem Informasi E-Monitoring Pengelolaan Pembangunan Desa Dengan Menggunakan Metode Blackbox Testing," *TEKNIKA: Jurnal Ilmiah Bidang Ilmu Rekayasa*, vol. 16, no. 2, pp. 303-310–303 – 310, Nov. 2022, doi: 10.5281/ZENODO.7535944.
- [2] D. Wintana, D. Pribadi, and M. Y. Nurhadi, "Analisis Perbandingan Efektifitas White-Box Testing dan Black-Box Testing," *Jurnal Larik: Ladang Artikel Ilmu Komputer*, vol. 2, no. 1, pp. 8–16, Jul. 2022, Accessed: Nov. 02, 2022. [Online]. Available: <http://103.75.24.116/index.php/larik/article/view/1382>
- [3] Mohd. E. Khan, "Different Approaches to Black Box Testing Technique for Finding Errors." Jul. 21, 2021. Accessed: Oct. 28, 2022. [Online]. Available: <https://papers.ssrn.com/abstract=3890672>
- [4] T. Murnane and K. Reed, "On the effectiveness of mutation analysis as a black box testing technique," *Proceedings of the Australian Software Engineering Conference, ASWEC*, vol. 2001-January, pp. 12–20, 2001, doi: 10.1109/ASWEC.2001.948492.
- [5] E. Sonalitha, B. Nurdewanto, A. Zubair, S. R. Asriningtias, K. Yudhistiro, and I. Mujahidin, "Blackbox Testing Model Boundary Value of Mapping Taxonomy Applications and Data Analysis

- of Art and Artworks,” *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2020*, pp. 7–11, Dec. 2020, doi: 10.1109/ISRITI51436.2020.9315406.
- [6] R. Thapa, D. Ma, and X. Jiao, “HDXplore: Automated Blackbox Testing of Brain-Inspired Hyperdimensional Computing,” *Proceedings of IEEE Computer Society Annual Symposium on VLSI, ISVLSI*, vol. 2021, pp. 90–95, Jul. 2021, doi: 10.1109/ISVLSI51109.2021.00027.
- [7] E. Sonalitha, B. Nurdewanto, A. Zubair, S. R. Asriningtias, K. Yudhistiro, and I. Mujahidin, “Blackbox Testing Model Boundary Value of Mapping Taxonomy Applications and Data Analysis of Art and Artworks,” *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2020*, pp. 7–11, Dec. 2020, doi: 10.1109/ISRITI51436.2020.9315406.
- [8] C. H. Tsai, S. C. Tsai, and S. K. Huang, “REST API Fuzzing by Coverage Level Guided Blackbox Testing,” *IEEE International Conference on Software Quality, Reliability and Security, QRS*, vol. 2021, pp. 291–300, Jan. 2021, doi: 10.1109/QRS54544.2021.00040.
- [9] H. Maulana, N. Marcheta, A. T. Muharram, K. R. Permana, and A. P. Aisyah, “Design and Build a Attendance System and Employee Performance Assessment with a Website-Based Profile Matching Method,” *2022 7th International Conference on Informatics and Computing, ICIC 2022*, p. Online, doi: 10.1109/ICIC56845.2022.10006914.
- [10] H. Supriyono, R. F. Rahmadzani, and M. S. Adhantoro, “Developing computer application for interactive javanese letters learning,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 3112–3119, 2019, doi: 10.30534/ijatcse/2019/72862019.
- [11] D. Yuliawati, A. Andriyadi, and N. Nursiyanto, “Penguajian Sistem Informasi E-Monitoring Pengelolaan Pembangunan Desa Dengan Menggunakan Metode Blackbox Testing,” *TEKNIKA*, vol. 16, no. 2, pp. 303 – 310–303 – 310, Nov. 2022, doi: 10.5281/ZENODO.7535944.
- [12] J. Cao, Y. Zhou, C. Sun, L. He, Z. Xi, and Y. Liu, “Firebolt: Finding Bugs in Programmable Data Plane Generators,” *Proceedings of the 2022 USENIX Annual Technical Conference, ATC 2022*, pp. 819–834, Jan. 2022.
- [13] S. Limanto, F. D. Kartikasari, and M. Oeitheurisa, “Improved Learning Outcomes of Descriptive Statistics through the Test Room and Data Processing Features in the Mobile Learning Model,” *Proceeding - 2020 2nd International Conference on Industrial Electrical and Electronics, ICIEE 2020*, pp. 139–142, Oct. 2020, doi: 10.1109/ICIEE49813.2020.9277408.
- [14] F. Akbar, S. Tirozi, and H. Suryamen, “Web-based mapping of electric customer distribution of pln sub-district sawahlunto,” *2020 International Conference on Information Technology Systems and Innovation, ICITSI 2020 - Proceedings*, pp. 204–210, Oct. 2020, doi: 10.1109/ICITSI50517.2020.9264975.
- [15] Noerlina, A. Chandra, and T. N. Mursitama, “Sales Revenue Sharing Model using Dynamics NAV Modification in Health Industries,” *IOP Conf Ser Earth Environ Sci*, vol. 426, no. 1, p. 012162, Mar. 2020, doi: 10.1088/1755-1315/426/1/012162.
- [16] A. Widyanto, D. I. Sensuse, and Opitasari, “Web-based knowledge management system design by applying the becerra-fernandez knowledge management process approach: Case study at pt yafii solusi internasional,” *International Journal of Scientific and Technology Research*, vol. 8, no. 10, pp. 287–294, Oct. 2019.
- [17] S. Kannan and K. T. Tian, “Locating errors in faulty formulas,” *ACM Transactions on Algorithms*, vol. 15, no. 3, p. 0079, May 2019, doi: 10.1145/3313776.