

Pengujian Black Box Testing pada Aplikasi Inventori Barang Berbasis Web di PT. AINO Indonesia

Ahmad Fahrezi¹, Fahry Noer Salam², Gilang Mahardhika Ibrahim³, Rifki Rahman Syaiful⁴, Aries Saifudin⁵

^{1,2,3,4,5}*Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

E-mail: ¹fahrezi101014@gmail.com, ²fahrysalam42@gmail.com, ³gilangmahardhika18@gmail.com, ⁴ridkirahmansyaiful@gmail.com, ⁵aries.saifudin@unpam.ac.id

Abstrak- Aplikasi *Inventori* milik PT. Aino Indonesia merupakan sebuah aplikasi yang digunakan oleh bagian gudang untuk mencatat seluruh barang masuk dan barang keluar. Apabila terjadi error pada aplikasi ini akan menimbulkan masalah besar bagi bagian gudang karena pencatatan harus dilakukan manual dengan cara stock opname. Dalam menyelesaikan permasalahan ini, penulis melakukan testing menggunakan metode *black box* testing teknik *Equivalence Partitions*. Pengujian ini merupakan pengujian yang dilakukan dengan cara melakukan inputan secara acak dengan tolak ukur tertentu untuk kerberhasilannya. Dengan menggunakan rancangan *test case* penulis melakukan pengujian untuk memastikan apakah terdapat error pada proses *login* ke dalam aplikasi *Inventori* ini. Hasil dari pengujian ini adalah *system* dapat berjalan dengan baik dan tidak ditemukan *error*, sehingga bagian gudang dapat langsung menggunakan aplikasi ini untuk proses pencatatan barang masuk dan barang keluarnya.

Kata Kunci: *Black Box*, *Equivalence Partitions*, Aplikasi *Inventori*, *Test Case*, Gudang

Abstract- *Inventori application* belonging to PT. Aino Indonesia is an application used by the warehouse department to record all incoming and outgoing goods. If an error occurs in this application, it will cause big problems for the warehouse because the recording must be done manually by means of stock taking. In solving this problem, the author conducted testing using the *black box* testing method of the *Equivalence Partitions* technique. This test is a test carried out by performing random input with certain benchmarks for success. By using a test case design, the author conducts a test to determine whether there is an error in the process of logging into this *Inventori* application. The expected result of this test is that the system can run well and no errors are found, so that the warehouse department can directly use this application for the process of recording incoming and outgoing goods.

Keywords: *Black Box*, *Equivalence Partitions*, Application *Inventori*, *Test Case*, Warehouse

1. PENDAHULUAN

Software Testing atau pengujian perangkat lunak merupakan sebuah proses pengekseskuan sebuah program atau aplikasi yang telah dikembangkan oleh pengembang atau developer. Pengujian perangkat lunak sendiri pada umumnya dilakukan untuk menemukan kendala seperti bug pada suatu perangkat lunak demi memastikan kualitas dari perangkat lunak tersebut. Pengujian perangkat lunak sangat penting dilakukan, karena biasanya pada setiap pembuatan aplikasi akan terdapat hal yang terlewat hingga pengujian sebuah aplikasi perangkat lunak sangatlah penting untuk dilakukan (Cholifah et al., 2018).

Software yang akan diuji adalah sebuah aplikasi pencatatan *Inventori* berbasis *website* milik PT. Aino Indonesia. Aplikasi ini digunakan oleh PT. Aino Indonesia untuk melakukan pencatatan jumlah barang masuk dan juga barang keluar setiap harinya sehingga diharapkan dapat meningkatkan efisiensi kerja di bagian gudang penyimpanan. Jika aplikasi ini mengalami masalah atau error saat melakukan login maka dapat mengakibatkan masalah bagi bagian gudang.

Bagi bagian gudang, jika karyawan atau staff tidak dapat login dan menjalankan aplikasi pencatatan *Inventori* ini kerugian yang akan muncul adalah kerugian waktu kerja dimana karyawan atau staff bagian gudang harus melakukan stock opname secara manual sehingga menyita waktu dan tenaga yang sangat banyak serta menurunkan efisiensi kerja dibagian gudang.

Kegiatan pengujian dilakukan dengan menggunakan metode pengujian *Black Box* dimana pengujian dilakukan dengan cara mengamati hasil *input* dan *output* dari perangkat lunak yang akan diuji

tanpa membuka *script* kode dari perangkat lunak tersebut. *Black box Testing* adalah metode pengujian perangkat lunak yang meneliti fungsi (*Functional Testing*) dari aplikasi tanpa melihat ke dalam struktur internal atau kinerja aplikasi. Metode uji ini dapat diterapkan untuk hampir setiap tingkat pengujian perangkat lunak seperti unit, integrasi, sistem dan penerimaan (Febiharsa et al., 2018).

Rencana pengujian yang akan di uji pada aplikasi pencatatan *Inventori* berbasis web ini menggunakan metode pengujian *Black Box* dengan teknik *Equivalence Partitions*. Teknik ini merupakan salah satu teknik pengujian yang adal pada metode pegujian *Black Box* dimana pengujian perangkat lunak dilakukan dengan melakukan input kedalam aplikasi dengan bantuan kasus uji tertentu.

2. METODOLOGI

Pengujian wajib dilakukan dalam pembuatan sistem informasi dengan tujuan untuk memastikan apakah aplikasi yang dibangun telah memenuhi standar teknis dan bisnis yang diharapkan oleh developer sebelum di seberkan kepada user. Menurut (Wibisono & Baskoro, 2017), pengujian perangkat lunak adalah proses untuk mencari kesalahan pada setiap item perangkat lunak, mencatat hasilnya, mengevaluasi setiap aspek pada setiap komponen (sistem) dan mengevaluasi fasilitas-fasilitas dari perangkat lunak yang akan dikembangkan. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dieksekusi pada perangkat lunak dan kemudian keluar dari perangkat lunak dicek apakah telah sesuai yang diharapkan(Nurudin et al., 2019)

Pengujain *Black Box* sendiri dapat diartikan sebagai sebuah pengujian yang melakukan pendekatan pengujian untuk mengetahui apakah semua fungsi perangkat lunak telah berjalan semestinya sesuai dengan kebutuhan fungsional yang telah didefinisikan (Sulistyanto & SN, 2017).

Pada pengujian ini kami akan melakukan input dengan memasukan data secara acak untuk memastikan apakah system pada aplikasi pencatatan *Inventori* ini dapat memunculkan hasil yang sesuai dengan desain pengujian yang ada pada *test case* yang dibuat (Debiyanti et al., 2020). Pembuatan *test case* bertujuan untuk memastikan apakah sebuah system akan berjalan sesuai dengan tujuan awal dari pembuatannya dan juga mampu memberikan output atau respon yang sesuai saat diinputkan data yang valid maupun tidak valid(Christian et al., 2019).

Dalam melakukan pengujian akan dilakukan pengujian berdasarkan *test case* dari aplikasi yang diujikan menggunakan teknik *Equivalence Partitions*, kemudian akan dilakukan penginisialisasian dari hasil yang kemudian akan di dokumentasikan untuk hasil dari pengujian itu sendiri. Tabel *test case* digunakan sebagai bahan acuan berhasil atau tidaknya scenario pengujian. Berikut adalah gambar dari halaman input login aplikasi pencatatan *Inventori* milik PT. Aino Indonesia dan rabel *test case*.



Gambar. 1 Halaman Log in Aplikasi Inventori (Sumber : Data Pribadi)

Pada Gambar.1 terdapat rencana pengujian dimana akan dilakukan tesing pada kolom *username*, *test case* yang dilakukan yaitu dengan mengosongkan *username* kemudian menekan tombol *sign in*, tolak ukur yang diharuskan terjadi pada *test case* ini adalah *system* akan menolak *inputan* dan proses tidak berlanjut, kemudian akan diuji juga memasukan *username* yang valid namun tidak mengisikan *password*, pada *test case* ini tolak ukur keberhasilan yang diharapkan adalah *system* akan menolak *inputan* dan proses tidak akan berlanjut. Kemudian rencana pengujian juga akan dilakukan pada kolom *password*, *test case* yang dilakukan yaitu mengosongkan *username* dan menginputkan *password* yang valid, tolak ukur yang diharapkan pada *test case* ini adalah *system* akan menolak *inputan* dan proses tidak berlanjut.

Rencana pengujian selanjutnya pada Gambar. 1 adalah melakukan *inputan username* dengan data tidak valid bersamaan dengan *inputan password* dengan data yang valid, tolak ukur yang diharapkan pada pengujian ini adalah *system* akan menolak *inputan* dan proses tidak berlanjut. Skenario pengujian yang terakhir adalah menginputkan *username* dengan data yang valid bersamaan dengan menginputkan *password* dengan data yang valid, tolak ukur yang diharapkan pada pengujian ini adalah *system* akan menerikma *inputan* dan proses berlanjut ke halaman *dashboard* aplikasi.

Tabel 1. Rancangan *Test Case* Pengujian Login Aplikasi Inventori

ID	Action	Hasil yang diharapkan
T001	Mengosongkan kolom <i>username</i> dan <i>password</i> dan kemudian menekan tombol <i>sign in</i>	System akan menolak <i>inputan</i> dan muncul pesan peringatan, serta proses tidak berlanjut.
T001	Input “Admin” pada kolom <i>username</i> dan mengosongkan kolom <i>password</i> dan kemudian menekan tombol <i>sign in</i>	System akan menolak <i>inputan</i> dan muncul pesan peringatan, serta proses tidak berlanjut.
T003	Mengosongkan kolom <i>username</i> dan input “123” pada kolom <i>password</i> dan kemudian menekan tombol <i>sign in</i> .	System akan menolak <i>inputan</i> dan muncul pesan peringatan, serta proses tidak berlanjut.
T004	Input “Suherman” pada kolom <i>username</i> dan input “123” pada kolom <i>password</i> dan kemudian menekan tombol <i>sign in</i>	System akan menolak <i>inputan</i> dan muncul pesan peringatan, serta proses tidak berlanjut.
T005	Input “Admin” pada kolom <i>username</i> dan input “123” pada kolom <i>password</i> dan kemudian menekan tombol <i>sign in</i>	System menerima <i>inputan</i> dan halaman berpindah ke halaman <i>dashboard</i> .

3. HASIL DAN PEMBAHASAN

Setelah membuat rancangan pengujian dan melakukan pengujian didapatkan hasil dan kemudian hasil tersebut dicatat pada form pengujian. Setelah dicatat, hasil pengujian tersebut akan dianalisis dan ditark kesimpulannya berdasarkan dari penilaian antara keselarasan hasil yang didapat dengan tolak ukur yang diharapkan. Apabila hasil sesuai akan di berikan kesimpulan valid dan apabila hasil tidak sesuai maka akan diberikan kesimpulan tidak valid. Berikut adalah tabel kesimpulan dari hasil pengujian yang sudah kami lakukan berdasarkan rancangan *test case*.

Tabel 1.2 Hasil Pengujian Berdsarkan Rancangan *Test Case*

ID	SKENARIO PENGUJIAN	TEST CASE	HASIL YANG DIHARAPKAN	HASIL PENGUJIAN	KESIMPULAN
T001	Mengosongkan semua isi data login, lalu langsung klik tombol masuk	Nama : - Kata sandi :-	Sistem akan menolak akses login dan menampilkan pesan “mohon isi dulu nama admin dan kata sandi”	Sesuai harapan	valid
T002	Hanya mengisi nama admin dan mengosongkan kata sandi, lalu langsung mengklik tombol “masuk”.	Nama : admin Kata sandi : -	Sistem akan menolak akses login dan menampilkan pesan “mohon isi data yang salah satu masih kosong”	Sesuai harapan	valid
T003	Hanya mengisi data kata sandi dan mengosongkan data admin , lalu langsung mengklik “masuk”	Nama : - Kata sandi : 123	Sistem akan menolak akses login dan menampilkan pesan “mohon isi data yang salah satu masih kosong”	Sesuai harapan	valid
T004	Masukan username salah dengan password benar	Nama : Suherman Kata Sandi : 123	Sistem akan menolak akses login dan menampilkan pesan “username dan password salah”	Sesuai harapan	Valid
T005	Memasukan username benar dengan password yang benar	Nama : Admin Kata sandi : 123	Sistem akan menerima inputan, dan halaman berpindah ke halaman dashboard aplikasi	Sesuai harapan	Valid

Pengujian yang dilakukan pada aplikasi Inventori milik PT. Aino Indonesia dilakukan pada 1 form, yaitu form LOG IN dan di uji sebanyak 5 kali pengujian.

4. KESIMPULAN

Hasil dari pengujian aplikasi Inventori pada PT. Aino Indonesia dapat terlihat sebuah kesimpulan bahwa pengujian menggunakan metode *black box* sangatlah dibutuhkan, karena setiap pembuatan perangkat lunak haruslah diuji terlebih dahulu sebelum digunakan oleh *user*. Dari pengujian yang kami lakukan, didapatkan hasil bahwa seluruh *system* berjalan sesuai dengan *test case* yang telah dibuat sehingga aplikasi ini dinyatakan dapat digunakan dengan baik. Dengan demikian pengujian perangkat lunak sangatlah dibutuhkan demi memastikan dan mengoptimalkan pembangunan sebuah perangkat lunak atau aplikasi sebelum diserahkan kepada user untuk digunakan.

5. SARAN

Saran dari kami untuk peneliti selanjutnya, coba untuk mengembangkan kembali *test case* yang ada agar dapat menyaring *system* lebih banyak lagi, kemudian lakukan lah testing secara berulang dan dilakukan oleh beberapa orang agar dapat memastikan bahwa hasil yang didapat akan sama walaupun user dan device yang digunakan berbeda. Hasil dari pengujian ini diharapkan dapat memberikan gambaran yang cukup jelas bahwa pengujian dengan metode *black box* sangatlah dibutuhkan.

REFERENSI

- Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2018). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 3(2), 206. <https://doi.org/10.30998/string.v3i2.3048>
- Christian, I. C., Supianto, A. A., & Rokhmawati, R. I. (2019). Pengembangan Aplikasi Media Pembelajaran Berbasis Website Untuk Materi Laju Reaksi di Tingkat SMA (Studi Pada SMA Brawijaya Smart School Malang). *Jurnal Pengembangann Teknologi Informasi Dan Ilmu Komputer*, 3(7), 7051–7059.
- Debiyanti, D., Sutrisna, S., Budrio, B., Kamal, A. K., & Yulianti, Y. (2020). Pengujian Black Box pada Perangkat Lunak Sistem Penilaian Mahasiswa Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(2), 162. <https://doi.org/10.32493/informatika.v5i2.5446>
- Febiharsa, D., Sudana, I. M., & Hudallah, N. (2018). Uji Fungsionalitas (Blackbox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik dengan AppPerfect Web Test dan Uji Pengguna. *Joined Journal (Journal of Informatics Education)*, 1(2), 117. <https://doi.org/10.31331/joined.v1i2.752>
- Nurudin, M., Jayanti, W., Saputro, R. D., Saputra, M. P., & Yulianti, Y. (2019). Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 4(4), 143. <https://doi.org/10.32493/informatika.v4i4.3841>
- Sulistiyanto, H., & SN, A. (2014). Urgensi Pengujian pada Kemajemukan Perangkat Lunak dalam Multi Perspektif. *KomuniTi*, 6(1), 65–74.
- Wibisono, W., & Baskoro, F. (2002). Pengujian Perangkat Lunak Dengan Menggunakan Model Behaviour UML Waskitho Wibisono , Fajar Baskoro. *Juti*, 1(1), 43–50.