

Comparative Analysis of Automated Testing Tools on GUI WEB-Based Applications

Analisis Perbandingan Tools Pengujian Otomatis pada GUI Aplikasi Berbasis WEB

Septi Melia^{1*}, Fajri Profesio Putra²

^{1,2}Department of Software Engineering, Faculty of Technical Information,
Bengkalis State Polytechnic Riau

E-Mail: ¹septimelia55@gmail.com, ²fajri@polbeng.ac.id

Corresponding Author: Septi Melia

Abstract

There are many automated test instruments available today, but only a few of each piece of software can perform a wide variety of tests. Based on predefined parameters and test cases, the purpose of this research is to compare automated testing tools for Graphical User Interface (GUI). This study utilizes Selenium Webdriver and Katalon Studio, both of which have a number of test parameters, such as Test Case Execution Time, Delay, Image-based testing, Scrolling, and Documentation of Execution Results. The DiaryMe application is the subject of this study, and the Software Testing Life Cycle (STLC) method is used for software testing. The findings of this study indicate that both tools successfully completed the test based on predetermined parameters and test cases. After testing is complete, there are observable differences between the two tools, especially in terms of how they are used and how the functions and features are utilized.

Keyword: Automated Testing, GUI, Katalon Studio, Selenium Webdriver, Software Testing Life Cycle

Abstrak

Ada banyak instrumen pengujian otomatis yang tersedia saat ini, namun hanya sedikit dari setiap perangkat lunak yang dapat melakukan berbagai macam pengujian. Berdasarkan parameter yang telah ditentukan dan kasus uji, tujuan dari penelitian ini adalah untuk membandingkan alat pengujian otomatis untuk *Graphical User Interface (GUI)*. Penelitian ini memanfaatkan *Selenium Webdriver* dan *Katalon Studio*, keduanya memiliki sejumlah parameter pengujian, seperti Waktu Eksekusi Test Case, *Delay*, *Image-based testing*, *Scrolling*, dan Dokumentasi Hasil Eksekusi. Aplikasi *DiaryMe* adalah subjek dari penelitian ini, dan metode *Software Testing Life Cycle (STLC)* digunakan untuk pengujian perangkat lunak. Temuan penelitian ini menunjukkan bahwa kedua alat berhasil menyelesaikan tes berdasarkan parameter yang telah ditentukan dan kasus uji. Setelah pengujian selesai, ada perbedaan yang dapat diamati antara kedua alat tersebut, terutama dalam hal cara penggunaannya dan bagaimana fungsi dan fitur digunakan.

Kata Kunci: GUI, Katalon Studio, Pengujian Otomatis, Selenium Webdriver, Software Testing Life Cycle

1. PENDAHULUAN

Antarmuka program yang dikenal sebagai "*graphical user interface*" (GUI) adalah sarana komunikasi antara perangkat lunak dan pengguna. Karena GUI adalah komponen dari suatu program, setiap perangkat lunak perlu diuji, termasuk perangkat lunak seluler dan situs web. Pengujian GUI adalah pengujian yang dilakukan pada aplikasi yang memiliki titik koneksi dengan cara mencari beberapa cara untuk menguji bagian-bagian GUI. Karena kualitas sistem secara keseluruhan dipengaruhi oleh fungsionalitas widget GUI [6].

Pengujian manual dan pengujian otomatis adalah dua metode pengujian. Pengujian manual akan dicoba atau dilakukan langsung oleh penganalisa atau penguji itu sendiri. Pengujian otomatis berguna untuk meminimalkan sesuatu yang tidak dapat dicadangkan selama pengujian manual atau menghindari kesalahan

manusia. Penguji berinteraksi dengan sistem untuk menguji berbagai fungsionalitas sistem [13], tetapi pengujian manual tidak mencakup berbagai pengujian dan penguji sering melakukan hal yang sama berulang kali [6]. Selenium Webdriver, Katalon Studio, dan sejumlah alat lainnya adalah beberapa di antara banyak alat pengujian otomatis [16].

Selenium Webdriver menikmati manfaat sebagai alat selenium tercepat di antara berbagai bagian departemen alat Selenium. Pengujian lintas-browser dan pengujian dinamis halaman situs didukung oleh Selenium WebDriver [15]. Katalon Studio memiliki kelebihan yaitu katalon mendukung pengujian API, Web, dan Seluler dan tidak mewajibkan penggunaannya memiliki kemampuan pemrograman. Kelemahan katalon tidak mendukung pengujian terdistribusi [16].

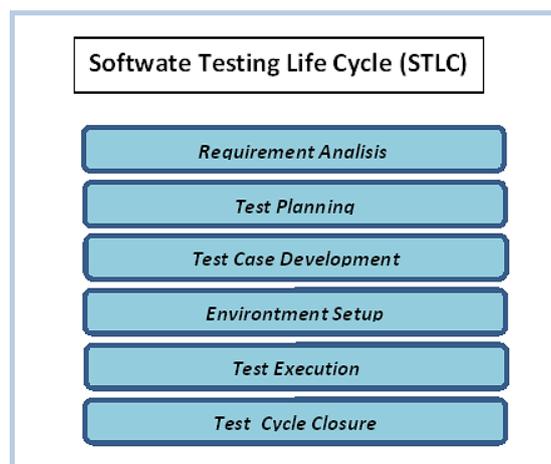
Banyak sekali *tools* pengujian otomatis yang beredar saat ini, namun tidak setiap *tools* memiliki kemampuan untuk melakukan semua jenis pengujian seperti *Fungsional, Load, Performance, Unit* dan beberapa kemampuan lainnya. Jadi, sangat penting untuk memahami aplikasi yang akan diuji dan memilih *tools* pengujian otomatis yang sesuai untuk menguji sebuah aplikasi. Untuk penelitian ini, *tools* pengujian otomatis akan dibandingkan pada parameter yang berbeda-beda dengan tujuan membandingkan *tools* pengujian otomatis berdasarkan parameter dan kasus uji yang telah disediakan. Adapun parameter yang akan digunakan pada penelitian kali ini yaitu: Waktu Eksekusi Test Case, Penundaan (*delay*), *Image-based Testing, Scrolling*, Dokumentasi Hasil Eksekusi.

Waktu Eksekusi TestCase, Parameter ini digunakan untuk melihat waktu yang dibutuhkan oleh setiap perangkat pengujian untuk mengeksekusi situasi percobaan/pengujian yang telah ditentukan [4][10]. Penundaan (*delay*), Penundaan atau bisa juga disebut dengan *delay* atau *sleep*, sangat penting didalam sebuah proses pengujian perangkat lunak karena dapat berguna untuk melihat detail dari masing-masing proses pengujian yang dilakukan [4]. *Image-based Testing*, Rancangan pengujian pada parameter ini yaitu dengan membandingkan hasil yang didapatkan dari masing-masing *tools* setelah melakukan percobaan menguji sebuah gambar sesuai dengan test case yang telah ditentukan [4] [5]. *Scrolling*, Parameter ini digunakan untuk menampilkan bagian yang tidak pas di layar, pengujian akan melakukan *scrolling* yang diperlukan. Parameter ini digunakan untuk melihat dan membandingkan hasil dari *scrolling* tersebut dari masing-masing *tools* yang ada [4]. Dokumentasi Hasil Eksekusi, Parameter ini digunakan untuk membandingkan dokumentasi dari hasil eksekusi test case [4][10].

Aplikasi DiaryMe merupakan sebuah aplikasi penerbitan yang cara kerjanya seperti website blog yang usernya dapat mempublish karya-karya yang ingin ia *publish*. Aplikasi DiaryMe adalah aplikasi berbasis website yang ditujukan untuk para penulis baik bagi para penulis pemula maupun yang sudah berpengalaman untuk mengembangkan bakatnya dengan menuliskan karya-karya atau pengalaman yang pernah dialaminya untuk di *publish* di Aplikasi DiaryMe.

Pada penelitian ini peneliti akan melakukan analisis perbandingan *tools* pengujian otomatis pada GUI Aplikasi berbasis web. Adapun objek yang telah ditetapkan untuk dijadikan wadah perbandingan *tools* yaitu Aplikasi DiaryMe. Peneliti ingin melihat perbandingan yang lebih signifikan dari *tools* yang akan dibandingkan dengan beberapa parameter sebagai pembandingnya yaitu: Waktu Eksekusi Test Case, Penundaan (*delay*), *Image-based Testing, Scrolling*, Dokumentasi Hasil Eksekusi.

2. BAHAN DAN METODE



Gambar 1. Alur Metodologi Penelitian

Metodologi *Software Testing Life Cycle* (STLC) digunakan pada penelitian kali ini untuk menguji dan membandingkan Alat. Proses pengujian STLC dibagi menjadi enam tahap: *requirement analysis, test planning,*

test case development, environment setup, test execution, dan test cycle closure. Pada penelitian ini, metode STLC digunakan untuk menjelaskan tahapan kegiatan yang lebih spesifik dengan menggunakan pendekatan terstruktur [3].

Berdasarkan Gambar 1 didapat langkah-langkah menggunakan metode STLC (*Software Testing Life Cycle*) adalah sebagai berikut:

1. *Requirement Analysis*

Pada tahapan ini peneliti akan melakukan analisis kebutuhan aplikasi baik fungsional maupun non-fungsional. Adapun kebutuhan fungsional dan non-fungsional yang digunakan pada aplikasi DiaryMe yaitu:

Tabel 1. Kebutuhan Aplikasi DiaryMe

Fungsional	Non-Fungsional
<ul style="list-style-type: none"> • Sistem bisa melakukan registrasi ke aplikasi • Sistem bisa melakukan login ke aplikasi • Sistem bisa mengirimkan kendala dan saran yang ditulis user melalui contact. • Sistem bisa melakukan CRUD tambah data post karya. • Sistem dapat menampilkan tampilan halaman detail karya dari image karya yang di klik. 	<ul style="list-style-type: none"> • Sistem dapat diakses menggunakan sistem operasi yang berbeda seperti, Windows, Linux, bahkan Android dengan syarat Anda memiliki program untuk mengakses web. • Sistem dapat membuat batasan dashboard untuk user dan Admin. • Sistem dapat mengirimkan verify akun user di email untuk mengaktifkan akun setelah register. • Sistem harus mem verify bahwa file cover yang di upload oleh user berekstensi: JPG, JPEG, PNG dan GIF, selain itu sistem harus menolak dan menampilkan alert. • Sistem harus mem verify bahwa akun email yang dimasukkan harus benar. • Kecepatan akses pada saat masuk ke Aplikasi DiaryMe (Kecepatan Loading)

2. *Test Planning*

Pada tahapan ini akan dilakukan perancangan pengujian, dimulai dengan menentukan *tools* yang akan diuji, parameter perbandingan, objek penelitian, dan gambaran pengujian. Beberapa kegiatan yang akan dilakukan secara terstruktur dan berurutan akan digabungkan untuk membuat rencana proses pengujian. Adapun tahapan yang akan dilakukan ialah:

- Menentukan *tools* yang akan dibandingkan, yaitu: Selenium Webdriver dan Katalon Studio.
- Menentukan parameter-parameter yang cocok dan sesuai untuk digunakan sebagai pembandingan *tools* yang akan diuji. Adapun parameter yang dipilih dan yang akan digunakan adalah: Waktu Eksekusi Test Case, Penundaan (*Delay*), *Image-based Testing*, *Scrolling*, dan Dokumentasi Hasil Eksekusi.
- Pilih objek yang akan diteliti. Aplikasi DiaryMe adalah subjek penelitian ini.
- Merancang Pengujian.

3. *Test Case Development*

Pada tahap ini akan dibuat beberapa skenario pengujian untuk menguji fungsi sistem sesuai dengan perencanaan.

Tabel 2. Test Case Development

Parameter	TC ID	Test Case Name
P1 (Waktu Eksekusi Test Case)	TC01	Register
	TC01.1	Register 1
	TC01.2	Register 2
	TC01.3	Register 3
	TC01.4	Register 4
P2 (Penundaan atau <i>Delay</i>)	TC02	CRUD Post
	TC02.1	CRUD 1
	TC02.2	CRUD 2
	TC02.3	CRUD 3
P3 (<i>Image based Testing</i>)	TC03	Cover Post
	TC04	Contact
P4 (<i>Scrolling</i>)	TC04.1	Contact 1
	TC04.2	Contact 2

Parameter	TC ID	Test Case Name
P5 (Dokumentasi Hasil Eksekusi)	TC05	Login
	TC05.1	Login 1
	TC05.2	Login 2
	TC05.3	Login 3

4. *Environment Setup*

Pada tahap ini, penguji akan memeriksa lingkungan pengujian, menyiapkannya, dan menyusun daftar persyaratan perangkat lunak dan perangkat keras yang diperlukan untuk pengujian. Rincian peralatan dan pemrograman yang akan digunakan dalam penelitian ini adalah:

- Hardware*: Intel(R) Core (TM) i3-8130U CPU @ 2.20GHz 2.21 GHz. 8 GB RAM. 256 GB SSD.
- Network 100Mbps.
- Software*: *Selenium Webdriver*, *Katalon Studio* dan *Google Chrome*

5. *Test Execution*

Pada tahap ini penguji akan melakukan proses pengujian berdasarkan test skenario yang telah dibuat.

6. *Test Cycle Closure*

Tahapan ini merupakan tahapan terakhir yang akan dilakukan. Pada tahap ini hal yang dilakukan yaitu membuat kesimpulan dari hasil pengujian yang telah dilakukan.

2.1 Eksplorasi Alat Pengujian

1. Selenium Webdriver

Selenium merupakan salah satu alat pengujian otomatis. Selenium memiliki empat jenis konten uji yaitu Selenium Grid, Selenium IDE, Selenium RC (Remote Control) dan Selenium WebDriver. Selenium menawarkan jenis konten uji yang berbeda sesuai dengan kasus yang akan diuji. Selenium juga menyediakan dukungan multi-browser. Selain itu, selenium juga mendukung untuk beberapa sistem operasi dan bahasa pemrograman yang berbeda [5].

Selenium Webdriver akan bekerja dengan membuka web browser dan mengeksekusi *script* yang telah disiapkan oleh tester dengan menggunakan bahasa pemrograman yang dipilih, kemudian webdriver akan menjalankan script tersebut termasuk untuk membuka url. Dari url tadi maka webdriver akan mengarahkan script untuk membuka website dari link url yang telah disiapkan di dalam script tadi, kemudian teknik data *scraping* akan berperan dalam pengambilan informasi dan menyimpannya ke dalam variable [11].

2. Katalon Studio

Katalon Studio merupakan alat yang bisa menguji aplikasi berbasis mobile, browser, hingga API. Katalon Studio tidak mengharuskan penggunaannya untuk menguasai bahasa pemrograman karena katalon studio memiliki *user interface* yang simpel dan informatif sehingga mudah digunakan bagi pemula hingga yang sudah berpengalaman [16].

2.2 Analisis Parameter Perbandingan

Dua alat, Katalon Studio dan Selenium Webdriver, menjadi bahan perbandingan. Parameter yang akan dijadikan sebagai parameter perbandingan telah ditentukan sehingga dapat dilakukan perbandingan. Hasil analisis dan kajian pustaka menjadi dasar pemilihan parameter tersebut. Pengujian dilakukan terhadap satu aplikasi website yaitu Aplikasi DiaryMe yang merupakan sebuah aplikasi penerbitan yang cara kerjanya seperti website blog pada umumnya.

Parameter yang telah ditetapkan menjadi pertimbangan saat menguji fitur aplikasi. Artinya, pengujian yang dilakukan pada elemen-elemen aplikasi dilakukan untuk mendapatkan hasil yang sesuai dengan batas-batas pemeriksaan yang telah ditentukan. Adapun Parameter-parameter yang akan digunakan pada penelitian kali ini, yaitu sebagai berikut:

Tabel 3. Parameter Perbandingan

Parameter	Keterangan
P.1 Waktu Eksekusi Test Case [4][10]	Tujuan dari parameter ini adalah untuk membandingkan jumlah waktu yang diperlukan untuk setiap alat pengujian untuk melakukan pengujian pada kasus uji dan skenario pengujian yang telah ditentukan.
P.2 Penundaan (<i>Delay</i>) [4]	Penundaan atau bisa juga disebut dengan <i>delay</i> atau <i>sleep</i> , sangat penting didalam sebuah proses pengujian perangkat lunak karena dapat berguna untuk melihat apakah <i>tools</i> dapat

Parameter	Keterangan
P.3 <i>Image Based Testing</i> [4] [5]	berhenti sejenak sesuai waktu yang ditentukan atau tidak saat pengujian berlangsung. Parameter ini digunakan untuk membandingkan apakah <i>tools</i> dapat menguji sebuah gambar dari test case yang telah disiapkan.
P.4 <i>Scrolling</i> [4]	Parameter ini digunakan untuk membandingkan apakah <i>tools</i> dapat melakukan <i>scrolling</i> pada saat pengujian berlangsung atau tidak.
P.5 Dokumentasi Hasil Eksekusi [4][10]	Tujuan dari parameter ini adalah untuk membandingkan hasil dokumentasi yang didapatkan dari melakukan eksekusi test case. Misalkan test case yang disediakan yaitu test case login, maka test case tersebut akan dieksekusi oleh masing-masing <i>tools</i> pengujian, kemudian hasil dokumentasi dari masing-masing <i>tools</i> pengujian tersebut akan dibandingkan.

Adapun rancangan pengujian yang akan dilakukan berdasarkan parameter yang dipilih yaitu :

Tabel 4. Perancangan Pengujian

Parameter	Rancangan
P1	Rancangan pengujian pada <i>parameter</i> ini adalah dengan membuat <i>test case</i> sesuai dengan <i>parameter</i> yang akan diuji pada Aplikasi Diaryme. <i>Parameter</i> ini akan diuji dan dijalankan menggunakan dua <i>tools</i> pengujian. Setelah <i>testcase</i> selesai dijalankan, maka akan dibandingkanlah waktu yang dibutuhkan masing-masing <i>tools</i> dalam menguji <i>testcase</i> yang telah disediakan.
P2	Rancangan pengujian parameter ini melibatkan pembuatan kasus uji yang memberi alat pengujian kemampuan untuk menunda atau menunda proses pengujian. Dari melakukan ini, hasil eksperimen akan diperoleh terlepas dari apakah perangkat pengujian dapat ditunda selama pengujian atau tidak.
P3	Rancangan pengujian pada <i>parameter</i> ini yaitu dengan membandingkan hasil yang didapatkan dari masing-masing <i>tools</i> setelah melakukan percobaan menguji sebuah gambar sesuai dengan <i>test case</i> yang telah ditentukan.
P4	Rancangan pengujian untuk parameter ini melibatkan menjalankan kasus pengujian yang memberi alat pengujian kemampuan untuk menelusuri proses pengujian. Fitur <i>scroll</i> , yang memiliki banyak halaman web, akan diuji. Batasan ini digunakan untuk menentukan apakah instrumen dapat terlihat selama pengujian sistem.
P5	Rancangan pengujian untuk <i>parameter</i> ini dengan melihat, memperhatikan, dan membandingkan bagaimana <i>output</i> atau dokumentasi yang dihasilkan dari eksekusi <i>test case</i> yang dijalankan dengan menggunakan dua <i>tools</i> pengujian.

2.3 Eksplorasi Objek Pengujian

Aplikasi DiaryMe merupakan sebuah aplikasi penerbitan yang cara kerjanya seperti website blog pada umumnya. Pada Aplikasi DiaryMe user dapat mempublish karya-karya yang telah ia tulis dan yang ingin ia publish. Aplikasi DiaryMe adalah aplikasi berbasis website yang ditujukan untuk para penulis baik bagi para penulis pemula maupun yang sudah berpengalaman untuk mengembangkan bakatnya dengan menuliskan karya-karya atau pengalaman yang pernah dialaminya untuk di *publish* di Aplikasi DiaryMe.

3. HASIL DAN PEMBAHASAN

Berikut hasil penelitian dari Analisis Perbandingan *Tools* Pengujian Otomatis Pada Aplikasi berbasis Web:

Tabel 5. Tabel Perbandingan Hasil akhir Selenium Webdriver dan Katalon Studio

Parameter	Test Skenario	Selenium Webdriver	Katalon Studio
P1	TC01	7.696 s	15.215 s
P2	TC02	✓	✓
P3	TC03	Require installing additional libraries	Built-in support
P4	TC04	✓	✓
P5	TC05	Not have reporting, but can use a reporting template. HTML & CMD	Html, EXCEL dan XML

Hasil eksekusi dari P1 (Waktu Eksekusi TestCase), waktu yang dibutuhkan masing-masing *tools* pengujian otomatis untuk mengeksekusi test case yang sama dan yang telah ditetapkan yaitu keduanya memiliki waktu masing-masing yang menyesuaikan dengan alat/aplikasi itu bekerja dan setelah dianalisa, yang tercepat diperoleh oleh Selenium Webdriver dengan jumlah rata-rata dari proses pengujian beberapa testcase yaitu 7.696s, proses pengujian yang dilakukan dengan Katalon Studio tergolong lambat karena hasil yang didapatkan lebih besar dibandingkan selenium yaitu 15.215s.

Hasil eksekusi dari P2 (Penundaan) yaitu Sesuai dengan kasus pengujian yang telah ditentukan, kedua alat mendukung penundaan selama proses pengujian. Selama proses pengujian, kedua alat dapat di jeda dan dilanjutkan setelah jangka waktu yang telah ditentukan.. Namun untuk menggunakan *delay* di Selenium Webdriver, diharuskan mengimport *time* terlebih dahulu karena *delay* tidak akan bekerja jika tidak di *import* terlebih dahulu, sedangkan penggunaan *delay* di Katalon Studio tidak diharuskan untuk menginstall atau mengimport apapun karena Katalon telah menyediakan fitur untuk mengatur *delay*.

Hasil eksekusi dari P3 (*Image-based Testing*) yaitu kedua *tools* dapat melakukan pengujian gambar atau image pada saat proses pengujian berlangsung, namun Selenium webdriver memerlukan *library* untuk dapat menjalankan proses pengujian menguji gambar yang telah disesuaikan dengan test case yang telah ditetapkan, sedangkan Katalon tidak memerlukan *library* karena *tools* tersebut mensupport melakukan pengujian gambar.

Hasil eksekusi dari P4 (*Scrolling*) adalah Selama proses pengujian, kedua alat dapat melakukan *scroll* sesuai dengan kasus uji yang telah ditentukan. Test Case dijalankan saat alat sedang melakukan *scrolling*, di mana ia harus *scrolling* hingga menemukan elemen atau kata kunci yang ditentukan di halaman situs web.

Hasil eksekusi dari P5 (Dokumentasi Hasil Eksekusi) adalah kedua *tools* dapat menghasilkan *report* atau dokumentasi hasil eksekusi test case yang telah ditetapkan. Sebenarnya Selenium tidak mendukung pembuatan reporting, namun kita bisa mendapatkan hasil dokumentasinya jika kita menggunakan *framework*, *template* atau memanggil *library* untuk membuat report, jika kita memanggil *library* pytest untuk membuat *report* hasil yang kita dapatkan berupa Dokumen HTML dan CMD dari hasil eksekusi. Sedangkan untuk Katalon Studio tidak diperlukan menggunakan atau menginstall apapun karena Katalon Studio mendukung pembuatan report dengan menghasilkan dokumen HTML, EXCEL dan XML.

4. KESIMPULAN

Dari segi banyaknya parameter yang terpenuhi maka keduanya memenuhi parameter yang telah ditetapkan. Namun jika dilihat dari kemudahan instalasi, antarmuka, dan menggunakan fitur-fitur yang ada di dalamnya maka Katalon Studio memiliki fitur-fitur yang telah tersedia sehingga tidak perlu lagi menambahkan *library* atau *framework* hanya untuk menggunakan fitur yang ada didalamnya. Dari segi kecepatan eksekusi test case maka Selenium Webdriver lebih unggul dibandingkan Katalon Studio karena proses eksekusi menggunakan Selenium Webdriver lebih cepat dibandingkan Katalon Studio.

REFERENSI

- [1] Absharina, D., Agustina, F., dan Program, E. M. (2019). Survey Paper : Software Automated Testing Tool Using. Pilar Nusa Mandiri: Journal of Computing and Information System, 16(2), 175–182.
- [2] Anwar, M. A. H., dan Kurniawan, Y. (2019). Dokumentasi Software Testing Berstandar Ieee 829-2008 Untuk Sistem Informasi Terintegrasi Universitas. Kurawal - Jurnal Teknologi, Informasi dan Industri, 2(2), 118–125. <https://doi.org/10.33479/kurawal.2019.2.2.118-125>
- [3] Ardan, T. S., Zahra, D. F., Junaedi, F. R., dan Rheno, S. (2021). Dokumentasi Software Testing Berstandar IEEE 829-2008 untuk Learning Management System Fakultas Ilmu Komputer Universitas Subang. 6(2), 179–191.
- [4] Arlinta Christy Barus, L. S. (2019). Studi Perbandingan Alat Pengujian Otomatis untuk Aplikasi Android. JTIK : Jurnal Teknologi Informasi dan Ilmu Komputer, 6(6). <https://doi.org/10.25126/jtiik.20196953>
- [5] Mishra, A., dan Atesogullari, D. (2020). Automation Testing Tools : a Comparative View Automation Testing Tools : a Comparative. International Journal on Information and Computer Security, 12(December), 63–76.
- [6] Muhtadi, M. M., Friyadi, M. D., dan Rahmani, A. (2019). Analisis GUI Testing pada Aplikasi E-Commerce menggunakan Katalon. Prosiding Industrial Research Workshop and National Seminar, 10(1), 1387–1393. <https://doi.org/10.35313/irwns.v10i1.1443>
- [7] Ningrum, F. C., Suherman, D., Aryanti, S., Prasetya, H. A., dan Saifudin, A. (2019). Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions. Jurnal Informatika Universitas Pamulang, 4(4), 125. <https://doi.org/10.32493/informatika.v4i4.3782>
- [8] Nyoman, N., Trisnawati, A., Suwija, I. M., dan Oka, A. A. K. (2021). Uji Fungsionalitas Sistem Informasi Manajemen Pegawai dengan Metode Black Box. Ilmiah Teknologi dan Komputer, 2(3).
- [9] Okezie, F., Odun-Ayo, I., dan Bogle, S. (2019). A Critical Analysis of Software Testing Tools. Journal of Physics: Conference Series, 1378(4). <https://doi.org/10.1088/1742-6596/1378/4/042030>
- [10] Pelivani, E., dan Cico, B. (2021). A comparative study of automation testing tools for web applications.

- 2021 10th Mediterranean Conference on Embedded Computing, MECO 2021, 7–10. <https://doi.org/10.1109/MECO52532.2021.9460242>
- [11] Puteri, A., Budi, S., dan Ph, D. (2021). Implementasi Automation dalam Perbandingan Data pada Website Pokemon. 3(November), 446–453.
- [12] Rafi, M. Y., Arifin, I. Y., Safutri, D., Fadilah, D., Riyanto, J., Pamulang, U., Studi, P., Informatika, T., Teknik, F., dan Pamulang, U. (2021). Pengujian White Box Testing Menggunakan Teknik Loop Testing pada Aplikasi Sistem Informasi Perpustakaan (Studi Kasus SMKN 3 Kota Tangerang Selatan). *Jurnal Sains, Teknologi dan Masyarakat*, 1(3), 214–221.
- [13] Rusli, H. (2020). Analisa perbandingan black-box automated testing dan manual testing pada aplikasi accmart.
- [14] Jain & Rajnish, (2018). Comparative Study of So ware Automation Testing Tools : OpenScript and Selenium Comparative Study of Software Automation Testing Tools : *Journal of Engineering Research and Application*, Vol. 8(Issue 2), 29-33 www.ijera.com. <https://doi.org/10.9790/9622-0802032933>
- [15] Setiawan, F. A., Putra, S. D., dan Sahlinal, D. (2015). Pengujian Proyek Website Otomatisasi Dengan Pendekatan Integrasi Antara Selenium Dan Testng. 1–14.
- [16] Zulianto, A., Purbasari, A., Suryani, N., Susanti, A. I., Rinawan, F. R., dan Purnama, W. G. (2021). Pemanfaatan Katalon Studio untuk Otomatisasi Pengujian Black-Box pada Aplikasi iPosyandu. *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, 7(3), 370. <https://doi.org/10.26418/jp.v7i3.46954>