

Penerapan Metode Blackbox Pada Perangkat Lunak Menggunakan Katalon Studio (Studi Kasus: Aplikasi Absensi di PT Astra Sedaya Finance)

Virginia Tirza Tempomona¹, Hendry²

^{1,2}Universitas Kristen Satya Wacana, Jl. Diponegoro 52-60, Salatiga, Indonesia

Email: 672018261@student.uksw.edu¹, hendry@uksw.edu²

Abstract - The quality of the software before commercialization, requires that the software goes through a set testing process of its various properties in order to avoid errors, maintain the quality and increase the value of the software. In this research article, we apply the black box testing method, where this test focuses on the quality of the software from the user's point of view. Increasingly sophisticated technology makes it possible to run tests automatically using tools. This research uses Katalon Studio's tools for automated testing. This allows you to use the function more efficiently. As a result of this test, we successfully implemented automated tests using Katalon Studio's tools. It reduces the time for repeated tests with large amounts of data, automatically prints and displays more detailed test results, and finds easily understandable errors.

Keywords - Automation, Blackbox, Katalon, Software, Testing

Intisari - Kualitas suatu perangkat lunak sebelum dipasarkan perlu melalui proses pengujian dimana perangkat lunak tersebut akan diuji kembali kelayakan berbagai fiturnya untuk mencegah adanya kesalahan, menjaga kualitas, meningkatkan nilai dari perangkat lunak tersebut. Pada penelitian artikel ini akan menerapkan metode *Blackbox Testing* dimana pengujian ini berorientasi pada kualitas dari perangkat lunak dengan sudut pandang pengguna. Dengan teknologi yang semakin maju memungkinkan untuk melakukan pengujian secara otomatis dengan menggunakan bantuan *tools*. Penelitian ini akan memakai *Katalon Studio* untuk *tools* dalam pengujian otomatis yang tentunya membantu pekerjaan lebih efisien dengan serangkaian fiturnya. Hasil dari pengujian ini yaitu berhasil menerapkan pengujian otomatis dengan *Katalon Studio* yang mempersingkat waktu untuk pengujian berulang dengan data yang banyak dan dapat mencetak hasil pengujian keseluruhan secara otomatis lebih rinci untuk menemukan kesalahan yang muncul serta dapat dengan mudah untuk mengerti.

Kata Kunci - Automation, Blackbox, Katalon, Software, Testing

I. PENDAHULUAN

Kualitas suatu perangkat lunak sebelum dipasarkan perlu melalui proses pengujian baik untuk mengetahui kelayakan serta meningkatkan *value* dari perangkat lunak tersebut. Oleh karena itu, pengujian sangat penting dilakukan untuk mencari kesalahan yang disebabkan oleh pengembang[1]. Terutama di era teknologi yang berkembang begitu cepat, banyak aplikasi baik yang berbasis web ataupun mobile diluncurkan untuk mempermudah pekerjaan manusia lebih fleksibel.

Tahun 2020 Indonesia dilanda pandemic dimana perusahaan perlu melakukan migrasi di berbagai hal, termasuk PT Astra Sedaya Finance yang memerlukan sebuah aplikasi absensi yang bertujuan untuk mempermudah pengawasan karyawan yang melakukan absensi secara langsung dan di kantor maupun di luar kantor atau disebut *work from home*. Sebelum aplikasi akan digunakan oleh semua karyawan, tentunya akan melewati proses *software*

testing untuk diuji fitur-fiturnya dari sudut pandang pengguna. Metode *Black Box Testing* berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetasan pada spesifikasi fungsional program[2]. Pengujian otomatis adalah penerapan dari teknologi untuk menguji perangkat lunak dengan memberikan kemampuan lebih efisien dan efektif dalam membangun perangkat lunak dengan kualitas lebih baik. Bagian pekerjaan Quality Control di PT Astra Sedaya Finance yaitu melakukan pengujian baik secara otomatis maupun manual. Pengujian otomatis salah satunya menggunakan *tools* Katalon Studio yang bisa digunakan untuk menguji aplikasi berbasis *website, desktop, mobile*, bahkan API sekaligus bisa memberikan laporan dokumentasi dari hasil pengujian yang telah dijalankan.

Penelitian ini terkait dengan beberapa penelitian terdahulu yang telah dilakukan, salah satunya yaitu “Comprehensive Study of Software Testing Techniques and Strategies: A Review” di tahun 2020 oleh Divyani Shivkumar Taley bersama Dr. Bagshree Pathak menyatakan bahwa pengujian perangkat lunak membantu untuk mengidentifikasi bug atau kesalahan dan bertujuan untuk membuat perangkat lunak stabil dan berkualitas sehingga dapat digunakan oleh pengguna[3]. Pada penelitian yang dilakukan oleh Hartono Ruslis pada tahun 2020 yaitu “Analisa Perbandingan Black-Box Automated Testing Dan Manual Testing Pada Aplikasi ACCMart” dimana hasil penelitiannya yaitu pengujian manual baik dilakukan pada aplikasi yang belum pernah dilakukan pengujian dan sebaliknya pengujian otomatis baik dilakukan pada pengujian dengan alur yang jelas [4]. Selanjutnya di tahun 2020 penelitian oleh G. Ken, P. Mudjihartono, Y. Harjoseputro pada artikelnya “Pengujian Aplikasi Mobile untuk Lelang mobil dengan Metode Black Box menggunakan Automation Testing” menjelaskan bahwa *tools* Katalon Studio dari sisi *interface* sangat mudah digunakan[5].

Dedi Wahyono dalam penelitiannya di tahun 2020 “Automation Regression Testing Pada Aplikasi iFocus Mobile Menggunakan Katalon Studio PT GUE” menjelaskan bahwa pengujian regresi terotomatisasi menggunakan Katalon Studio khususnya untuk pengujian aplikasi perangkat mobile dengan komputer [1]. Terbaru pada tahun 2021 oleh Fityan Ardi dan Hanson Prihantoro Putro pada penelitiannya yaitu “Pengujian Black Box Aplikasi Mobile Menggunakan Katalon Studio (Studi Kasus: ACC Partner PT. Astra Sedaya Finance)” menjelaskan hasil pengujian otomatis lebih efektif dibandingkan dengan pengujian secara manual yang memerlukan banyak waktu[6].

Masalah yang ingin dipecahkan dalam penelitian ini yaitu mengetahui hasil pengujian otomatis pada aplikasi, menemukan dan memperbaiki kesalahan dari hasil pengujian, serta mengetahui kelebihan dan kekurangan dari jalannya pengujian otomatis menggunakan *tools* Katalon Studio. Manfaat dari penelitian ini diharapkan dapat menambah ilmu pengetahuan dan informasi bagi peneliti maupun calon peneliti terkait metode *Black Box Testing* dan pengujian otomatis dengan Katalon Studio, meningkatkan minat lebih banyak orang mengenai *software testing*, serta berkontribusi untuk memperbaiki ataupun mengembangkan dari sisi kualitas aplikasi yang diujikan

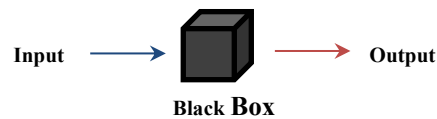
II. SIGNIFIKANSI STUDI

Pengujian perangkat lunak merupakan proses untuk mengevaluasi dan memverifikasi suatu perangkat lunak atau aplikasi melakukan apa yang seharusnya dilakukan. Pengujian perangkat lunak tidak hanya untuk menemukan kesalahan atau *bug* tetapi bertujuan untuk meningkatkan nilai kualitas perangkat lunak tersebut.

A. Studi Literatur

1. Black Box Testing.

Black box testing yaitu pengujian sistem tanpa perlu mengetahui struktur di dalamnya seperti kode atau program[7]. Penguji dapat memberikan masukan dan mengamati keluaran yang dihasilkan oleh sistem yang diuji dapat dilihat pada Gambar 1.

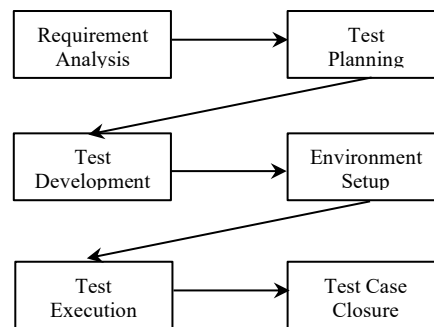


Gambar 1 Black Box Testing

Hal ini memungkinkan untuk mengidentifikasi bagaimana kerja sistem merespon tindakan relevan dari pengguna yang diharapkan, tidak diharapkan, waktu, bahkan masalah kegunaan dan kendala[8]. Keuntungan melakukan pengujian dengan metode ini adalah pengujian berorientasi pada kualitas dari perangkat lunak dari sudut pandang pengguna sehingga untuk melakukan pengujian tidak terlalu rumit[9].

2. *Software Testing Life Cycle*

Software Testing Life Cycle (STLC) adalah tahapan proses pengujian yang dilaksanakan secara spesifik untuk memastikan bahwa tujuan kualitas dan perangkat lunak terpenuhi. STLC terkait langsung dengan *Software Development Life Cycle* (SDLC). Proses *software testing* dapat dilihat pada Gambar 2 berikut.



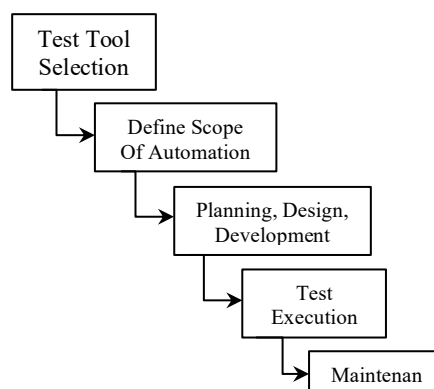
Gambar 2 Proses STLC

3. *BUG*

Bug dalam pengujian perangkat lunak adalah masalah atau *error* yang menyebabkan perangkat lunak mengalami kegagalan untuk menjalankan fungsinya[10].

4. *Automation Testing*

Automation testing merupakan teknik pengujian perangkat lunak dimana prosesnya membutuhkan alat perangkat lunak pengujian pengujian otomatis khusus untuk menjalankan rangkaian pengujian yang telah ditentukan[11]. Tahapan proses dari *automation testing* dapat dilihat pada Gambar 3.



Gambar 3 Proses Automation Testing

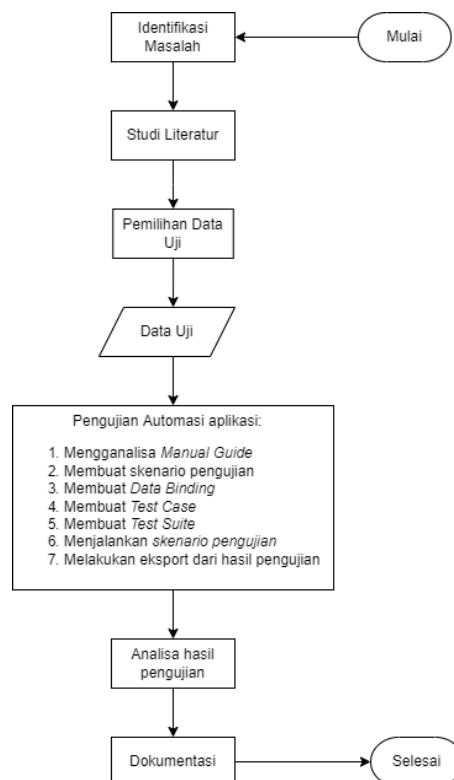
Dalam pengujian ini, penguji dapat memasukan data uji ke dalam sistem pengujian, membandingkan antara hasil yang diharapkan dan aktualnya, serta menghasilkan laporan dari pengujian yang telah dilakukan[12].

5. *Katalon Studio*

Katalon Studio adalah suatu alat pengujian otomatis yang menggunakan kemampuan *library* Selenium dan Appium yang keduanya sudah diinstal sebelumnya untuk menjalankan pengujian pada *mobile*[13]. Katalon Studio menyediakan solusi pengujian otomatis untuk perangkat lunak berbasis *web, mobile*, bahkan API, kemudian dapat membuat pengujian terotomatisasi secara cepat dengan berbagai fiturnya yang sangat mempermudah seorang penguji melakukan pekerjaannya di berbagai platform[14]. Selain hal tersebut Katalon Studio juga memiliki kelebihan dibanding *tools* lainnya pada tampilan antarmuka 1 menarik, dapat terintegrasi dengan platform lainnya, mendukung adanya pekerjaan bersamaan berkolaborasi seperti Git dan Microsoft Teams, pembuatan *script* dalam waktu singkat yang menjadikan pengguna dengan keterampilan pemrograman berbeda dan terbatas dapat menggunakannya dengan mudah dengan fitur yang ada, terdapat dasbor dan laporan analitik yang memungkinkan pengguna menganalisa pengujian berdasar skenario yang dijalankan dan dapat diekspor dalam bentuk file PDF, HTML, Excel, atau CSV.

B. *Metode Penelitian*

Penelitian ini berfokus pada pengujian aplikasi *mobile* pada aplikasi absensi untuk mengetahui fitur-fitur aplikasi tersebut sudah sesuai memenuhi proses dan hasil yang diharapkan. Metode yang akan digunakan dalam penelitian ini adalah *Black Box Testing* dimana pengujian aplikasi dengan sudut pandang dari pengguna. Pengujian aplikasi dilakukan secara otomatis menggunakan *tools* yang telah ditentukan yaitu *Katalon Studio* dimana aplikasi yang diuji berbasis *Mobile* dengan menerapkan prinsip STLC (*Software Testing Life Cycle*). Berdasar metode penelitian dan topik yang diangkat oleh peneliti untuk menyelesaikan peneliti secara terstruktur, maka alur penelitian akan seperti pada Gambar 4 berikut.



Gambar 4 Alur Penelitian

Alur penelitian ini dimulai dengan melakukan identifikasi masalah yang terkait dalam topik penelitian yang sudah ditentukan. Studi literatur menjadi pilihan peneliti untuk mencari penelitian terdahulu yang terkait dengan penelitian yang akan dilakukan bertujuan untuk membantu peneliti melakukan penelitian dengan lebih terpercaya. Kemudian, melakukan pemilihan data uji yang akan digunakan dan data uji tersebut diolah untuk bisa dilanjutkan dalam proses pengujian aplikasi secara otomatis menggunakan *tools* yang telah dipilih yaitu Katalon Studio dengan prosesnya[15] yaitu:

1. Menganalisa *manual guide*
2. Membuat skenario pengujian dan menyiapkan data *binding* dari data uji yang telah diolah.
3. Membuat *test case*
4. Membuat *test suite*
5. Menjalani skenario pengujian
6. Mengekspor dari hasil pengujian otomatis tersebut.
7. Menganalisa hasil pengujian dan diakhiri dengan dokumentasi dari keseluruhan proses pengujian yang dilakukan.

III. HASIL DAN PEMBAHASAN

Aplikasi Absensi PT Astra Sedaya Finance merupakan aplikasi yang dikhususkan bagi pegawai untuk memudahkan dalam melakukan absen kerja. Aplikasi ini dapat diunduh melalui *smartphone*. Untuk dapat mengakses seluruh fitur yang ada dalam aplikasi ini memerlukan *username* dan kata sandi yang sudah terdaftar dalam sistem. Pengguna aplikasi ini memiliki tiga role yaitu Staff, Lead, dan Head dimana Staf yang sudah melakukan absen akan melewati proses persetujuan dari masing-masing Lead, kemudian absen Lead akan melewati proses persetujuan dari Head. Fitur utama aplikasi ini yaitu masuk ke aplikasi, melakukan absen masuk dan keluar, *approval* bagi user Lead dan Head, *Task* yang harus dikerjakan, Riwayat melakukan absen, dan terakhir Pengaturan Akun.

Berdasar penjelasan aplikasi yang terdapat pada *Manual Guide*, berikutnya skenario pengujian dapat dibuat seperti pada Tabel I, Tabel II, dan Tabel III terutama untuk melakukan pendaftaran dan masuk ke dalam aplikasi.

TABEL I
SKENARIO LOGIN

Test Case ID	Feature	Detail Test Case	Test Step	Expected Result
TC01001	Login	Berhasil. <i>User login</i> dengan <i>username</i> benar dan <i>password</i> benar.	1. User membuka Aplikasi 2. User mengisi <i>username</i> yang benar 3. User mengisi <i>password</i> yang salah 4. Klik <i>button</i> "Login"	<i>User</i> berhasil masuk aplikasi.
TC01002	Login	Tidak Berhasil. <i>User login</i> dengan <i>username</i> salah dan <i>password</i> benar.	1. User membuka Aplikasi 2. User mengisi <i>username</i> yang salah 3. User mengisi <i>password</i> yang benar 4. Klik <i>button</i> "Login"	<i>User</i> tidak berhasil masuk aplikasi.
TC01003	Login	Tidak Berhasil. <i>User login</i> dengan <i>username</i> benar dan <i>password</i> salah.	1. User membuka Aplikasi 2. User mengisi <i>username</i> yang benar 3. User mengisi <i>password</i> yang salah 4. Klik <i>button</i> "Login"	<i>User</i> tidak berhasil masuk aplikasi.
TC01004	Login	Tidak Berhasil. <i>User login</i> dengan mengosongkan <i>username</i> dan <i>password</i> .	1. User membuka Aplikasi 2. User tidak mengisi <i>username</i> 3. User tidak mengisi <i>password</i> 4. Klik <i>button</i> "Login"	<i>User</i> tidak berhasil masuk aplikasi.

TABEL II
SKENARIO REGISTRASI

Test Case ID	Feature	Detail Test Case	Test Step	Expected Result
TC02001	Registrasi	Berhasil. <i>User</i> mengisi semua field pada form registrasi dengan data yang benar dan username belum terdaftar.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Klik <i>button</i> “Buat Akun” 3. <i>User</i> mengisi <i>Nama</i> 4. <i>User</i> mengisi <i>Username</i> yang baru 5. <i>User</i> mengisi <i>Password</i> 6. <i>User</i> mengisi <i>Alamat</i> 7. <i>User</i> mengisi <i>Nomor Telepon</i> 8. <i>User</i> memilih <i>Pekerjaan</i> 9. <i>User</i> memilih <i>Role</i> 	<i>User</i> berhasil mendaftar ke sistem.
TC02002	Registrasi	Tidak Berhasil. <i>User</i> mengisi semua field pada form registrasi dengan data yang benar dan username sudah terdaftar.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Klik <i>button</i> “Buat Akun” 3. <i>User</i> mengisi <i>Nama</i> 4. <i>User</i> mengisi <i>Username</i> yang sudah terdaftar. 5. <i>User</i> mengisi <i>Password</i> 6. <i>User</i> mengisi <i>Alamat</i> 7. <i>User</i> mengisi <i>Nomor Telepon</i> 8. <i>User</i> memilih <i>Pekerjaan</i> 9. <i>User</i> memilih <i>Role</i> 	<i>User</i> tidak berhasil mendaftar ke sistem.
TC02003	Registrasi	Tidak Berhasil. <i>User</i> mengosongkan field pada form registrasi.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Klik <i>button</i> “Buat Akun” 3. <i>User</i> tidak mengisi <i>Nama</i> 4. <i>User</i> tidak mengisi <i>Username</i> 5. <i>User</i> tidak mengisi <i>Password</i> 6. <i>User</i> tidak mengisi <i>Alamat</i> 7. <i>User</i> tidak mengisi <i>Nomor Telepon</i> 8. <i>User</i> tidak memilih <i>Pekerjaan</i> 9. <i>User</i> tidak memilih <i>Role</i> 	<i>User</i> tidak berhasil mendaftar ke sistem.
TC02004	Registrasi	Tidak Berhasil. <i>User</i> tidak mengisi semua field pada form registrasi.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Klik <i>button</i> “Buat Akun” 3. <i>User</i> mengisi <i>Nama</i> 4. <i>User</i> tidak mengisi <i>Username</i> 5. <i>User</i> mengisi <i>Password</i> 6. <i>User</i> mengisi <i>Alamat</i> 7. <i>User</i> mengisi <i>Nomor Telepon</i> 8. <i>User</i> memilih <i>Pekerjaan</i> 9. <i>User</i> memilih <i>Role</i> 	<i>User</i> tidak berhasil mendaftar ke sistem.
TC02005	Registrasi	Tidak Berhasil. <i>User</i> mengisi semua field pada form dengan data yang benar dan username sudah terdaftar.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Klik <i>button</i> “Buat Akun” 3. <i>User</i> mengisi <i>Nama</i> 4. <i>User</i> mengisi <i>Username</i> yang sudah terdaftar. 5. <i>User</i> mengisi <i>Password</i> 6. <i>User</i> mengisi <i>Alamat</i> 7. <i>User</i> mengisi <i>Nomor Telepon</i> 8. <i>User</i> memilih <i>Pekerjaan</i> 9. <i>User</i> memilih <i>Role</i> 	<i>User</i> tidak berhasil mendaftar ke sistem.

TABEL III
SKENARIO APPROVAL

Test Case ID	Feature	Detail Test Case	Test Step	Expected Result
TC03001	Approval	Berhasil. Login dengan <i>User</i> Lead dan menyetujui absens masuk dari Staff	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Login dengan <i>User</i> Lead 3. Klik <i>button</i> “Login” 4. Pilih menu <i>Approve</i> 5. Pilih icon menyetujui absensi 	<i>User</i> Lead berhasil melakukan persetujuan absen.
TC03002	Approval	Berhasil. Menu approval tidak muncul jika login dengan <i>User</i> Staff.	<ol style="list-style-type: none"> 1. <i>User</i> membuka Aplikasi 2. Login dengan <i>User</i> Staff 3. Klik <i>button</i> “Login” 	Menu Approval tidak muncul pada aplikasi.

Test Case ID	Feature	Detail Test Case	Test Step	Expected Result
TC03003	Approval	Berhasil. Login dengan <i>User</i> Head dan menyetujui absens masuk dari Lead	1. User membuka Aplikasi 2. Login dengan User Head 3. Klik <i>button</i> "Login" 4. Pilih menu <i>Approve</i> 5. Pilih icon menyetujui absensi	<i>User</i> Head berhasil melakukan persetujuan absen.

Data Binding dibuat melalui Microsoft Excel yang nantinya akan diimport ke Katalon Studio dengan fitur Data File untuk dijadikan data ketika menjalankan skenario pengujian, berikut data yang sudah disediakan pada Tabel IV dan Tabel V.

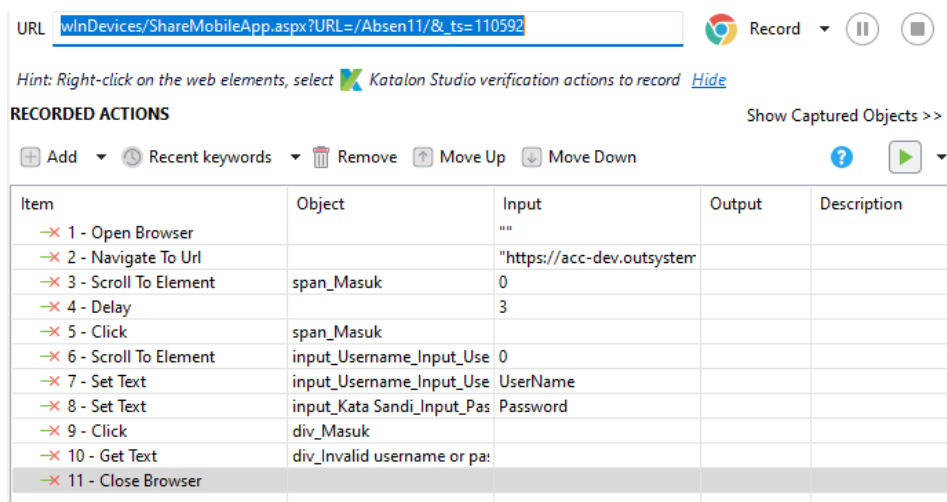
TABEL IV
DATA BINDING LOGIN

Username	Password	Expected
null	null	fail
UserSalah	null	fail
null	PasswordSalah	fail
virgi123	null	fail
null	virgi123	fail
virgi123	PasswordSalah	fail
UserSalah	virgi123	fail
virgi123	virgi123	pass

TABEL V.
DATA BINDING REGISTRASI

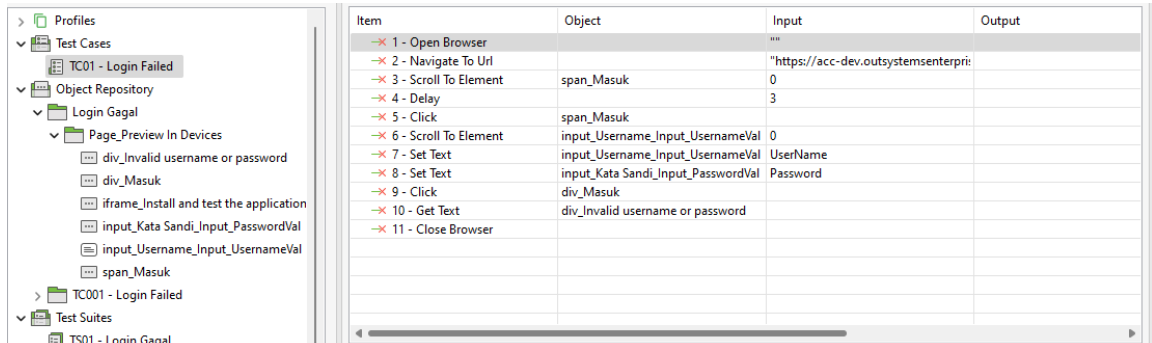
Nama	Username	Password	Email	Alamat	NomorTelepon	Pekerjaan	Role
null	null	null	null	null	null	null	null
Winter	null	null	null	null	null	null	null
Winter	winterkim	null	null	null	null	null	null
Winter	winterkim	winterkim	null	null	null	null	null
Winter	winterkim	winterkim	winterkim@gmail.com	null	null	null	null
Winter	winterkim	winterkim	winterkim@gmail.com	Bandung	null	null	null
Winter	winterkim	winterkim	winterkim@gmail.com	Bandung	089712340087	null	null
Winter	winterkim	winterkim	winterkim@gmail.com	Bandung	089712340087	Quality Control	null
Winter	winterkim	winterkim	winterkim@gmail.com	Bandung	089712340087	Quality Control	STAFF

Dalam membuat suatu *test case* hal yang pertama dilakukan yaitu mengidentifikasi *object* dapat dilakukan dengan dua fitur yang ada pada Katalon Studio melalui *spy* dan *record object*. Untuk pengujian kali ini penulis memilih menggunakan fitur *record* seperti pada Gambar 5 berikut.



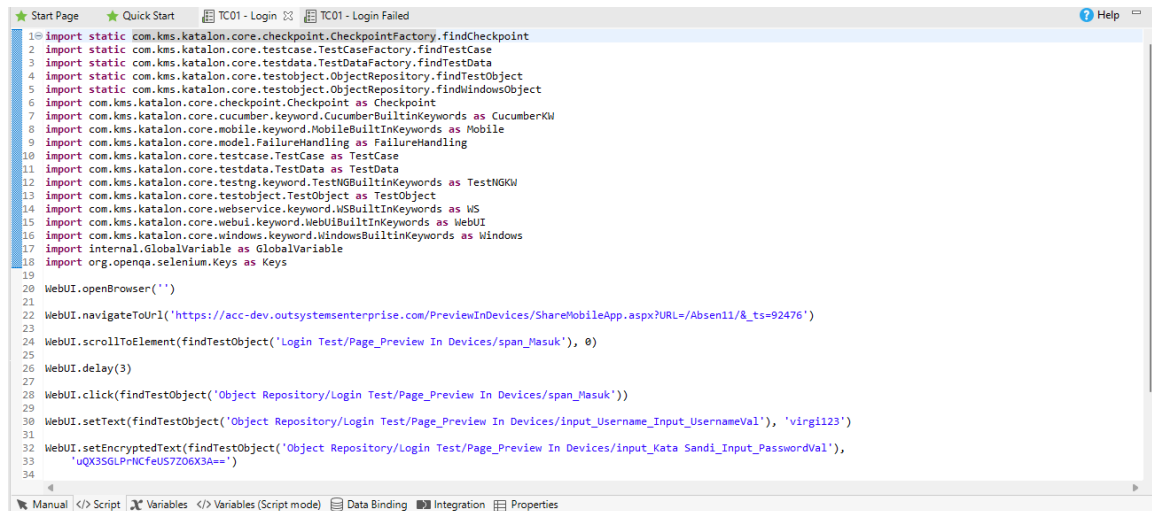
Gambar 5 Record Object

Dengan menggunakan fitur *record object* maka lebih memudahkan dalam melakukan pengujian karena akan merekam setiap langkah dan *object* yang akan menghasilkan sebuah *script* secara otomatis pada Gambar 6 berikut ini:



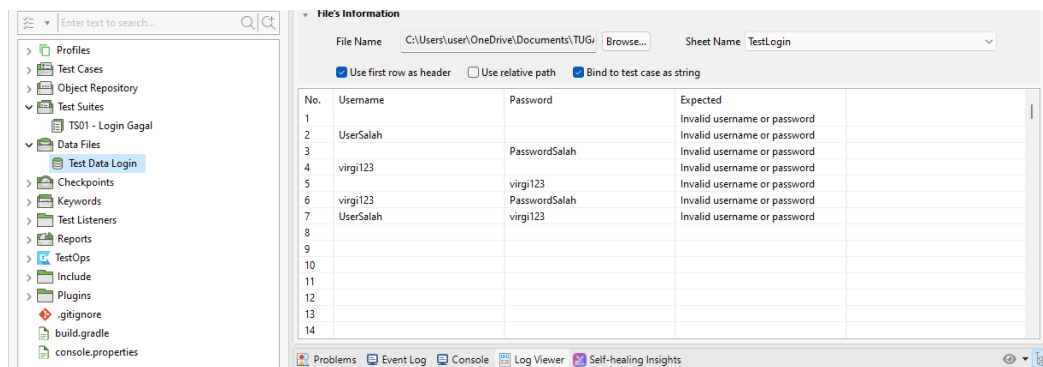
Gambar 6 Script Hasil Record Object

Berikutnya dari hasil perekaman *object* dan *script* tersebut dapat dibuat suatu *test case* untuk menjalankan pengujian. Jika ingin menambah *script* kode tertentu, maka dapat juga ditambahkan pada *script manual* seperti pada Gambar 7.



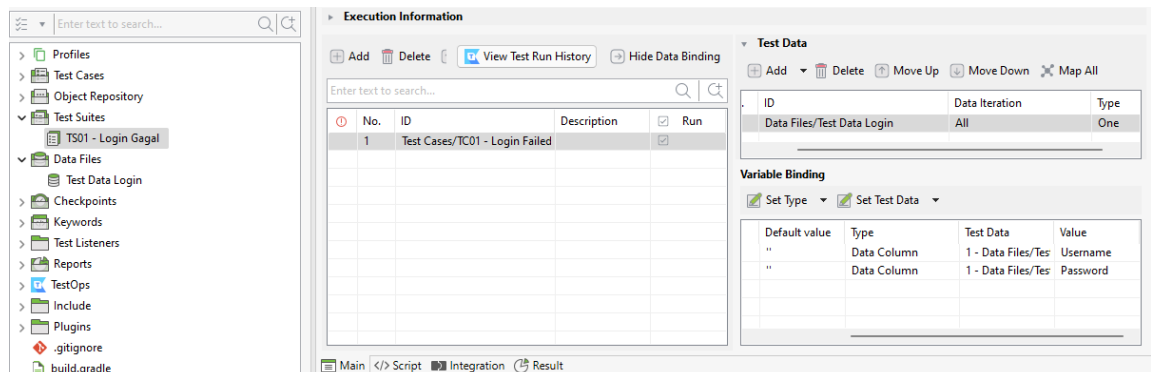
Gambar 7 Tambah Script Secara Manual

Hal penting sebelum membuat suatu *test suite* baru yaitu pastikan melakukan import data yang telah disiapkan dari Microsoft Excel pada Data File. Detailnya seperti pada Gambar 8.



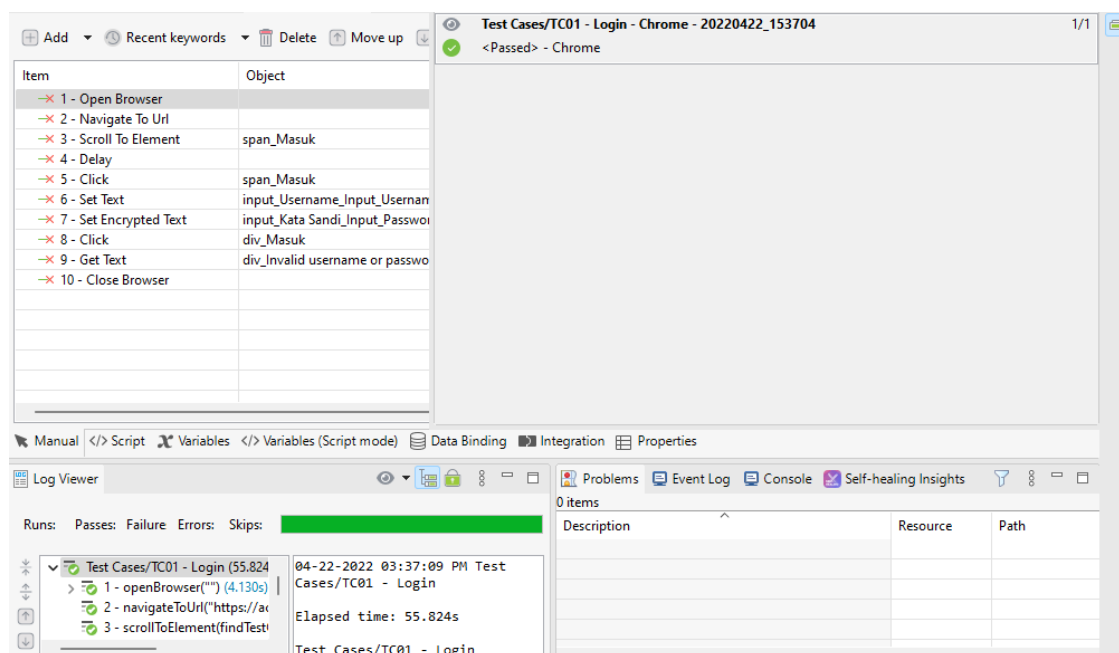
Gambar 8 Import Data Binding dari Excel

Kemudian membuat *test suite* baru, penguji harus memilih Data Files dan Test Case untuk menjalankan testing. Selanjutnya membuat variabel yang dapat saling sinkron seperti pada Gambar 9.



Gambar 9 Mapping Data Binding

Ketika Test Case, Data Files, dan Test Suite berhasil dibuat maka skenario pengujian secara otomatis dapat dilakukan dengan Run pada Test Suite. Semua *script* dan data akan diproses sesuai urutan ataupun pengulangan jika diperlukan seperti pada Gambar 10 berikut ini.



Gambar 10 Proses Jalannya Test Suite

Untuk melihat detail dari jalannya proses pengujian ini dapat dilihat pada bagian Log Viewer sesuai *script* pada Gambar 11 berikut.



Gambar 11 Tampilan Detail Proses Test Suite

Serangkaian pengujian otomatis telah dilakukan, Katalon Studio juga menyediakan fitur Report hasil ketika Tes Suite dijalankan maka dapat dilakukan *export* hasil pengujian dalam bentuk **pdf*, **csv*, dan **html*. Berikut detail hasil dari pengujian yang disediakan dengan file PDF pada Gambar 12.

TS01 - Login Gagal

Execution Environment

Host name	user - DESKTOP-VIRGI
OS	Windows 10 64bit
Katalon version	7.9.0.208
Browser	Chrome 100.0.4896.127

Summary

ID	Test Suites/TS01 - Login Gagal		
Description			
Total	15		
Passed	5	Failed	9
Error	0	Incomplete	1
Start	2022-04-22 14:52:02	End	2022-04-22 15:06:03
Elapsed	14m - 1.078s		

#	ID	Description	Status
1	Test Cases/TC01 - Login Failed		FAILED
2	Test Cases/TC01 - Login Failed		FAILED
3	Test Cases/TC01 - Login Failed		FAILED
4	Test Cases/TC01 - Login Failed		FAILED
5	Test Cases/TC01 - Login Failed		FAILED
6	Test Cases/TC01 - Login Failed		FAILED
7	Test Cases/TC01 - Login Failed		FAILED
8	Test Cases/TC01 - Login Failed		PASSED
9	Test Cases/TC01 - Login Failed		PASSED
10	Test Cases/TC01 - Login Failed		PASSED
11	Test Cases/TC01 - Login Failed		PASSED
12	Test Cases/TC01 - Login Failed		FAILED
13	Test Cases/TC01 - Login Failed		FAILED
14	Test Cases/TC01 - Login Failed		PASSED
15	Test Cases/TC01 - Login Failed		INCOMPLETE

Gambar 12 Hasil Report dari Test Suite

IV. KESIMPULAN

Penerapan pengujian otomatis pada aplikasi absensi ini berhasil dilakukan sesuai dengan skenario dan data yang telah dibuat. Dalam melakukan pengujian otomatis dengan menggunakan *tools* Katalon Studio sangat memudahkan dan menghemat waktu suatu pengujian karena fiturnya cukup lengkap, seperti *record object* yang mudah dimengerti saat dijalankan serta *object* yang diperlukan terekam dengan baik pada hasil *script* walau terkadang *object* tidak berhasil teridentifikasi karena pengaruh dari aplikasi terkait dalam memberikan nama atribut yang memungkinkan kesalahan membaca *object* yang akhirnya memakan waktu lebih untuk memberikan *script* tambahan. Selain itu ketika menjalankan *test case* terkait jika menemukan *bugs* akan ditampilkan lebih rinci untuk letak kesalahannya. Fitur-fitur yang ada di aplikasi absensi terkait juga berhasil diuji kualitasnya, masih terdapat kekurangan yang harusnya bisa diberi nilai tambah. Dengan menggunakan Katalon Studio dapat memberikan fitur untuk memungkinkan hasil *report* semua pengujian yang telah dilakukan dengan lebih lengkap dan dapat diekspor dalam file Excel, PDF, HTML, dan CSV.

REFERENSI

- [1] D. Wahyono, "Automation Regression Testing Pada Aplikasi iFocus Mobile Menggunakan Katalon Studio Studi Kasus PT GUE," Universitas Atma Jaya Yogyakarta, 2020.
- [2] M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, "PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)," vol. I, no. 3, pp. 31–36, 2015.
- [3] Divyani Shivkumar Taley, "Comprehensive Study of Software Testing Techniques and Strategies: A Review," *Int. J. Eng. Res.*, vol. V9, no. 08, pp. 817–822, 2020, doi: 10.17577/ijertv9is080373.
- [4] H. Rusli, "Analisa perbandingan black-box automated testing dan manual testing pada aplikasi accmart," 2020.
- [5] G. Ken, P. Mudjihartono, and Y. Harjoseputro, "Pengujian Aplikasi Mobile untuk Lelang Mobil dengan Metode Black Box menggunakan Automation Testing Tool," *J. Buana Inform. UAJY*, vol. 1, no. 1, pp. 79–87, 2020.
- [6] F. Ardi and H. P. Putro, "Pengujian Black Box Aplikasi Mobile Menggunakan Katalon Studio (Studi Kasus: ACC Partner PT. Astra Sedaya Finance)," *Automata*, vol. 2, no. 1, 2021.
- [7] M. Nurudin, W. Jayanti, R. D. Saputro, M. P. Saputra, and Y. Yulianti, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 143, 2019, doi: 10.32493/informatika.v4i4.3841.
- [8] Y. F. Achmad and A. Yulfitri, "Pengujian Sistem Pendukung Keputusan Menggunakan Black Box Testing Studi Kasus E-Wisudawan Di Institut Sains Dan Teknologi AL-Kamal," *J. Ilmu Komput.*, vol. 5, p. 42, 2020.
- [9] S. P. Utami, K. Eviyanti, W. Sari, and S. C. Haryanti, "Rancang Bangun Aplikasi Edukasi Tuberkulosis Menggunakan Metode Scrum," *J. INOVTEK POLBENG - SERI Inform.*, vol. 7, Bo.1, pp. 83–96, 2022.
- [10] A. V. Sefiddarboon, "A Study of Bugs in Test Code and a Test Model for Analyzing Tests," no. October, 2016.
- [11] H. Herlinda, D. Katarina, and E. W. Ambarsari, M.Kom., "Automation Testing Tool dalam Pengujian Aplikasi Belajar Tajwid pada Platform Android," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 4, no. 2, p. 205, 2019, doi: 10.30998/string.v4i2.5285.
- [12] P. Mahajan, H. Shedge, and U. Patkar, "Automation Testing In Software

- Organization,” *Int. J. Comput. Appl. Technol. Res.*, vol. 5, no. 4, pp. 198–201, 2016, doi: 10.7753/ijcatr0504.1004.
- [13] A. A. Alotaibi and R. J. Qureshi, “Novel Framework for Automation Testing of Mobile Applications using Appium,” *Int. J. Mod. Educ. Comput. Sci.*, vol. 9, no. 2, pp. 34–40, 2017, doi: 10.5815/ijmeecs.2017.02.04.
- [14] A. Zulianto, A. Purbasari, N. Suryani, A. I. Susanti, F. R. Rinawan, and W. G. Purnama, “Pemanfaatan Katalon Studio untuk Otomatisasi Pengujian Black-Box pada Aplikasi iPosyandu,” *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 3, p. 370, 2021, doi: 10.26418/jp.v7i3.46954.
- [15] S. Tjandra, I. Maryati, and J. Theopilus, “Automated Software Testing for Multi Platform Applications using Katalon,” *Widya Tek.*, vol. 20, no. 1, pp. 10–16, 2021, [Online]. Available: <http://journal.wima.ac.id/index.php/teknik/article/view/3114>.

UCAPAN TERIMA KASIH

Penulis sangat berterimakasih pada pihak PT Sedaya Finance yang telah memberi izin untuk melakukan penelitian terkait *automation testing* yang telah digunakan oleh perusahaan selama ini dan memberi kesempatan untuk menguji aplikasi baru yang akan dikembangkan kedepannya.