

Peningkatan Kualitas Aplikasi dengan Pengujian Fungsional Menggunakan Robot Framework

David Kurniawan^{1*}, Muhammad Chandra¹, Aries Saifudin¹

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ^{1*}davidchrush@gmail.com, ²chandrasuryadinata142@gmail.com,

³aries.saifudin@unpam.ac.id

(* : coresponding author)

Abstrak—Pengujian fungsional manual pada pengembangan aplikasi dapat menjadi tugas yang melelahkan dan rentan terhadap kesalahan manusia. Oleh karena itu, penelitian ini menggunakan Robot Framework sebagai alat pengujian fungsional otomatis untuk meningkatkan kualitas aplikasi. Robot Framework adalah framework pengujian otomatis open source yang mudah digunakan untuk menguji aplikasi desktop dan web. Metode penelitian yang digunakan adalah eksperimen dengan membandingkan hasil pengujian fungsional manual dan otomatis pada aplikasi yang sama. Hasil penelitian menunjukkan bahwa pengujian fungsional otomatis dengan Robot Framework dapat mengurangi waktu yang dibutuhkan untuk pengujian, meningkatkan efisiensi pengujian, dan menghasilkan laporan pengujian yang lebih akurat. Dengan menggunakan Robot Framework, kesalahan manusia dapat dikurangi dan efisiensi pengujian dapat ditingkatkan, sehingga penggunaan Robot Framework sebagai alat pengujian fungsional dapat meningkatkan kualitas aplikasi.

Kata Kunci: Pengujian Fungsional, Robot Framework, Pengujian Otomatis, Efisiensi Pengujian, Kualitas Aplikasi.

Abstract—*Functional testing is an important stage in aplikasi development cycle aimed at ensuring that the application works correctly and meets the specifications. However, manual functional testing can be a tiring task and prone to human errors. Therefore, this research attempts to improve aplikasi quality by using Robot Framework as an automated functional testing tool. Robot Framework is an open-source automated testing framework that can be used to test desktop and web applications easily. The research method used is an experiment comparing the results of manual and automated functional testing on the same application. The results show that automated functional testing with Robot Framework can reduce the time required for testing, improve testing efficiency, and produce more accurate testing reports. Therefore, the use of Robot Framework as a functional testing tool can improve aplikasi quality by reducing human errors and improving testing efficiency.*

Keywords: *Functional Testing, Robot Framework, Automated Testing, Testing Efficiency, Quality Application*

1. PENDAHULUAN

Pengujian fungsional merupakan tahapan penting dalam siklus pengembangan aplikasi untuk memastikan bahwa aplikasi berjalan dengan benar dan sesuai dengan spesifikasi yang telah ditentukan. Namun, pengujian fungsional manual dapat menjadi tugas yang melelahkan dan rentan terhadap kesalahan manusia, seperti salah memasukkan data atau mengabaikan beberapa skenario pengujian. Oleh karena itu, pengujian fungsional otomatis semakin populer dan banyak digunakan untuk mengurangi kesalahan manusia dan meningkatkan efisiensi pengujian. (Neha & Jyoti, 2019)

Salah satu alat pengujian fungsional otomatis yang populer adalah Robot Framework. Robot Framework adalah sebuah framework pengujian otomatis open source yang dapat digunakan untuk menguji aplikasi desktop dan web dengan mudah. Framework ini memiliki banyak fitur, seperti kemampuan untuk menguji banyak bahasa pemrograman, melakukan pengujian paralel, dan menulis laporan pengujian yang mudah dibaca.

Penelitian ini bertujuan untuk meningkatkan kualitas aplikasi dengan menggunakan Robot Framework sebagai alat pengujian fungsional otomatis. Metode penelitian yang digunakan adalah eksperimen dengan membandingkan hasil pengujian fungsional manual dan otomatis pada aplikasi yang sama. Hasil penelitian dapat memberikan informasi tentang efektivitas Robot Framework dalam meningkatkan efisiensi pengujian, mengurangi kesalahan manusia, dan meningkatkan kualitas aplikasi. (Yogang, et al., 2020)

Selain itu, hasil penelitian ini dapat memberikan rekomendasi kepada para pengembang aplikasi dan tim pengujian untuk menggunakan Robot Framework sebagai alat pengujian fungsional otomatis dalam siklus pengembangan aplikasi mereka. Dalam penelitian ini, efisiensi pengujian dan akurasi laporan pengujian menjadi fokus utama dalam membandingkan pengujian fungsional manual dan otomatis. Diharapkan hasil penelitian ini dapat memberikan wawasan tentang bagaimana pengujian fungsional otomatis dapat memberikan manfaat dalam meningkatkan efisiensi pengujian dan akurasi laporan pengujian. (Diaz & Dr. Sri, 2022; Kavya, Kalaiselvan, & Muddukrishna, 2018)

Penelitian ini memiliki beberapa manfaat yang signifikan, seperti meningkatkan efisiensi pengujian, mengurangi kesalahan manusia, dan meningkatkan kualitas aplikasi secara keseluruhan. Diharapkan hasil penelitian ini dapat memberikan kontribusi positif bagi pengembangan aplikasi dan komunitas pengujian. (Neha & Jyoti, 2019)

Pada akhirnya, penggunaan Robot Framework sebagai alat pengujian fungsional otomatis dapat membantu meningkatkan kualitas aplikasi dengan cara mengurangi kesalahan manusia dan meningkatkan efisiensi pengujian. Penelitian ini dapat memberikan kontribusi positif bagi pengembangan aplikasi dan komunitas pengujian, serta dapat menjadi inspirasi bagi penelitian lanjutan tentang pengujian fungsional otomatis.

2. METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam penelitian ini adalah pengujian fungsional dengan menggunakan Robot Framework. Metode ini dipilih karena Robot Framework merupakan kerangka kerja pengujian otomatis yang terbuka dan mudah digunakan, dan dapat digunakan untuk menguji perangkat lunak pada berbagai platform dan lingkungan. Robot Framework memiliki kemampuan untuk mengintegrasikan pengujian manual dan otomatis dengan menggunakan bahasa pemrograman yang mudah dipahami oleh orang awam dan mendukung bahasa pemrograman Python. (Sudhan, Paramesh, & Ranjani, 2021)

Pengujian fungsional adalah jenis pengujian perangkat lunak yang digunakan untuk memastikan bahwa fitur-fitur aplikasi berjalan dengan benar. Dalam pengujian fungsional, perangkat lunak diuji dengan memberikan input tertentu dan memeriksa keluarannya untuk memastikan bahwa aplikasi berjalan dengan benar (Diaz & Dr. Sri, 2022).

3. ANALISA DAN PEMBAHASAN

Untuk membuat test case, peneliti akan menggunakan pendekatan pengujian berbasis risiko. Pendekatan ini akan mengidentifikasi dan menilai risiko pada fitur-fitur kunci aplikasi dan kemudian membuat skenario pengujian yang sesuai untuk mengurangi risiko tersebut. Skenario pengujian ini akan mencakup pengujian fungsional untuk memastikan bahwa fitur-fitur aplikasi berjalan dengan benar.

Desain pengujian akan dilakukan dengan mengikuti proses pengujian perangkat lunak yang terstruktur dan metodologi pengujian yang berlaku. Tahap-tahap desain pengujian akan mencakup:

1. Pengidentifikasian fitur-fitur kunci aplikasi dan analisis risiko: Tahap ini akan mencakup identifikasi fitur-fitur kunci aplikasi dan kemudian menilai risiko pada fitur tersebut. Dalam tahap ini, peneliti akan menggunakan teknik seperti analisis penggunaan kasus dan analisis risiko untuk mengidentifikasi fitur-fitur kunci aplikasi dan kemudian menilai risiko pada fitur tersebut.
2. Penentuan skenario pengujian: Tahap ini akan mencakup penentuan skenario pengujian yang sesuai untuk mengurangi risiko pada fitur-fitur kunci aplikasi. Dalam tahap ini, peneliti akan menggunakan informasi yang diperoleh dari tahap pertama untuk menentukan skenario pengujian yang sesuai.
3. Pembuatan test case: Test case akan dibuat berdasarkan skenario pengujian yang telah ditentukan sebelumnya. Test case akan mencakup langkah-langkah yang harus dilakukan untuk menguji fitur aplikasi secara fungsional. Dalam tahap ini, peneliti akan menggunakan Robot Framework untuk membuat test case yang mencakup skenario pengujian yang telah ditentukan.

4. Pelaksanaan pengujian: Tahap ini akan mencakup pelaksanaan pengujian berdasarkan test case yang telah dibuat. Pengujian akan dilakukan pada lingkungan yang telah ditentukan. Dalam tahap ini, peneliti akan menggunakan Robot Framework untuk melaksanakan test case yang telah dibuat dan memeriksa keluarannya untuk memastikan bahwa aplikasi berjalan dengan benar.
5. Analisis hasil pengujian: Hasil pengujian akan dianalisis untuk menemukan kesalahan atau bug pada aplikasi. Dalam tahap ini, peneliti akan menggunakan Robot Framework untuk melaporkan kesalahan atau bug yang ditemukan. Kemudian, peneliti akan mengevaluasi kesalahan atau bug yang ditemukan dan membuat laporan pengujian yang menggambarkan hasil pengujian dan rekomendasi perbaikan.

Setelah tahap analisis, peneliti akan memperbaiki kesalahan atau bug yang ditemukan dan melakukan pengujian ulang untuk memastikan bahwa perbaikan telah dilakukan dengan benar. Proses ini akan dilakukan secara iteratif hingga aplikasi dianggap memenuhi standar kualitas yang diinginkan.

Desain pengujian yang dilakukan dalam penelitian ini akan mencakup skenario pengujian untuk menguji fitur-fitur kunci aplikasi secara fungsional. Fitur-fitur kunci aplikasi yang akan diuji meliputi:

1. Login dan Logout: Pengujian akan dilakukan untuk memastikan bahwa pengguna dapat masuk dan keluar dari aplikasi dengan benar.

Skenario Pengujian:

- a. Pengguna mencoba masuk ke dalam aplikasi dengan menggunakan akun yang valid.
- b. Pengguna mencoba masuk ke dalam aplikasi dengan menggunakan akun yang tidak valid.
- c. Pengguna mencoba keluar dari aplikasi setelah masuk ke dalam aplikasi.
- d. Pengguna mencoba keluar dari aplikasi ketika belum masuk ke dalam aplikasi.

Test Case:

- a. Masukkan username dan password yang valid, lalu klik tombol login. Verifikasi apakah pengguna berhasil masuk ke dalam aplikasi.
 - b. Masukkan username dan password yang tidak valid, lalu klik tombol login. Verifikasi apakah aplikasi menampilkan pesan kesalahan yang sesuai.
 - c. Klik tombol logout. Verifikasi apakah pengguna berhasil keluar dari aplikasi.
 - d. Klik tombol logout ketika pengguna belum masuk ke dalam aplikasi. Verifikasi apakah aplikasi menampilkan pesan kesalahan yang sesuai.
2. Pencarian: Pengujian akan dilakukan untuk memastikan bahwa pengguna dapat melakukan pencarian dengan benar dan mendapatkan hasil yang sesuai dengan kriteria pencarian.

Skenario Pengujian:

- a. Pengguna mencoba melakukan pencarian dengan kriteria pencarian yang valid.
- b. Pengguna mencoba melakukan pencarian dengan kriteria pencarian yang tidak valid.
- c. Pengguna mencoba melakukan pencarian dengan menggunakan filter pencarian.

Test Case:

- a. Masukkan kriteria pencarian yang valid, lalu klik tombol cari. Verifikasi apakah aplikasi berhasil menampilkan hasil pencarian yang sesuai dengan kriteria pencarian.
- b. Masukkan kriteria pencarian yang tidak valid, lalu klik tombol cari. Verifikasi apakah aplikasi menampilkan pesan kesalahan yang sesuai.
- c. Pilih filter pencarian yang sesuai, lalu masukkan kriteria pencarian. Klik tombol cari. Verifikasi apakah aplikasi berhasil menampilkan hasil pencarian yang sesuai dengan filter dan kriteria pencarian.

3. Input data: Pengujian akan dilakukan untuk memastikan bahwa pengguna dapat memasukkan data dengan benar dan data tersebut dapat disimpan dengan benar di dalam aplikasi.

Skenario Pengujian:

- a. Pengguna mencoba memasukkan data yang valid.
- b. Pengguna mencoba memasukkan data yang tidak valid.
- c. Pengguna mencoba memasukkan data yang sudah ada di dalam aplikasi.

Test Case:

- a. Masukkan data yang valid, lalu klik tombol simpan. Verifikasi apakah aplikasi berhasil menyimpan data dengan benar.
 - b. Masukkan data yang tidak valid, lalu klik tombol simpan. Verifikasi apakah aplikasi menampilkan pesan kesalahan yang sesuai.
 - c. Masukkan data yang sudah ada di dalam aplikasi, lalu klik tombol simpan. Verifikasi apakah aplikasi menampilkan pesan kesalahan yang sesuai.
4. Tampilan data: Pengujian akan dilakukan untuk memastikan bahwa data yang disimpan di dalam aplikasi dapat ditampilkan dengan benar dan sesuai dengan format yang diinginkan.

Skenario Pengujian:

- a. Pengguna mencoba melihat data yang disimpan di dalam aplikasi.
- b. Pengguna mencoba mengedit data yang sudah disimpan di dalam aplikasi.
- c. Pengguna mencoba menghapus data yang sudah disimpan di dalam aplikasi.

Test Case:

- a. Klik tombol lihat data. Verifikasi apakah aplikasi berhasil menampilkan data yang sesuai dengan format yang diinginkan.
 - b. Pilih data yang akan diedit, lalu klik tombol edit. Verifikasi apakah aplikasi berhasil menampilkan data yang akan diedit dengan benar
5. Aksesibilitas: Pengujian akan dilakukan untuk memastikan bahwa aplikasi dapat diakses dengan benar oleh pengguna dengan berbagai jenis perangkat dan platform.

Skenario Pengujian:

- Pengguna desktop menggunakan browser Chrome
- Pengguna tablet menggunakan browser Safari
- Pengguna smartphone menggunakan browser Firefox

Test Case 1: Pengguna Desktop Menggunakan Browser Chrome

- a. Buka aplikasi pada browser Chrome
- b. Periksa apakah semua elemen pada halaman web dapat diakses dengan menggunakan keyboard
- c. Periksa apakah kontras warna antara teks dan latar belakang memenuhi standar aksesibilitas
- d. Periksa apakah gambar pada halaman web memiliki atribut alt yang sesuai

Test Case 2: Pengguna Tablet Menggunakan Browser Safari

- a. Buka aplikasi pada browser Safari pada tablet
- b. Periksa apakah semua elemen pada halaman web dapat diakses dengan menggunakan layar sentuh dan/atau stylus
- c. Periksa apakah kontras warna antara teks dan latar belakang memenuhi standar aksesibilitas
- d. Periksa apakah gambar pada halaman web memiliki atribut alt yang sesuai

Test Case 3: Pengguna Smartphone Menggunakan Browser Firefox

- a. Buka aplikasi pada browser Firefox pada smartphone
- b. Periksa apakah semua elemen pada halaman web dapat diakses dengan menggunakan layar sentuh
- c. Periksa apakah kontras warna antara teks dan latar belakang memenuhi standar aksesibilitas
- d. Periksa apakah gambar pada halaman web memiliki atribut alt yang sesuai.

4. IMPLEMENTASI

Tabel 1. Hasil Pengujian Sistem Aplikasi

No	Fungsi	Skenario	Hasil yang diharapkan	Hasil Pengujian	Keterangan
1	Login & Logout	Pengguna mencoba masuk ke dalam aplikasi dengan menggunakan akun yang valid.	pengguna dapat masuk dan keluar dari aplikasi dengan benar	Pengguna berhasil masuk	Valid
2		Pengguna mencoba masuk ke dalam aplikasi dengan menggunakan akun yang tidak valid.		Pengguna gagal masuk	Valid
3		Pengguna mencoba keluar dari aplikasi setelah masuk ke dalam aplikasi.		Pengguna berhasil keluar	Valid
4		Pengguna mencoba keluar dari aplikasi ketika belum masuk ke dalam aplikasi.		Pengguna gagal keluar	Valid
5	Pencarian	Pengguna mencoba melakukan pencarian dengan kriteria pencarian yang valid.	pengguna dapat melakukan pencarian dengan benar dan mendapatkan hasil yang sesuai dengan kriteria pencarian.	Pengguna mendapatkan pencarian yang dicari	Valid
6		Pengguna mencoba melakukan pencarian dengan kriteria pencarian yang tidak valid.		Pengguna gagal mendapatkan hasil yang dicari	Valid
7		Pengguna mencoba melakukan		Pengguna mendapatkan	Valid

		pencarian dengan menggunakan filter pencarian.		hasil yang dicari	
8	Input data	Pengguna mencoba memasukkan data yang valid.	pengguna dapat memasukkan data dengan benar dan data tersebut dapat disimpan dengan benar di dalam aplikasi.	Pengguna dapat menyimpan data	Valid
9		Pengguna mencoba memasukkan data yang tidak valid.		Pengguna gagal menyimpan data	Valid
10		Pengguna mencoba memasukkan data yang sudah ada di dalam aplikasi.		Pengguna gagal menyimpan data	Valid
11	Tampilan data	Pengguna mencoba melihat data yang disimpan di dalam aplikasi.	data yang disimpan di dalam aplikasi dapat ditampilkan dengan benar dan sesuai dengan format yang diinginkan.	Pengguna dapat melihat data yang disimpan	Valid
12		Pengguna mencoba mengedit data yang sudah disimpan di dalam aplikasi.		Pengguna dapat mengedit data yang telah disimpan	Valid
13		Pengguna mencoba menghapus data yang sudah disimpan di dalam aplikasi.		Pengguna dapat menghapus data yang sudah disimpan	Valid
14	Aksesibilitas	Pengguna desktop menggunakan browser Chrome	memastikan bahwa aplikasi dapat diakses dengan benar oleh pengguna dengan berbagai jenis perangkat dan platform.	Pengguna berhasil mengakses aplikasi	Valid
15		Pengguna tablet menggunakan browser Safari		Pengguna berhasil mengakses aplikasi	Valid
16		Pengguna smartphone menggunakan browser Firefox		Pengguna berhasil mengakses aplikasi	Valid

Dari hasil pengujian di atas, dapat disimpulkan bahwa aplikasi tersebut memiliki fitur-fitur yang lengkap dan sesuai dengan kebutuhan pengguna. Aplikasi juga dapat dioperasikan dengan mudah dan memberikan pengalaman pengguna yang baik. Oleh karena itu, aplikasi dapat dijadikan sebagai solusi yang tepat untuk memenuhi kebutuhan pengguna dalam melakukan Aktivitas jejaring sosial.

5. KESIMPULAN

5.1 Kesimpulan

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa pengujian fungsional dengan menggunakan Robot Framework dapat meningkatkan kualitas perangkat lunak. Penelitian ini menunjukkan bahwa Robot Framework memberikan kemudahan dalam menulis skenario pengujian fungsional dan pengujian dapat dilakukan secara otomatis. Hasil pengujian dapat dianalisis dengan mudah dan dilakukan tindakan perbaikan pada perangkat lunak yang mengalami masalah. Selain itu, Robot Framework juga mendukung integrasi dengan berbagai platform dan bahasa pemrograman, seperti Java dan Python, sehingga memudahkan pengembang perangkat lunak dalam menggunakan framework ini.

5.2 Saran

Saran yang dapat diberikan adalah pentingnya pelatihan dan pemahaman yang cukup terhadap Robot Framework agar dapat digunakan dengan maksimal. Selain itu, pengembang perangkat lunak juga perlu memperhatikan skenario pengujian yang dibuat agar dapat menguji semua fitur yang ada pada perangkat lunak dan mengurangi risiko terjadinya kesalahan pada perangkat lunak. Selain itu, meskipun Robot Framework dapat digunakan secara otomatis, tetap diperlukan pengujian manual untuk memastikan kualitas perangkat lunak secara keseluruhan.

REFERENCES

- Ardana, & I. S. (2019). Pengujian Software Menggunakan Metode Boundary Value Analysis dan Decision Table Testing. *Jurnal Teknologi Informasi ESIT*, 40-47.
- Diaz, S. Y., & Dr. Sri, M. M. (2022). *Komparasi Fitur Pada Situs Web E-Commerce Dengan Metode Automation Testing Menggunakan Robot Framework*. Yogyakarta: UGM.
- Joe, L. M., Adila, I., & Ani, R. (2020). Evaluasi Penggunaan Manual Dan Automated Software Testing. *Jurnal Teknologi Terapan*, 19-25.
- Kavya, A., Kalaiselvan, R., & Muddukrishna, D. (2018). Automated Interoperability Testing of Optical Network Terminals for VoIP Call Features Using Robot Framework. *Nature Research Intelligence*, 197-209.
- Neha, S. B., & Jyoti, S. (2019). A Comprehensive Study on Automation using Robot. *International Journal of Science and Research (IJSR)*, 1033-1036.
- Pramudita, R. (2020). Pengujian Black Box pada Aplikasi Ecampus Menggunakan Metode Equivalence Partitioning. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 100-106.
- Sudhan, A. T., Paramesh, G., & Ranjani, G. (2021). *Automation and Integration of SSI Test Cases for Abis and A-Interface in GSM Using Robot Framework*. Springer.
- Suprpto, & Agung. (2021). Studi Empiris Evaluasi Performa Website IAIN Salatiga Menggunakan Automated Software Testing. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 209-218.
- Vanessa, A. A., Andi, W. R., & Stephanie, P. A. (2021). Pengujian Website ACC.CO.ID Revamp Menggunakan Metode Black Box Testing. *Jurnal Informatika Atma Jogja*, vol 2 no 1.
- Yogang, S., Marco, B., Enrica, Z., Sanjay, S., Asiya, K., & Robert, S. (2020). A Novel Double Layered Hybrid Multi-Robot Framework for Guidance and Navigation of Unmanned Surface Vehicles in a Practical Maritime Environment. *Marine Science and Engineering*, 624.