

PENGUJIAN OTOMATIS SISTEM INFORMASI INDO TOWING DENGAN KATALON STUDIO

Mutia¹, Nurul Qomariyah², Dewi Kania Widyawati³

^{1,3} Program Studi Manajemen Informatika, Politeknik Negeri Lampung

² Program Studi Teknologi Rekayasa Internet, Politeknik Negeri Lampung

INFORMASI ARTIKEL

Diterima 26 September 2023

Direvisi 30 September 2023

Diterbitkan 2 Oktober 2023

Kata kunci:

Pengujian;
Software;
Automation Testing;
Equivalence Partition;
Katalon Studio;
STLC

ABSTRAK

Pengembangan sistem informasi berbasis *web* dan *mobile* yang mendukung berbagai aspek pekerjaan manusia telah menjadi fokus utama dalam era teknologi saat ini. Pentingnya *automation testing* dalam memastikan kualitas perangkat lunak menjadi semakin mendalam. Kualitas sistem ini sangat penting untuk mendukung operasi bisnis yang sukses. *Automation testing* dengan metode STLC dan teknik *equivalence partitioning* diterapkan dalam rangka memastikan kualitas sistem Indo Towing. Alat pengujian yang digunakan adalah Katalon Studio, yang memfasilitasi pelaksanaan *automation testing* dengan efisien. Dengan demikian, studi ini menekankan pentingnya *automation testing* sebagai langkah krusial dalam pengembangan perangkat lunak. Hasil pengujian menggunakan Katalon Studio berhasil dilakukan berjalan sesuai dengan fungsi sistem informasi tanpa ditemukan *bug* pada sistem. Pengujian yang dilakukan secara otomatis menggunakan Katalon Studio dapat disimpulkan bahwa sistem sudah berjalan dengan baik.

AUTOMATED TESTING OF INDO TOWING INFORMATION SYSTEM WITH KATALON STUDIO

ARTICLE INFO

Received September 26, 2023

Revised September 30, 2023

Published October 2, 2023

Keyword:

Testing;
Software;
Automation Testing;
Equivalence Partition;
Katalon Studio;
STLC

ABSTRACT

The development of web-based and mobile information systems that support various aspects of human work has become the main focus in the current technological era. The importance of automation testing in ensuring software quality is becoming increasingly profound. The quality of these systems is critical to supporting successful business operations. Automation testing using the STLC method and equivalence partitioning technique is applied to ensure the quality of the Indo Towing system. The testing tool used is Katalon Studio, which facilitates the implementation of automated testing with efficiency. Thus, this study emphasizes the importance of automation testing as a crucial step in software development. The results of the implementation of automation testing using Katalon Studio were successfully carried out according to the information system function without any bugs being found in the system. Tests carried out automatically using Katalon Studio can be concluded that the system is running well.

This work is licensed under a [Creative Commons Attribution 4.0](https://creativecommons.org/licenses/by/4.0/)



Corresponding Author:

Corresponding Author Name, Affiliation

Email: nqomariyah@polinela.ac.id, Program Studi Teknologi Rekayasa Internet, Politeknik Negeri Lampung

1. PENDAHULUAN

Pada era teknologi yang berkembang pesat saat ini, sistem informasi berbasis *web* dan *mobile* telah menjadi bagian integral dari kehidupan sehari-hari manusia. Sistem ini dirancang untuk memberikan kemudahan, fleksibilitas, dan efisiensi dalam berbagai aspek pekerjaan manusia [1]. Namun, dengan kompleksitas pengembangan perangkat lunak yang semakin meningkat, pentingnya *automation testing* juga semakin terasa [2]. *Automation testing* adalah komponen kritis dalam proses pengujian perangkat lunak yang bertujuan untuk mendeteksi kesalahan dan masalah potensial yang dapat muncul selama pengembangan [3].

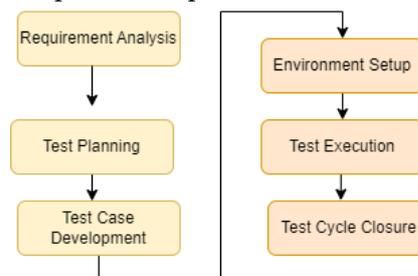
Sistem Informasi Indo Towing adalah sebuah *platform website* yang bertujuan untuk memudahkan pemesanan pelanggan, dengan berbagai fitur seperti pemesanan online dan pengecekan tarif towing di seluruh Indonesia. Namun, keberhasilan platform ini sangat bergantung pada kualitas sistem yang kuat. Oleh karena itu, penting untuk melakukan *automation testing* pada sistem ini untuk memastikan bahwa semua komponen berfungsi dengan baik dan dapat diandalkan [4].

Solusi untuk mengatasi permasalahan ini, penelitian ini akan menjelaskan solusi dengan melakukan *automation testing* pada sistem informasi Indo Towing menggunakan metode STLC (*Software Testing Life Cycle*) [5] dengan teknik *equivalence partitioning*. Pengerjaan *automation testing* akan menggunakan alat bantu bernama Katalon Studio, yang merupakan perangkat lunak yang sangat efisien dalam proses pengujian perangkat lunak, baik berbasis *web*, *desktop*, maupun *mobile* [3],[6].

Tujuan utama dari penelitian ini adalah untuk memastikan kualitas sistem informasi Indo Towing, sehingga dapat memberikan pengalaman pengguna yang memuaskan dan menghindari potensi kerugian finansial yang disebabkan oleh kegagalan sistem. Dengan demikian, penelitian ini memiliki dampak positif yang signifikan dalam pengembangan teknologi informasi dan pelayanan pelanggan di era digital saat ini.

2. METODE

Metodologi yang digunakan dalam implementasi *automation testing* pada sistem informasi Indo Towing menggunakan Katalon Studio didasarkan pada pendekatan *Software Testing Life Cycle* (STLC). Proses STLC terdiri dari beberapa tahapan penting yang membantu dalam memastikan pengujian perangkat lunak berjalan efektif dan sesuai dengan persyaratan yang telah ditentukan. Gambar 1 menunjukkan tahapan-tahapan dalam pendekatan STLC.



Gambar 1. Metode STLC

2.1. Requirement Analysis

Tahap ini akan dilakukan analisis menyeluruh terhadap persyaratan perangkat lunak dari sudut pandang pengguna. Ini mencakup identifikasi kebutuhan fungsional dan non-fungsional yang harus diuji. Persyaratan ini mencakup kemampuan sistem untuk menyimpan dan mengelola data yang sedang berjalan, tampilan antarmuka pengguna yang mudah digunakan, kemudahan berinteraksi dengan sistem oleh pengguna, kemampuan pencarian dan filter data, serta responsif yang baik dari sistem informasi.

2.2. Test Planning

Tahap ini digunakan untuk merencanakan pengujian dengan mengidentifikasi, mengatur prioritas, dan menjadwalkan aktivitas pengujian yang diperlukan untuk memverifikasi bahwa perangkat lunak memenuhi persyaratan dan beroperasi dengan baik.

2.3. Test Case Development

Tahap ini merupakan tahap yang digunakan untuk mengembangkan *test case* yang akan digunakan dalam *automation testing*. Ini mencakup pembuatan test data yang sesuai dengan persyaratan yang telah ditetapkan sebelumnya.

2.4. Environment Setup

Tahapan ini merupakan tahapan yang digunakan untuk memastikan bahwa lingkungan pengujian (termasuk *hardware* dan *software* yang digunakan) telah siap. Serta menyiapkan *environment test* dan *test data* yang diperlukan untuk pelaksanaan pengujian.

2.5. Test Execution

Tahap ini akan melibatkan eksekusi *test plan* dan *test case* yang telah disiapkan sebelumnya. Selama tahap ini, hasil pengujian dicatat, dan setiap ketidaksesuaian dengan persyaratan dicatat sebagai "Tidak Sesuai dengan yang diharapkan." Jika bug ditemukan, tim pengembang diberitahu dan melakukan koreksi. Setelah *bug* dikoreksi, *automation testing* dapat diulang. Selain itu, selama tahap ini, dokumentasi hasil pengujian disusun untuk keperluan laporan, termasuk laporan *bug* jika ada.

2.6. Test Cycle Closure

Tahapan ini merupakan akhir dari siklus pengujian, dengan menyediakan laporan hasil *automation testing* yang telah diselesaikan, dan mengumpulkan data hasil pengujian dan melakukan evaluasi serta analisis terhadap pengujian yang telah dilakukan. Tujuan tahap ini adalah untuk mengurangi hambatan dalam proses pengujian di masa depan dan meningkatkan kualitas STLC pada waktu yang akan datang.

3. HASIL DAN PEMBAHASAN

Pada bagian ini, akan dibahas hasil dan pembahasan dari implementasi *automation testing* pada sistem informasi Indo Towing menggunakan Katalon Studio. Sistem informasi terdapat 16 *test case* yang mencakup berbagai fitur, Proses pengujian melibatkan serangkaian langkah, mulai dari analisis kebutuhan hingga penutupan siklus pengujian.

3.1. Requirement Analysis

Tahap pertama dalam *Software Testing Life Cycle* (STLC) adalah analisis kebutuhan. Sistem informasi Indo Towing memiliki 6 menu utama: Berita, Galeri, Informasi Pemesanan, Lihat Estimasi Harga, Pemesanan, dan Perusahaan. Setiap menu memiliki rincian sub-menu sebagai berikut:

- a) Menu Berita: Berisi berita-berita mengenai *truck towing* di Lampung dan sekitarnya.
- b) Galeri: Menampilkan dokumentasi foto aktivitas *truck towing* yang telah dilakukan.
- c) Informasi Pemesanan: Tahap input data diri pemesan saat memesan kendaraan di sistem informasi Indo Towing.
- d) Lihat Estimasi Harga: Memungkinkan pengguna untuk mengetahui estimasi harga pengiriman sesuai dengan tujuan sebelum melakukan pemesanan.
- e) Pemesanan: Tahap proses pemesanan yang memungkinkan pengguna memilih rute pengiriman, jenis kendaraan, jenis pengiriman, dan asuransi kendaraan jika ada.

- f) Perusahaan: Berisikan dropdown "Tentang Kami" dan informasi legalitas PT. Bisma Berkah Mandiri, perusahaan yang bergerak dalam jasa *towing*.

3.2. Test Planning

Pada tahap ini, dilakukan perencanaan pengujian dengan menyusun rencana terperinci tentang aktivitas pengujian. Rencana ini mencakup tujuan, ruang lingkup, dan hasil yang diharapkan dari pengujian. Berikut adalah beberapa poin dari rencana pengujian:

- Registrasi Pengguna: Sistem harus memiliki fitur registrasi pengguna yang memungkinkan pengguna untuk membuat akun dengan menyediakan informasi seperti nama, alamat, nomor telepon, dan alamat email.
- Pemesanan Layanan *Towing*: Pengguna harus dapat memesan layanan *towing* melalui sistem. Pemesan harus dapat memilih jenis layanan (darurat atau reguler) dan memberikan informasi tentang kendaraan mereka.
- Penjadwalan Tanggal dan Waktu: Sistem harus memungkinkan pengguna untuk menjadwalkan waktu dan tanggal untuk layanan *towing* yang mereka pesan.
- Verifikasi Kode Pemesanan: Setelah memesan layanan *towing*, pengguna harus dapat memasukkan kode pemesanan untuk melihat detail pemesanan.

3.3. Test Case Development

Test case ini telah dibuat berdasarkan tahap *requirement analysis* dan perencanaan pengujian. Tabel berikut menunjukkan salah satu contoh *test case* yang disusun dengan mengacu pada tahapan sebelumnya, yaitu *test planning*. Tabel *test case* disajikan pada Tabel 1.

Tabel 1. Test Case Berita

Test case (1)	Hasil yang diharapkan (2)
TC01_ Valid berita	Pencarian berita di masukkan judul pada form berita, sistem akan menampilkan berita tersebut.
TC02_ Invalid berita	Jika memasukkan judul pada form berita dengan nama salah, akan muncul "informasi tidak ditemukan".
TC03_ Valid Kategori	Mencoba setiap pilihan pada kategori, kemudian akan menampilkan berita sesuai dengan pilihan pada kategori

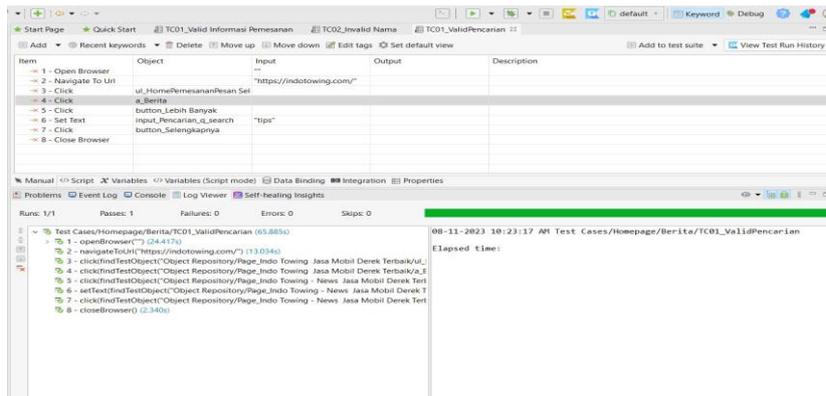
Tabel 1 di atas menunjukkan hasil dari *test case* berita yang telah dijalankan. *Test case* ini mencakup berbagai skenario pengujian, mulai dari pencarian berita hingga memilih kategori.

3.4. Environment Setup

Pada tahap ini, dijelaskan tentang persyaratan lingkungan yang harus dipenuhi agar dapat menjalankan automation testing dengan Katalon Studio. Persyaratan tersebut mencakup instalasi Katalon Studio pada berbagai *platform*, pengaturan JDK versi 8 atau yang lebih baru, serta dukungan untuk beberapa browser seperti Google Chrome, Mozilla Firefox, dan Microsoft Edge.

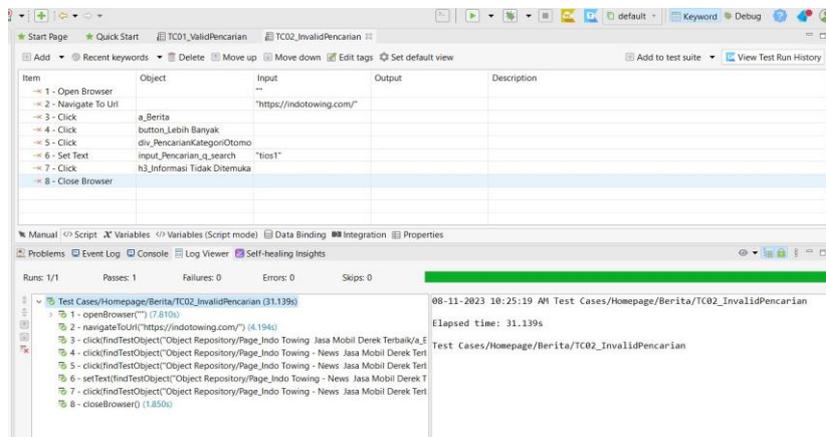
3.5. Test Execution

Tahap eksekusi pengujian adalah langkah penting dalam proses ini. *Test case* dijalankan sesuai dengan urutan yang telah ditentukan. Hasil pengujian ditampilkan dalam bentuk grafik yang mencakup *test case* yang berhasil (warna hijau) dan *test case* yang gagal (warna merah). Berikut adalah beberapa hasil eksekusi *test case* yang akan disajikan dalam Gambar 2.



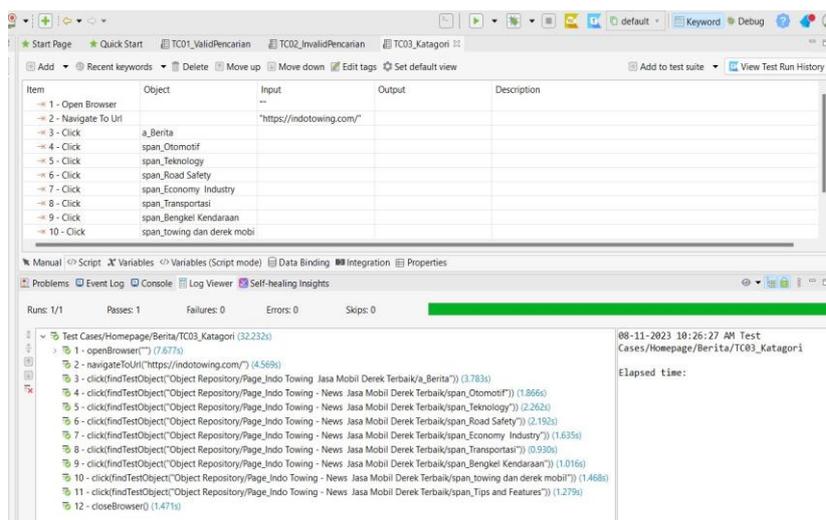
Gambar 2. Hasil test case TC01_Valid Pencarian

Pada Gambar 2 menunjukkan bahwa hasil dari eksekusi test case TC01_Valid Pencarian berhasil dilakukan, Dapat melakukan pencarian berita pada sistem.



Gambar 3. Hasil test case TC02_Invalid Pencarian

Gambar 3 menunjukkan bahwa hasil eksekusi test case TC02_Invalid Pencarian, berhasil dilakukan

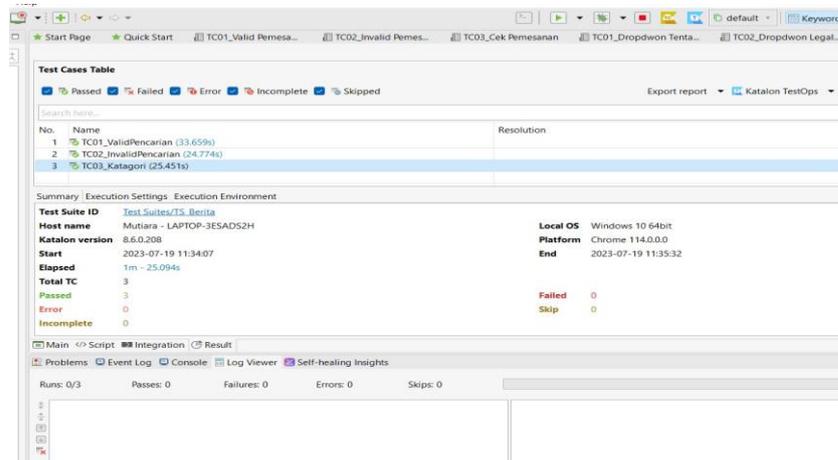


Gambar 4. Hasil test case TC03_Kategori

Gambar 4 menunjukkan bahwa hasil eksekusi *test case* TC03_Kategori, dapat menampilkan berita sesuai dengan pilihan kategori berhasil dilakukan.

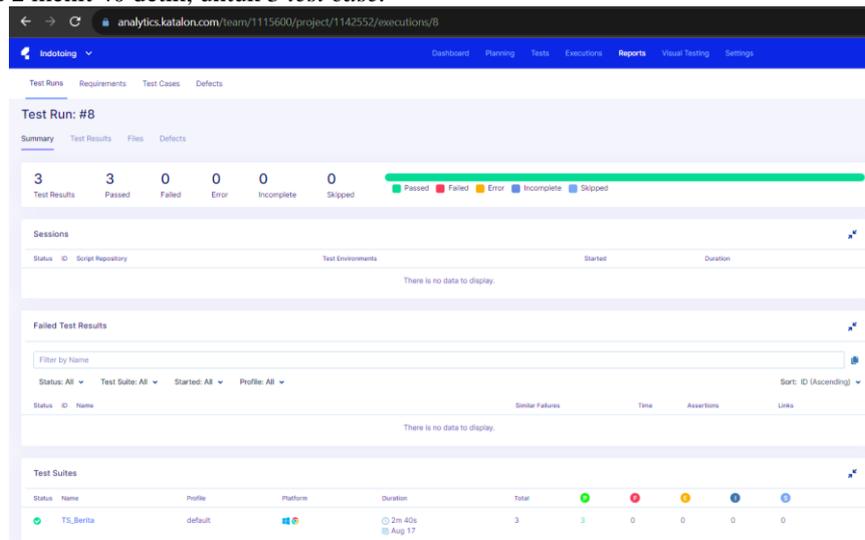
3.6. Test Cycle Closure

Proses pengujian telah dilakukan sebelumnya, tahapan ini digunakan untuk melihat hasil dari pengujian yang telah dilakukan. Hasil dari pengujian yang dilakukan akan di gambarkan pada Gambar 4.



Gambar 5. TS_Berita

Gambar 5 menampilkan hasil Result pada test suites TS_Berita, terdapat 3 *test case* berhasil dilakukan dengan waktu 2 menit 40 detik, untuk 3 *test case*.



Gambar 6. Hasil Test Run

Hasil dari keseluruhan contoh *test case* yang dilakukan automation testing berjumlah 3 *test case* semua berhasil, sesuai dengan yang diinginkan dan tidak ditemukan *bug* pada sistem informasi indo towing, dapat dilihat pada Gambar 6 merupakan hasil pada *test runs* yang terdapat di <https://analytics.katalon.com/home> pada menu *test runs*.

4. KESIMPULAN

Implementasi automation testing menggunakan Katalon Studio pada sistem informasi Indo Towing telah sukses dengan 16 *test case* yang mencakup berbagai fitur. Keuntungan utama meliputi efisiensi, kemudahan penggunaan, dan peningkatan kualitas pengujian. Penggunaan Katalon Studio sebagai *tools* untuk *automation testing* sangat menguntungkan dan efisien karena memiliki fitur yang lengkap, termasuk kemampuan untuk merekam *web*. Dengan demikian, *implementasi automation*

testing menggunakan Katalon Studio dan penerapan STLC memberikan manfaat dalam hal efisiensi, kemudahan, dan kualitas pengujian pada sistem informasi Indo Towing.

DAFTAR PUSTAKA

- [1] Danuri, M. (2019). Perkembangan Dan Transformasi Teknologi Digital. *Jurnal Ilmiah Infokam*, 15(2).
- [2] D. Rizky.(2019).Manual Testing Vs Automated Testing | Dot Intern Medium.<https://Medium.Com/Dot-Intern/Manualtesting-Vs-Automated-Testing-9244aaed1ed5>
- [3] Herlinda, Herlinda, Et Al. "Automation Testing Tool Dalam Pengujian Aplikasi Belajar Tajwid Pada Platform Android." *String (Satuan Tulisan Riset Dan Inovasi Teknologi)* 4.2 (2019): 205-212.
- [4] Y. Wang, "Test Automation Maturity Assessment," *Ieee 11th International Conference On Software Testing, Verification And Validation (Icst)*, 2018.
- [5] J. F. Andry And Reinaldo, "Testing Dan Implementasi Aplikasi Parkir Di Pt Abc Menggunakan Metode Black Box," *Prosiding Seminar Nasional Multidisiplin Ilmu*, 2017.
- [6] Prakoso, B., & Sujarwo, A. (2022). Perancangan Automated Testing Pada StudiKasus Website Indicar. *Jurnal Ilmiah Teknologi Informasi Dan Robotika*, 4(1), 29-32. Doi:<https://Doi.Org/10.33005/Jifti.V4i1.69>