

Pengujian Black Box Berbasis Equivalence Partitions Pada Aplikasi Administrasi Penjualan Furniture

Yudha Ardiyansyah¹, Ahmad Muhadi², Renal Aji Pangestu³, Rino Bahtiar⁴, Aris Setiawan⁵, Joko Riyanto⁶

¹⁻⁶Universitas Pamulang; Jl. Raya Puspitek No. 46 buaran, serpong, Kota Tangerang Selatan. Provinsi Banten 15310. (021) 741-2566 atau 7470 9855

¹⁻⁶Jurusan Teknik Informatika, Fakultas Teknik, Universitas Pamulang

e-mail: ¹yudha.ardiyansyah07@gmail.com, ²ahmadmuhadi62@gmail.com, ³renalaji08@gmail.com, ⁴rinbahtiar357@gmail.com, ⁵aris.seti990@gmail.com, ⁶jokoriyanto@unpam.ac.id..

Abstrak

Demi menjamin aplikasi (*software*) Administrasi Penjualan Furniture agar bebas dari *error* maka perlu dilakukan pengujian pada kualitas aplikasi. Aplikasi ini memiliki masalah ketika menginput data yang terkadang tidak bisa menginput data secara berulang-ulang kecuali kita refresh terlebih dahulu, maka aplikasi akan diuji dengan Metode *Black Box* berbasis *Equivalence Partition* secara menyeluruh berkenaan dengan penggunaan, manfaat, dan hasil yang didapati dari pemanfaatan perangkat lunak tersebut. Dalam pengujian perangkat lunak Seleksi Promosi Kenaikan Jabatan menggunakan Metode *Weight Product* terdiri dari 5 form. Metode *Black Box* berbasis *Equivalence Partition* menguji kualitas aplikasi yang akan dilakukan dokumentasi pengujian perangkat lunak dengan ditemukannya kesalahan pada setiap form yang dibagi menjadi tiga model kesalahan, yaitu kesalahan pada fungsi, struktur data dan *interface*. Hasil dari pengujian ini, untuk menjamin kualitas aplikasi Administrasi Penjualan *Furniture* bebas dari kesalahan. Agar dapat digunakan dengan nyaman tanpa adanya kesalahan pada saat menggunakannya.

Kata kunci: Pengujian, Software, Black Box, Equivalence Partition

I. PENDAHULUAN

Pengujian perangkat lunak merupakkan hal yang perlu kita lakukan untuk mendapatkan informasi tentang kualitas dari perangkat lunak yang telah kita buat, pengujian perangkat lunak harus dilakukan setelah membuat sebuah perangkat lunak agar kita dapat mendeteksi adanya kesalahan pada perangkat lunak yang dapat menyebabkan perangkat lunak tidak berfungsi dengan baik. Tujuan utama dari pengujian perangkat lunak untuk mencari kesalahan yang ada dari perangkat lunak. Hal ini diperlukan untuk menjamin bawah sofware yang telah kita buat dapat berjalan dengan baik tanpa ada kegagalan sama sekali. Pada proses pengujian perangkat lunak setiap kasus yang di uji harus memiliki identitas dan saling terhubung antara sekumpulan masukkan dengan hasil yang dinginkan. *Testing* dan *Debugging* adalah aktivitas yang berbeda tetapi *debugging* harus diakomodasikan pada setiap strategi testing. Jika ingin mengimplementasikan strategi testing software yang sukses Tom Gilb menyarankan prosedur yang digunakan adalah (Hidayat & Muttaqin, 2018):

- 1) Menetapkan seluruh kebutuhan produk *software* dalam perhitungan sebelum memulai testing.
- 2) Status obyek testing harus jelas.
- Memahami pengguna software dan mengembangkan sebuah profil untuk setiap kategori user.
- 4) Mengembangkan rencana testing yang menekankan pada "*Rapid Cycle Testing*".
- 5) Membangun *software* yang sempurna yang didesain untuk menguji dirinya sendiri.
- 6) Menggunakan tinjauan ulang yang formal sebagai filter sebelum pengujian.



- 7) Melakukan tinjauan ulang secara formal untuk menilai strategi tes dan kasus tes itu sendiri.
- 8) Mengembangkan pendekatan peningkatan yang berkelanjutan untuk proses *testing*.

Program yang akan kita uji pada pengkajian ini merupakan Aplikasi Administrasi Penjualan Furniture dengan metode Blackbox Testing, di mana tujuan dari pengujian ini untuk melihat program tersebut dapat berjalan sesuai dengan tugas program tersebut tanpa mengetahui kode program yang dipakai. Tahapan pertama pada pengujian dengan menggunakan Blackbox Testing adalah mengidentifikasi masukkan lalu diuji agar kita mengetahui letak dari kesalahannya. Pengujian menggunakan *Blackbox Testing* merupakan sebuah pengujian yang digunakan untuk melengkapi pengujian sebelumnya yaitu Whitebox Testing agar aplikasi yang kita buat memiliki kualitas yang baik serta waktu yang digunakan akan lebih efektif, sehingga dapat menguntungkan bagi perusahaan (Sinulingga et al., 2020). Equivalence Partitions merupakan sebuah pengujian berdasarkan masukkan data pada setiap form yang ada pada Aplikasi Administrasi Penjualan Furniture, setiap menu yang dimasukkan akan dilakukan pengujian dan dikelompokkan berdasarkan fungsinva baik itu bernilai valid ataupun tidak valid (Sinulingga et al., 2020).

II. METODE PELAKSANAAN

Pengujian sistem merupakan langkah terpenting dalam pembuatan sistem. pengujian sistem dilakukan karena setiap orang pasti bisa membuat kesalahan apa lagi saat membuat sistem yang rumit. Langkah ini dilakukan untuk menguji apakah sistem yang telah kita buat sudah layak untuk digunakan dan sesuai dengan kebutuhan, sehingga sangat perlu dilakukan pengujian untuk mengurangi kesalahan pada sistem yang dibangun yang dapat menyebabkan kerugian. Black box testing merupakan teknik pengujian software yang fokus pada spesifikasi fungsi-fungsi yang ada papa perangkat lunak yang dikembangkan. Black box testing cenderung dapat menemukan beberapa hal seperti fungsional yang tidak benar atau tidak ada, kesalahan struktur data, kesalahan akses basis data, kesalahan antar muka, kesalahan performance serta kesalahan inisialisasi dan terminasi (Rahadi & Vikasari, 2020) Proses Black Box Testing dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan (Ningrum et al., 2020).

Equivalence partitioning merupakan salah satu metode black box testing yang akan membagi domain input dari suatu program ke dalam kelas-kelas data, dimana test case dapat diturunkan. Metode black box equivalence partitioning berdasarkan pada premis masukan dan keluaran dari suatu komponen yang dipartisi ke dalam kelas-kelas, menurut spesifikasi dari komponen tersebut, yang akan diperlakukan sama (ekuivalen) oleh komponen tersebut, dan dapat diasumsikan bahwa masukan yang sama akan menghasilkan respon yang sama (Rahadi & Vikasari, 2020). Dalam tahap pengujian Equivalence Partitions yang dilakukan adalah sebagai berikut:

- 1. Buatlah suatu rancangan *test case* yang berdasarkan fungsi di dalam pengujian perangkat lunak.
- 2. Buatlah batasan pengujuan *Equivalence Partitions*.
- 3. Buatlah model pengujian berdasarkan skenario pengujian dan hasil yang diharapkan.
- 4. Melakukan pengujian berdasarkan rancangan model yang telah dibuat.



Gambar 2. 1 Tampilan Form Login

Pengujian ini dimulai dengan *Form Login*. Rencana pengujian *Username* data akan valid jika data yang dimasukan sama dengan data yang ada di dalam *database*, karena sistem pada *form login* ini harus memasukan data yang sama dengan data yang sudah tersimpan di dalam *database* jika sebaliknya maka tidak akan bisa masuk kedalam aplikasi. Begitu juga dengan *Password* yang dimasukan harus sama dengan yang sudah ada di *database*.

Tabel 2. 1 Rancangan Test Case Form Login

ID	Deskripsi Pengujian	Hasil yang diharapkan
L01	Menginput <i>Username</i> dengan	Akses <i>Login</i> berhasil, lalu



	"yuda22" dan	sistem akan
	Password	pindah kedalam
	"kelompok3" dengar	n tampilan <i>Home</i> .
	ketentuan data	
	tersebut sudah	
	tersimpan di	
	database kemudian	
	klik tombol Masuk.	
	Menginput	Akses Login
	Username dengan	Gagal, sistem
	"aji66" dan	tidak dapat
1.02	Password	membuka menu
	"ayahadi21" dengan	Home dan
L02	ketentuan data	tampilan tetap
	tersebut belum	pada form <i>Login</i> .
	tersimpan di	
	database kemudian	
	klik tombol Masuk.	
File Form Report	-	- 0 X
0 &		
	MANAGE PRODUK BARANG	
ID Barang	K103	🗸 Tambah 🔍 Delete
Nama Barang	Dring Chair	🔓 Update 🛛 🛈 Resit
Harga Barang	50000	
Status Barana	Jo	
Juniously		
No 1	Idiproduk Nama produk Harga produk K011 Lourse Chain 610300	Jumish Stok preduk Status produksi 35 Tercedia

Gambar 2. 2 Tampilan Form Produk Barang

Pengujian berikutnya dilakukan pada *form* Produk Barang, pada *form* ini kita akan merencanakan pengujian pada *field* harga barang dan jumlah stok barang karena pada kedua field itu merupakan *type integer* dimana data yang dimasukan harus berupa angka agar *valid*, dan sebaliknya jika diisi dengan huruf misal "abc" maka tidak akan *valid* atau tidak akan bisa tersimpan ke dalam sistem. Disini kita akan menguji *field* harga barang dengan menambahkan "Rp" dibagian depan, lalu pada jumlah stok barang akan di isi dengan kata "Tujuh" dan kita akan lihat apakah data dapat tersimpan atau tidak.

ID	Deskripsi Pengujian	Hasil Yang Diharapkan
B01	Pilih id "K003", lalu	Berhasil, dan
	Nama barang	data akan
	"Dining Chair"	tersimpan pada
	Mengisi Harga	<i>database</i> lalu

	1	[
	barang "500000"	data secara
	dan jumlah stok	otomatis akan
	barang "35" dengan	tampil pada table
	status barang	alternative.
	"Tersedia" jika	
	sudah terisi semua	
	selanjutnya kita	
	tekan tambah.	
B02	Pilih id "M001", lalu	Gagal,dan data
	Nama barang "Bar	tidak akan
	Table" Mengisi	tersimpan
	Harga barang "Rp.	kedalam database
	3000000" dan	serta data tidak
	jumlah stok barang	akan tampil pada
	"5" dengan status	table alternative.
	barang "Tersedia"	
	jika sudah terisi	
	semua selanjutnya	
	kita tekan tambah.	
B03	Pilih id "L002", lalu	Gagal,dan data
	Nama barang	tidak akan
	"Lemari Document"	tersimpan
	Mengisi Harga	kedalam database
	barang "800000"	serta data tidak
	dan jumlah stok	akan tampil pada
	barang "Tujuh"	table alternative.
	dengan status barang	
	"Tersedia" jika	
	sudah terisi semua	
	selanjutnya kita	
	tekan tambah.	
Ele Form Report		- 0 ×
	MANAGE DATA PERUVALAN	
10 Transalesi 011 10 Barang KD	Junitali Produk 2 83 Totid Harge 100000	Update O Reset
Nama Sarang Dir	ing Chair Targgal Tomoko Oct 5, 2021	

Gambar 2. 3 Tampilan Form Data Penjualan

Rancangan berikutnya dilakukan pada *form* Data Penjualan, pada *form* ini kita akan merencanakan pengujian pada *field* Id Transaksi, Harga barang, jumlah produk dan total harga dimana pada *field-field* itu merupakan *type integer* dimana data yang dimasukan harus berupa angka, dan sebaliknya jika diisi dengan huruf misal "abc" atau simbol maka tidak akan *valid*



atau tidak akan bisa tersimpan ke dalam sistem. Disini kita akan menguji *field* Id transaksi dengan memasukan huruf misal "abc" begitu juga field yang lain untuk melihat apakah akan berhasil tersimpan atau tidak.

Tabel 2.3	Rancangan	Test	Case	Form	Data	Peniualan

Ш	Dockringi Donguijan	Hasil Yang
ID	Deskripsi Pengujian	Diharapkan
P01	Memasukan id	Berhasil, dan
	dengan "010003",	data akan
	lalu Pilih id "K003",	tersimpan pada
	lalu Nama barang	database lalu
	"Dining Chair"	data secara
	Mengisi Harga	otomatis akan
	barang "500000"	tampil pada <i>table</i>
	dan jumlah barang	alternative.
	"2" dengan total	
	harga "1000000",	
	lalu masukan tanggal	
	transaksi "Oct	
	5,2021" jika sudah	
	terisi semua	
	selanjutnya kita	
	tekan tambah.	
P02	Memasukan id	Gagal,dan data
	dengan "T003", lalu	tidak akan
	Pilih id "K003", lalu	tersimpan
	Nama barang	kedalam
	"Dining Chair"	database serta
	Mengisi Harga	data tidak akan
	barang "500000"	tampil pada <i>table</i>
	dan jumlah barang	alternative.
	"2" dengan total	
	harga "1000000",	
	lalu masukan tanggal	
	transaksi "Oct	
	5,2021" jika sudah	
	terisi semua	
	selanjutnya kita	
	tekan tambah.	
P03	Memasukan id	Gagal,dan data
	dengan "010003",	tidak akan
	lalu Pilih id "K003",	tersimpan
	lalu Nama barang	kedalam
	"Dining Chair"	database serta
	Mengisi Harga	data tidak akan
	barang "Rp. 500000"	tampil pada <i>table</i>
	dan jumlah barang	alternative.
	"2" dengan total	

	harga "Rp.	
	1000000", lalu	
	masukan tanggal	
	transaksi "Oct	
	5,2021" jika sudah	
	terisi semua	
	selanjutnya kita	
	tekan tambah.	
P04	Memasukan id	Gagal,dan data
	dengan "010003",	tidak akan
	lalu Pilih id "K003",	tersimpan
	lalu Nama barang	kedalam database
	"Dining Chair"	serta data tidak
	Mengisi Harga	akan tampil pada
	barang "500000"	table alternative.
	dan jumlah barang	
	"Dua" dengan total	
	harga "1000000",	
	lalu masukan tanggal	
	transaksi "Oct	
	5,2021" jika sudah	
	terisi semua	
	selanjutnya kita	
	tekan tambah.	
P05	Memasukan id	Berhasil, dan
	dengan "030004",	data akan
	lalu Pilih id "L004",	tersimpan pada
	lalu Nama barang	<i>database</i> lalu
	"Lemari Buku"	data secara
	Mengisi Harga	otomatis akan
	barang "2000000"	tampil pada <i>table</i>
	dan jumlah barang	alternative.
	"2" dengan total	
	harga "4000000",	
	lalu masukan tanggal	
	transaksi "Sep	
	8,2021" jika sudah	
	terisi semua	
	selanjutnya kita	
	tekan tambah.	

III. HASIL DAN PEMBAHASAN

Berdasarkan Rancangan Uji kasus yang sudah dilakukan pada Tabel 2.1, 2.2 dan 2.3, maka akan mendapatkan Hasil Pengujian. Hasilnya dapat dilihat pada Tabel 3.1 Hasil Pengujian dibawah ini.



٦

	Des	Hasil	Hasil	
П	kripsi	yang	Deng	Kesim
Ш	Peng	Diharap	r eng	pulan
	ujian	kan	ujian	
L01	Meng	Akses	Sistem	Sesuai
	input	Login	akan	
	Username	berhasil,	langsung	
	dengan	lalu	masuk	
	"yuda22"	sistem	kedalam	
	dan	akan	form	
	Password	pindah	Home	
	"kelom	kedalam		
	pok3"	tampilan		
	dengan	Home.		
	ketentuan			
	data			
	tersebut			
	sudah			
	tersimpan			
	di			
	database			
	kemudian			
	klik			
	tombol			
	Masuk.			
L02	Meng	Akses	Sistem	Sesuai
	input	Login	akan me	
	Username	Gagal,	nampil	
	dengan	sistem	kan	
	"aji66"	tidak	pesan	
	dan	dapat	"user	
	Password	mem	name	
	"ayah	buka	atau	
	adi21"	menu	pass	
	dengan	Home	word	
	ketentuan	dan	salah"	
	data	tampilan	dan	
	tersebut	tetap	akan	
	belum	pada	tetap	
	tersimpan	form	pada	
	di	Login.	form	
	database		Login	
	kemudian			
	klik			
	tombol			
	Masuk.			

B01	Pilih id	Berhasil	Data	Sesuai
DUI	"K003"	dan data	berhasil	Debuui
	lalu Nama	akan ter	ter	
	harang	simpan	simnan	
	"Dining	nada	di dalam	
	Chair"	datahase	datahase	
	Mengisi	lalu data	lalu akan	
	Harga	secara	muncul	
	harang	otomatis	nesan	
	"500000"	akan	"data	
	dan	tampil	berhasil	
	iumlah	nada	ter	
	stok	table	simnan"	
	harang	alter	dan data	
	"35"	native	akan	
	dengan	<i>nunve</i> .	muncul	
	status		nada	
	harang		table	
	"Tersedia		alter	
	" iika		native	
	sudah		11011110	
	terisi			
	semua			
	selanjut			
	nya kita			
	tekan			
	tambah.			
B02	Pilih id	Gagal,	Data	Sesuai
	"M001",	dan data	gagal ter	
	lalu Nama	tidak	simpan	
	barang	akan ter	lalu akan	
	"Bar	simpan	muncul	
	Table"	kedalam	pesan	
	Mengisi	database	"terjadi	
	Harga	serta data	ke	
	barang	tidak	salahan	
	"Rp.	akan	pada	
	3000000"	tampil	database	
	dan	pada	" dan	
	jumlah	table	data	
	stok	alter	tidak	
	barang	native.	akan	
	"5"		muncul	
	dengan		pada	
	status		table	
	barang		alter	
	"Tersedia		native	
	" jika			



	sudah			
	terisi			
	semua			
	selanjut			
	nya kita			
	tekan			
	tambah.			
B03	Pilih id	Gagal,	Data	Sesuai
	"L002",	dan data	gagal ter	
	lalu Nama	tidak	simpan	
	barang	akan ter	lalu akan	
	"Lemari	simpan	muncul	
	Document	ke	pesan	
	" Mengisi	dalam	"teriadi	
	Harga	database	ke	
	harang	serta data	salahan	
	"800000"	tidak	nada	
	dan	akan	database	
	jumlah	tampil	" dan	
	stok	nada	data	
	barang	table	tidak	
	"Tuinh"	alter	akan	
	dengan	nativa	akali	
	atotua	nuiive.	nada	
	borong		table	
	"Tersedia		alter	
	" iika		nativa	
	JIKa		nuiive	
	torigi			
	sellua			
	nya Kita			
D01	tamban.	D. 1 .1	Det	Q
P01	Memasu	Berhasil,	Data	Sesuai
	Kan 1d	aan data	pernasil	
	dengan	akan ter	ter	
	~010003″,	sımpan	sımpan	
	Ialu Pilih	pada	di dalam	
	1d	database	database	
	"К003",	lalu data	lalu akan	
	lalu Nama	secara	muncul	
	barang	otomatis	pesan	
	"Dining	akan	"data	
	Chair"	tampil	berhasil	
	Mengisi	pada	ter	
	Harga	table	simpan"	
	barang	alter	dan data	

	"500000"	native.	akan	
	dan		muncul	
	jumlah		pada	
	barang		table	
	"?"		alter	
	dengan		nativo	
	total		παιινε	
	total			
	narga			
	", lalu			
	masukan			
	tanggal			
	transaksi			
	"Oct			
	5,2021"			
	jika sudah			
	terisi			
	semua			
	selanjut			
	nva kita			
	tekan			
	tambah			
D02	Memosu	Gagal	Data	Securi
102	kon id	dan data	Data goggi tor	Sesual
	dan gan	uali uata	gagai tei	
	"TOO2"	uuak	simpan	
	1003,			
		simpan	muncui	
	10	kedalam	pesan	
	"K003",	database	terjadi	
	lalu Nama	serta data	ke	
	barang	tidak	salahan	
	"Dining	akan	pada	
	Chair"	tampil	database	
	Mengisi	pada	" dan	
	Harga	table	data	
	barang	alter	tidak	
	"500000"	native.	akan	
	dan		muncul	
	jumlah		pada	
	barang		table	
	"2"		alter	
	dengan		native	
	total			
	harga			
	1101 ga 1111 ga			
	" loly			
	, idiu			
	magultar			
	masukan			



	transaksi]	P04	Memasu	Gagal,	Data	Sesuai
	"Oct						kan id	dan data	gagal ter	
	5,2021"						dengan	tidak	simpan	
	iika sudah						"010003".	akan ter	lalu akan	
	terisi						lalu Pilih	simpan	muncul	
	semua						id	ke dalam	pesan	
	selaniut						"K003".	database	"teriadi	
	nva kita						lalu Nama	serta data	ke	
	tekan						harang	tidak	salahan	
	tambah						"Dining	akan	nada	
P03	Memasu	Gagal	Data	Securi			Chair"	tamnil	database	
105	kon id	dan data	Data gogel tor	Sesual			Mongisi	nada	" dan	
	dangan	uall uata	gagai tei				Harga	table	data	
	uengan "010002"		simpan				horong	alter	uala tidal	
	010003,	akan ter					Uarang	uller	uuan	
			muncui				300000	nauve.		
	10	ke dalam	pesan				dan		muncul	
	"K003",	database	terjadi				jumian		pada	
	lalu Nama	serta data	ke				barang		table	
	barang	tidak	salahan				"Dua"		alter	
	"Dining	akan	pada				dengan		native	
	Chair"	tampil	database				total			
	Mengisi	pada	" dan				harga			
	Harga	table	data				"1000000			
	barang	alter	tidak				", lalu			
	"Rp.	native.	akan				masukan			
	500000"		muncul				tanggal			
	dan		pada				transaksi			
	jumlah		table				"Oct			
	barang		alter				5,2021"			
	"2"		native				jika sudah			
	dengan						terisi			
	total						semua			
	harga						selanjut			
	"Rp.						nya kita			
	1000000"						tekan			
	, lalu						tambah.			
	masukan					P05	Memasu	Berhasil,	Data	Sesuai
	tanggal						kan id	dan data	berhasil	
	transaksi						dengan	akan ter	ter	
	"Oct						"030004".	simpan	simpan	
	5,2021"						lalu Pilih	pada	di dalam	
	iika sudah						id	database	database	
	terisi						"L004".	lalu data	lalu akan	
	semua						lalu Nama	secara	muncul	
	selaniut						harang	otomatis	nesan	
	nya kita						"Lemari	akan	"data	
	tekan						Buku"	tampil	herhacil	
	tambab						Mongini	nada	tor	
	tamball.]		mengisi	paua	101	



Harga	table	simpan"	
barang	alter	dan data	
"2000000	native.	akan	
" dan		muncul	
jumlah		pada	
barang		table	
"2"		alter	
dengan		native	
total			
harga			
"4000000			
", lalu			
masukan			
tanggal			
transaksi			
"Sep			
8,2021"			
jika sudah			
terisi			
semua			
selanjut			
nya kita			
tekan			
tambah.			

Jumlah percobaan yang diuji ada 10, dengan masingmasing percobaan yang diuji yaitu *form login* diuji sebanyak 2 kali, form produk barang diuji sebanyak 3 kali dan *form* data penjualan diuji sebanyak 5 kali. Adapun fungsi tombol yang juga diuji yaitu ada 4 fungsi yaitu fungsi masuk, tambah, *update, delete*. Dengan masing-masing fungsi yang diuji yaitu masuk 2 kali, *update* 2 kali, *delete* 2 kali, tambah 8 kali dengan total fungsi yang dicoba yaitu 14 kali. Jumlah *error* yang ditemukan sebanyak 0 fungsi dan fungsi yang tidak ditemukan *error* 4 fungsi.

IV. SIMPULAN

Pengujian aplikasi dengan *Blackbox Testing* bertujuan mencoba menjalankan program untuk mengetahui sistem kerja program tersebut tanpa harus mengetahui kode program yang dipakai. Berdasarkan pengujian kualitas aplikasi *furniture*, dapat ditarik kesimpulan bahwa pengujian dengan metode *Black Box* berbasis *Equivalence Partitions* dapat membantu proses pembuatan case pengujian, uji kualitas dan menemukan kesalahan yang tidak terdeteksi yang disebabkan oleh kesalahan pengetikan pada code aplikasi atau database. Dalam pengujian aplikasi *furniture* tidak ditemukan kesalahan dalam fungsi tombol maupun percobaan yang dilakukan pada data yang sudah di masukan ke dalam aplikasi *furniture* ini. Untuk saat ini aplikasi furniture sudah dicoba dijalankan dengan hasil sebagaimana fungsinya, sehingga Diharapkan aplikasi ini dapat digunakan atau dikembangkan agar bisa bermanfaat untuk menyimpan data produk dan penjualan *furniture*.

DAFTAR PUSTAKA

- Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis. Jurnal Teknik Informatika UNIS JUTIS, 6(1), 2252–5351. www.ccssenet.org/cis
- Ningrum, F. C., Suherman, D., Aryanti, S., & Prasetya, H. A. (2020). Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions. 4(4), 125–130.
- Rahadi, N. W., & Vikasari, C. (2020). Pengujian Software Aplikasi Perawatan Barang Milik Negara Menggunakan Metode Black Box Testing Equivalence Partitions. Infotekmesin, 11(1),57–61. https://doi.org/10.35970/infotek mesin. v11i1.124
- Sinulingga, A. R., Zuhri, M., Mukti, R. B., Syifa, Z., & Saifudin, A. (2020). Pengujian Black Box pada Sistem Aplikasi Informasi Data Kinerja Menggunakan Teknik Equivalence Partitions. Jurnal Teknologi Sistem Informasi Dan Aplikasi, 3(1), 9. https://doi.org/10.32493/jtsi. v3i1.4303