

PENGUJIAN PADA APLIKASI MANAJEMEN ASET MENGGUNAKAN BLACK BOX TESTING

Fenti Agil Sakinah, Firza Prima Aditiawan, Afina Lina Nurlaili
Teknik Informatika, Universitas Pembangunan Nasional Veteran Jawa Timur
fentiagil@gmail.com

ABSTRAK

Aplikasi manajemen aset merupakan alat yang penting bagi organisasi untuk memantau dan mengelola aset inventaris. Pada penelitian ini, perusahaan telah mengembangkan aplikasi manajemen aset sesuai dengan kebutuhannya. Penelitian ini dilakukan pada aplikasi manajemen aset salah satu perusahaan BUMN. Dari pernyataan perusahaan, terlihat bahwa pengujian aplikasi ini sangat diperlukan untuk memastikan bahwa aplikasi tersebut beroperasi dengan baik dan bebas dari kesalahan. Pengujian aplikasi manajemen aset menggunakan Black Box Testing menjadi kunci penting sebelum aplikasi dipublikasikan untuk pengguna. Tujuan dari pengujian ini adalah untuk menguji fungsionalitas aplikasi dari perspektif pengguna tanpa memerlukan pemahaman teknis aplikasi. Metode pengujian yang digunakan adalah Black Box Testing dengan teknik Boundary Value Analysis dan Equivalence Partitioning. Hasil pengujian menunjukkan bahwa dari 227 skenario pengujian, 226 sesuai dengan harapan dan terdapat satu kesalahan ditemukan dengan persentase kesuksesan kasus uji sebesar 99 persen. Penemuan kesalahan ini menunjukkan bahwa keberlanjutan dan kesempurnaan aplikasi sebelum digunakan oleh pengguna akhir membutuhkan pengujian mendalam menggunakan teknik Black Box Testing.

Kata kunci : Black Box Testing, Aplikasi manajemen aset, Pengujian Perangkat Lunak

1. PENDAHULUAN

Aplikasi manajemen aset dibuat dan digunakan oleh organisasi untuk melacak dan mengelola aset inventarisnya[1]. Dalam hal ini penelitian dilakukan di PT Petrokima Gresik, dimana saat ini perusahaan tersebut telah membuat dan merancang aplikasi manajemen aset untuk kebutuhan perusahaan. Berdasarkan pernyataan dari pihak perusahaan, diperlukan adanya pengujian aplikasi manajemen aset ini guna memastikan aplikasi tersebut berjalan dengan baik dan tidak ditemukan kesalahan. Pengujian aplikasi ini diharapkan dapat mengukur kinerja aplikasi dalam memenuhi fungsinya.

Pengujian perangkat lunak adalah kumpulan prosedur yang dimaksudkan untuk menemukan kesalahan pada setiap komponen sistem[2]. Pengujian *Black Box* dan *White Box* merupakan jenis dari pengujian perangkat lunak. Pengujian *Black Box* berfokus untuk memeriksa kinerja sistem, sedangkan pengujian *White Box* berfokus untuk menguji struktur sistem di dalamnya[2]. Pada penelitian ini penulis berfokus pada pengujian kinerja aplikasi tanpa melihat bagian internal aplikasi, oleh karena itu penulis menggunakan metode *Black Box* dibandingkan metode *White Box*. Dengan menemukan kesalahan sebelum aplikasi digunakan, diharapkan dapat mengurangi masalah serius yang mungkin muncul jika aplikasi tersebut bermasalah saat digunakan.

Beberapa teknik yang dapat digunakan dalam pengujian sistem dalam pengujian *Black Box* yaitu *Cause-Effect Graph*, *Boundary Value Analysis*, dan *Equivalence Partitioning*[3]. Bill Elmendorf dari IBM menciptakan teknik *Cause-Effect Graph* pada tahun 1972, Teknik ini berfokus pada penggunaan grafik di mana input dianggap sebagai penyebab dan output dianggap sebagai efek[4]*Equivalence Partitioning*

berfokus pada nilai masukan dan membaginya menurut fungsinya untuk menentukan apakah bernilai valid atau tidak[5]. *Boundary Value Analysis* berfokus untuk memeriksa input di sekitar batasan nilai valid yang diizinkan system dengan memilih elemen dalam kelas ekuivalen pada sisi batas kelas [6].

Penggunaan teknik *Boundary Value Analysis* dan *Equivalence Partitioning* pada pengujian *Black Box* digunakan penulis pada penelitian ini. Hal ini dikarenakan menurut Santi dkk. [6] teknik *Boundary Value Analysis* dapat digunakan sebagai pelengkap teknik *Equivalence Partitioning* dimana perangkat lunak yang dibuat tidak selalu memenuhi persyaratan fungsional pengguna. Pada setiap inputan aplikasi diuji dan dikelompokkan sesuai fungsi valid atau tidak menggunakan *Equivalence Partitioning*. Penggunaan teknik *Boundary Value Analysis* digunakan sebagai acuan batas input untuk memverifikasi dan validasi output berdasarkan inputan, system tidak bisa melakukan proses ke database jika inputan melebihi atau dibawah batasan yang ditentukan.

Pada penelitian sebelumnya telah dilakukan pengujian dengan kedua Teknik tersebut. Dalam penelitian yang dilakukan Sholeh dkk. [7] teknik *Boundary Value Analysis* dan *Equivalence Partitioning* pada pengujian website ukmbantul.com. Pada penelitian ini *Equivalence Partitioning* digunakan dalam menentukan nilai valid atau tidak suatu inputan, sedangkan *Boundary Value Analysis* sebagai acuan batas input yang diuji [7]

Berdasarkan teori dan masalah yang telah dijelaskan, penulis ingin menemukan kesalahan dan memastikan aplikasi manajemen aset PT Petrokima Gresik berjalan dengan benar dengan menerapkan teknik *Boundary Value Analysis* dan *Equivalence Partitioning* dalam Black Box Testing.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Santi dkk. melakukan penelitian pertama tentang Black Box Testing dengan menggunakan Boundary Value Analysis dan Equivalence Partitioning pada sistem informasi akademik Universitas Mataram. [6]. Data akademik universitas diolah melalui sistem ini. Karena sistem masih memiliki kekurangan, penulis mengujinya. Studi ini menemukan 80 kasus gagal dari 322 skenario. Oleh karena itu, persentase keberhasilan data uji adalah 75,16%.

Dalam penelitian kedua yang dilakukan oleh Hutapea dkk. [8], penulis membahas pengujian Black Box dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning* pada aplikasi homelab yang digunakan sebagai pendukung pembelajaran siswa di Universitas Telkom. 19 fungsi diuji dengan teknik *Boundary Value Analysis* dan 2 fungsi dengan teknik *Equivalence Partitioning*. Hasil penelitian ini dapat menunjukkan bahwa meskipun beberapa input data tidak sesuai dengan dokumen SRS, sistem dapat mengidentifikasi kesalahan tersebut dan menghentikan proses penyimpanan data.

Vikasari [9] melakukan penelitian ketiga. Dalam penelitian tersebut, penulis membahas pengujian *Black Box Testing* dengan menggunakan *Boundary Value Analysis* pada sistem informasi magang industri Politeknik Negeri Cilacap. Sistem ini digunakan untuk memproses data magang industri mahasiswa. Penelitian ini menguji enam kolom, masing-masing dengan empat data, total 24 data. Hasilnya, penulis menemukan satu data yang gagal dari kolom tahun magang, yang menunjukkan tingkat kesuksesan total pengujian sebesar 95%.

Penelitian keempat dilakukan oleh Wijaya dan Astuti [10]. Dalam penelitian ini, penulis menggunakan teknik *Equivalence Partitioning* untuk melakukan pengujian *Black Box* pada sistem informasi PT INKA (Persero) untuk menilai kinerja karyawan. Dalam penelitian ini, penulis memeriksa sebelas test case dan menemukan bahwa sistem tidak menemukan kesalahan fungsionalitas di setiap fiturnya. Ini menunjukkan bahwa teknik dapat membantu dalam pembuatan test case, menguji kinerja sistem, dan menemukan kesalahan input.

Dalam penelitian kelima, Putri [5] melakukan penilaian *Black Box* pada aplikasi manajemen kas dan inventaris di Yayasan Tunas Mulia. Penulis menggunakan *Equivalence Partitioning* untuk melakukannya. Hasil penelitian menunjukkan bahwa aplikasi tersebut tidak menunjukkan kesalahan, yang berarti bahwa aplikasi tersebut berhasil. Dengan demikian, penulis menyimpulkan bahwa *Equivalence Partitioning* dapat membantu dalam menyusun pengujian secara fungsionalitas dan menemukan kesalahan test case saat memasukkan data.

2.2. Aplikasi Manajemen Aset yang diuji

Aplikasi manajemen aset adalah sistem pengelolaan aset yang terintegrasi untuk PT

Petrokimia Gresik. Sistem ini mencakup proses pengajuan hingga penghapusbukuan asset.

2.3. Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah prosedur untuk menemukan kesalahan sistem secara menyeluruh, mencatat hasilnya, dan mengevaluasi setiap komponen yang dibuat [2]. Untuk memastikan bahwa program tidak mengalami kerugian, pengujian sangat penting untuk menemukan dan memperbaiki kesalahan [10]. Sehubungan dengan [2], Glen Myers memberikan garis besar aturan yang dapat digunakan dalam penjelasan pengujian perangkat lunak:

- Pengujian adalah proses menjalankan program dengan tujuan utama untuk mengidentifikasi kesalahan.
- Kasus yang memiliki kemungkinan kesalahan yang tinggi dianggap baik.
- Hasil pengujian menunjukkan kesalahan

Jadi, pengujian yang baik dapat menemukan kesalahan perangkat lunak dan data uji, yang memungkinkan penemuan kesalahan yang lebih cepat dan teliti. Pengujian perangkat lunak dapat dilakukan dengan dua metode: *Black Box Testing* menguji fungsionalitas perangkat lunak dan *White Box Testing* menguji semua komponen program, seperti data internal, *loop*, logika keputusan, dan jalur [2].

2.4. Black Box Testing

Black Box Testing merupakan pengujian yang menguji input dan output sesuai dengan spesifikasinya, sehingga tidak perlu memahami struktur internal *software* [4]. Untuk memastikan bahwa fungsionalitas aplikasi sudah memenuhi persyaratan, pengujian *Black Box* dilakukan untuk memverifikasi hasil eksekusi aplikasi berdasarkan data uji yang diberikan [11]. *Black Box Testing* dilakukan untuk menemukan kesalahan perangkat lunak agar tidak terjadi lagi atau untuk memperbaikinya jika terjadi [12]. Sebagaimana dinyatakan dalam [12], metode pengujian *Black Box* terdiri dari berbagai pendekatan, seperti:

- Teknik *Equivalence Partitioning*: Teknik yang membagi input data menjadi beberapa partisi atau kelompok yang setara.
- Teknik *Boundary Value Analysis*: Teknik ini berfokus pada penemuan kesalahan di sekitar batas input minimum dan maksimum.
- Teknik *Fuzzing*: *Fuzzing* menemukan bug atau gangguan dalam perangkat lunak dengan menggunakan data yang cacat atau tidak valid.
- Teknik *Cause-Effect Graph*: Teknik ini menggunakan grafik untuk menunjukkan hubungan antara penyebab dan efek dalam perangkat lunak.
- Teknik *Orthogonal Array Testing*: Digunakan ketika domain input agak kecil tetapi kompleks, terutama untuk pengujian skala besar.

- f. Teknik *All Pair Testing*: Tujuan teknik ini untuk menghasilkan pasangan kasus uji yang mencakup setiap kombinasi diskrit yang dapat dimungkinkan dari input parameter.

2.5. Boundary Value Analysis

Boundary Value Analysis menguji batas nilai ekstrim tertentu untuk mendapatkan nilai valid yang cukup relevan [6]. Pilihan elemen kelas ekuivalen pada sisi batas kelas menggunakan *Boundary Value Analysis* dilengkapi dengan *Equivalence Partitioning* [12]. Untuk meningkatkan efisiensi pengujian perangkat lunak, *Boundary Value Analysis* dapat diterapkan bersamaan dengan pendekatan pengujian tambahan seperti *Equivalence Partitioning* [13]. Metode untuk menentukan batas atas dan bawah didasarkan pada [6]:

- a. Nilai $\{a-1\}$, a , $\{a+1\}$, $\{b-1\}$, b , dan $\{b+1\}$ ada karena ada batasan pada nilai a dan b .
- b. Uji nilai sebanyak $\{n-1\}$, n , dan $\{n+1\}$ jika masukan menunjukkan bahwa $n = \text{nilai}$.
- c. Untuk output, dua metode sebelumnya digunakan untuk membuat tabel pengujian hasil dari nilai maks dan min.
- d. Jika bentuk data internal program, seperti ukuran buffer dan batas array, diuji dengan data masukan, program memiliki jangkauan.

2.6. Equivalence Partitioning

Equivalence Partitioning adalah metode pengujian yang memeriksa masukan dan keluaran data yang dikelompokkan berdasarkan fungsinya apakah valid atau tidak valid [12]. Untuk menentukan test case, *Equivalence Partitioning* melihat kondisi input seperti nilai numerik, kisaran nilai, kumpulan nilai yang terkait, atau kondisi Boolean [6]. *Equivalence Partitioning* mengevaluasi input dari setiap menu, dan beberapa menu dimasukkan ke dalam kelompok dan diklasifikasikan berdasarkan fungsinya [10]. Sebagai contoh, metode berikut dapat digunakan untuk menentukan kelas ekuivalensi [14]:

- a. Satu kelas mengandung nilai yang valid dan dua kelas mengandung nilai yang tidak valid jika kondisi input mengatur rentang nilai.
- b. Dalam hal kondisi mengatur himpunan, satu kelas akan dibuat sebagai ekuivalen untuk semua nilai yang termasuk dalam himpunan, dan dua kelas lainnya akan dibuat untuk nilai yang tidak termasuk dalam himpunan.
- c. Jika kondisi input boolean, akan ada dua kelas ekuivalen untuk nilai benar dan nilai salah.

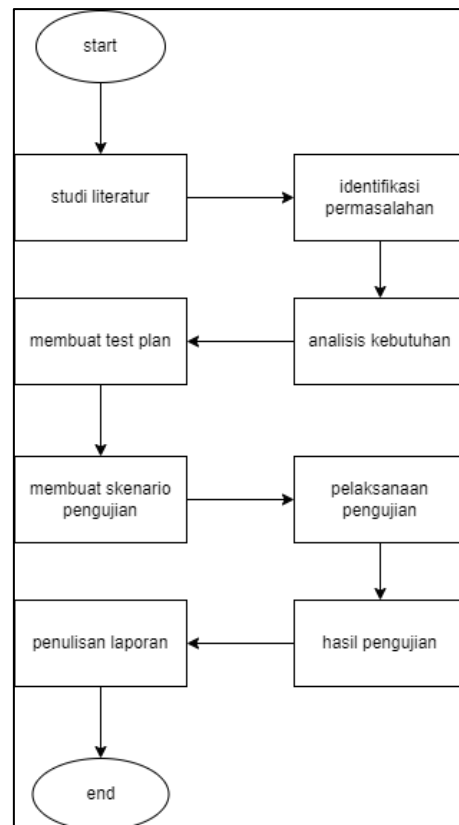
2.7. Skenario Pengujian

Serangkaian tindakan atau prosedur yang disebut skenario pengujian digunakan untuk menguji kinerja, fungsionalitas, dan keamanan perangkat lunak [15]. Skenario pengujian memiliki kemungkinan besar untuk menemukan kesalahan dengan waktu dan tenaga yang paling sedikit, sesuai dengan tujuan uji coba [16].

Menurut [16], desain skenario pengujian dapat dibuat dengan menggunakan berbagai teknik pengujian:

- a. *White Box Testing*, pengujian yang memastikan perangkat lunak bekerja sesuai dengan spesifikasinya.
- b. *Black Box Testing*, pengujian yang memastikan fungsi perangkat lunak bekerja dengan benar sesuai dengan kebutuhan fungsional yang telah didefinisikan.

3. METODE PENELITIAN



Gambar 1. Flowchart alur penelitian

Pada gambar diatas merupakan flowchart alur penelitian yang akan dilakukan oleh penulis, yaitu:

3.1. Studi literatur

Pada tahap ini, penulis mempelajari literatur untuk meningkatkan pemahaman tentang pengujian *Black Box* dengan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*. Penulis mengambil referensi dari majalah, buku, dan internet yang dapat diandalkan.

3.2. Identifikasi permasalahan

Proses identifikasi pertama melibatkan pengujian aplikasi AMS atau *Asset Management System* di PT. Petrokimia Gresik untuk menemukan masalah dan kesalahan

Proses pertama untuk mengidentifikasi masalah adalah menguji aplikasi manajemen aset PT.

Petrokimia Gresik untuk menemukan masalah dan kesalahan.

3.3. Analisis kebutuhan

Tujuan dari analisis kebutuhan adalah untuk menentukan kebutuhan dan tujuan utama dari aplikasi manajemen aset yang dibuat PT. Petrokimia Gresik. Saat menguji aplikasi, penulis dapat menggunakan kebutuhan dan tujuan utamanya sebagai acuan. Selanjutnya, aspek fungsional aplikasi akan dibandingkan dengan kebutuhan dan tujuan utama instansi yang dibuat sebelumnya.

3.4. Membuat test plan

Pada tahap ini penulis membuat *Test plan* yang mencakup gambaran teknik pengujian, strategi pengujian, dan jadwal pengujian yang akan dilakukan.

a. Teknik Pengujian

Metode *Black Box*, yang menggunakan teknik *Boundary Value Analysis* dan *Equivalence Partitioning*, digunakan untuk menguji aplikasi manajemen aset.

b. Strategi Pengujian

Strategi pengujian memeriksa fungsionalitas dan antarmuka sistem.

c. Jadwal

Pada tahap ini penulis membuat jadwal pengujian yang akan dilakukan beserta estimasi waktu yang diperlukan.

3.5. Membuat skenario pengujian

Skenario pengujian memungkinkan proses pengujian berjalan dengan cepat dan menghindari kehilangan waktu. Dokumen skenario pengujian juga berfungsi sebagai bukti pengujian fitur aplikasi manajemen aset. Hasil pengujian digunakan untuk mengevaluasi kinerja aplikasi dan menghitung efektivitasnya. Hasil ini digunakan sebagai dasar untuk memberikan rekomendasi perbaikan kepada perusahaan yang relevan untuk meningkatkan aplikasi. Skenario pengujian disusun berdasarkan fitur aplikasi yang memiliki form input dan berfokus pada masalah yang diidentifikasi.

3.6. Pelaksanaan pengujian

Pengujian *Black Box* dimulai dengan *Boundary Value Analysis* dan *Equivalence Partitioning* setelah skenario dibuat. Pengujian ini berkonsentrasi pada fungsionalitas aplikasi, tidak peduli apakah skenario menghasilkan hasil yang diharapkan. Aplikasi dapat dianggap sempurna jika kesalahan tidak ditemukan selama pengujian. Namun, jika kesalahan terus terjadi dan hasil yang diharapkan tidak tercapai, aplikasi harus ditingkatkan. Hasil pengujian ini mempengaruhi nilai kinerja yang selanjutnya dihitung untuk sampai pada kesimpulan tentang seberapa efektif aplikasi manajemen aset.

3.7. Hasil pengujian

Pada langkah ini, penulis menghitung efektivitas dari hasil yang diperoleh dari langkah sebelumnya. Hasil persentase kelayakan aplikasi adalah nilai efektivitas, yang menunjukkan seberapa efektif aplikasi bekerja.

a. Perhitungan Nilai Efektivitas Setiap Tabel

Sebelum menghasilkan nilai efektivitas aplikasi, penulis akan menghitung nilai efektivitas untuk setiap tabel. Setiap tabel mengandung sejumlah skenario pengujian, masing-masing dengan hasil yang berbeda. Menurut [6], untuk menentukan nilai efektivitas, rumus berikut digunakan:

$$\text{Nilai Efektivitas Tabel} = \left(\frac{\sum \text{Pengujian sesuai}}{\sum \text{Jumlah skenario pengujian}} \right) \times 100\% \quad (1)$$

di mana jumlah skenario yang berhasil atau sesuai dimasukkan, dibagi dengan jumlah skenario pengujian yang ada dalam tabel, dan kemudian dikalikan dengan 100 untuk menghasilkan persentase.

b. Perhitungan Nilai Efektivitas Secara Keseluruhan

$$\left(\frac{\text{Nilai tabel A} + \text{Nilai tabel B} + \dots \text{Nilai tabel n}}{\sum \text{Jumlah tabel skenario}} \right) \quad (2)$$

Setelah menghitung semua tabel, penulis akan menemukan nilai efektivitas total. Penulis akan melakukan ini dengan menggunakan rumus yang sama yang digunakan untuk menghitung rata-rata, nilai efektivitas dari setiap tabel akan dikumpulkan, kemudian dibagi dengan jumlah tabel skenario yang dibuat. Hasil perhitungan ini menghasilkan nilai efektivitas aplikasi manajemen aset. Nilai ini dihitung dengan menggunakan standar yang dikeluarkan oleh Litbang Depdagri (1991) untuk mengevaluasi apakah aplikasi tersebut memenuhi standar yang ditetapkan oleh penguji. Tabel berikut menunjukkan nilai-nilai ini.

Tabel 1. Standar acuan efektivitas berdasarkan litbang depdagri

Rasio Efektivitas	Tingkat Capaian
Di bawah 40	Sangat Tidak Efektif
40 – 59.99	Tidak Efektif
60 – 79.99	Cukup Efektif
Di atas 80	Sangat Efektif

Pada table diatas merupakan standar acuan efektifitas dalam pengujian aplikasi.

3.8. Penulisan laporan

Pada tahap ini, penulis menulis karya ilmiah yang berisi dokumentasi penelitian, hasil, dan kegiatan penelitian lainnya.

4. HASIL DAN PEMBAHASAN

Memuat hasil, Pengujian dan pembahasan tentang skripsi yang telah dilakukan.

4.1. Hasil pengujian

Pada tahap ini menunjukkan hasil pengujian dari aplikasi manajemen asset PT Petrokimia Gresik. Pengujian ini dilakukan berdasarkan skenario pengujian yang ada dan membandingkan hasil dengan hasil yang diharapkan. Tujuan dari pengujian ini adalah untuk menentukan apakah aplikasi tersebut sesuai dengan harapan atau tidak. Pada table 2 merupakan table yang berisi total jumlah scenario dan tabel pengujian tiap modulnya. Berikut merupakan salah satu hasil skenario pengujian pada tiap modulnya

Tabel 2. Total skenario pengujian

No	Modul pengujian	Jumlah tabel	Total skenario
1	Pengajuan Investasi	8 tabel	37 skenario
2	Pengajuan BASTBI	5 tabel	14 skenario
3	Pengelolaan Aset	4 tabel	15 skenario
4	Penghapusbukuan Aset	34 tabel	129 skenario
5	Penghapusbukuan Aset Tanah dan Bangunan	8 tabel	32 skenario
Total		59 tabel	227 skenario

a. Pengajuan Investasi

Tabel 3. Data uji halaman edit ajuan investasi by form

Data Input	Data Uji	
Lokasi Aset, Kode aset, dan Keterangan	<i>Equivalence Partitioning</i>	
	Partisi nilai invalid	Nilai 0
		Nilai lebih dari 500 (550)
	Partisi nilai valid	Nilai 1-500 karakter (15)
	<i>Boundary Value Analysis</i>	
	Nilai minimum, invalid (n-1)	Nilai 0
	Batas minimum yang valid	Nilai 1 dan 2
	Batas maksimum yang valid	Nilai 499 dan 500
	Nilai di atas batas maksimum, invalid (n+1)	Nilai 501
Perkiraan Umur Teknis	<i>Equivalence Partitioning</i>	
	Partisi nilai invalid	Nilai 0
		Nilai lebih dari 308 (350)
	Partisi nilai valid	Nilai 1-308 karakter (4)
	<i>Boundary Value Analysis</i>	
	Nilai minimum, invalid (n-1)	Nilai 0
	Batas minimum yang valid	Nilai 1 dan 2
	Batas maksimum yang valid	Nilai 307 dan 308
	Nilai di atas batas maksimum, invalid (n+1)	Nilai 309
Unit Kerja Pengguna, Klasifikasi, Cost Center, dan Estimasi Selesai	<i>Equivalence Partitioning</i>	
	Partisi nilai invalid	Dropdown kosong
	Partisi nilai valid	Dropdown terisi dengan salah satu opsi

Pada tabel diatas merupakan penggunaan metode *Equivalence Partitioning* dan *Boundary Value Analysis* pada setiap data input, yang akan digunakan dalam skenario pengujian.

Tabel 4. Hasil pengujian edit investasi by form

Test ID	Skenario	Expected Result	Result
SP001	Mengubah data investasi dengan menambahkan nilai valid ke form yang ada	1.Mampu mengubah data ajuan investasi 2. Dapat mengajukan dan menyimpan data ajuan investasi ke departemen PKM 3.Mampu menyimpan draft data ajuan investasi	Sesuai
SP002	Mengubah data investasi dengan mengosongkan salah satu form	Gagal mengubah data ajuan investasi dan muncul peringatan bahwa formulir harus diisi	Sesuai
SP003	Mengubah data investasi dengan mengosongkan semua form	Gagal mengubah data ajuan investasi dan muncul peringatan bahwa formulir harus diisi	Sesuai
SP004	Mengubah data investasi dengan menambah nilai di atas nilai validnya	Muncul notifikasi error karena kegagalan mengedit data ajuan investasi	Tidak Sesuai

Tabel di atas menunjukkan hasil tes pada formulir edit investasi menggunakan formulir 3 dari empat skenario tes yang dirancang sesuai dengan hasilnya.

Pada gambar 2 adalah tampilan form edit pengajuan investasi.

Gambar 2. Form edit pengajuan investasi

b. Penciptaan BASTBI

Tabel 5. Data uji halaman buat pengajuan BASTBI

Data Input	Data Uji	
Lokasi (free text)	Equivalence Partitioning	
	Partisi nilai invalid	Nilai 0
	Partisi nilai valid	Nilai lebih dari 1 (20)
	Boundary Value Analysis	
	Nilai minimum, invalid (n-1)	Nilai 0
Jenis Berita Acara, Dari Departemen, Kepada Departemen, Fund Center, Cost Center, Approver, Jenis Pengadaan, Aset tetap Sebelumnya, Barang Investasi, Cost Center Pengelola Terkecil, dan Perkiraan Umur Teknis	Equivalence Partitioning	
	Partisi nilai invalid	Dropdown kosong
	Partisi nilai valid	Dropdown terisi dengan salah satu opsi

Pada tabel diatas merupakan penggunaan metode *Equivalence Partitioning* dan *Boundary Value Analysis* pada setiap data input, yang akan digunakan dalam skenario pengujian.

Tabel 6. Hasil pengujian buat pengajuan BASTBI

Test ID	Skenario	Expected Result	Result
SP005	Mengisi semua form yang tersedia untuk membuat pengajuan.	Proses pembuatan draft dokumen BASTBI berhasil	Sesuai
SP006	Buat pengajuan dengan mengisi salah satu form yang ada.	Gagal membuat draft dokumen BASTBI dan muncul peringatan kolom yang harus diisi	Sesuai

Tabel di atas menunjukkan hasil tes pada form pengajuan BASTBI. Dua skenario tes yang dirancang sesuai dengan hasil tersebut.

Pada gambar 3 adalah tampilan form penciptaan BASTBI baru.

Gambar 3. Form penciptaan BASTBI

c. Pengelolaan Aset Tetap

Tabel 7. Data uji halaman import aset tetap

Data Input	Data Uji	
File aset tetap	Equivalence Partitioning	
	Partisi nilai invalid	File ajuan berupa file selain Excel
	Partisi nilai valid	File ajuan berupa file Excel dan data sesuai format.
	Partisi nilai invalid	File ajuan berupa file Excel, tetapi data tidak sesuai format.
		Ukuran file lebih dari 10 MB
	Boundary Value Analysis	
	Nilai minimum, invalid (n-1)	Nilai 0
	Batas minimum yang valid	Nilai 1 KB
	Batas maksimum yang valid	Nilai 10 MB
	Nilai di atas batas maksimum, invalid (n+1)	Nilai 11 MB

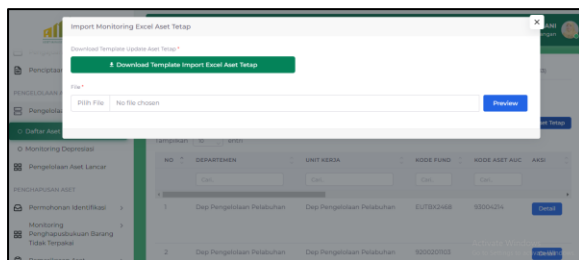
Pada tabel diatas merupakan penggunaan metode *Equivalence Partitioning* dan *Boundary Value*

Analysis pada setiap data input, yang akan digunakan dalam skenario pengujian.

Tabel 8. Hasil pengujian import asset tetap

Test ID	Skenario	Expected Result	Result
SP007	Mengupload file ajuan dengan data dan format Excel untuk melakukan import.	Sukses melakukan import aset tetap	Sesuai
SP008	Mengupload file ajuan dalam format yang berbeda dari Excel untuk melakukan import	Gagal mengimport aset tetap dan muncul notifikasi file tidak bisa diupload.	Sesuai
SP009	Melakukan import dengan mengupload file Excel ajuan, tetapi data tidak sesuai.	Gagal mengimport aset tetap dan tidak dapat menekan tombol "import".	Sesuai
SP010	Mengupload file ajuan dengan ukuran lebih dari 10 MB untuk melakukan import.	Gagal mengimport aset tetap dan muncul notifikasi file tidak bisa diupload.	Sesuai

Tabel di atas menunjukkan hasil tes pada halaman import aset tetap. Empat skenario tes yang dirancang sesuai dengan hasil tersebut.



Gambar 4. Import asset tetap

Pada gambar diatas adalah tampilan form import aset tetap.

d. Penghapusbukuan Aset BTT

Tabel 9. Data uji halaman disposisi permohonan identifikasi

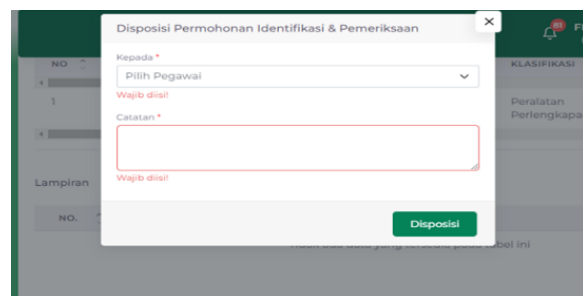
Data Input	Data Uji	
Catatan (free text)	<i>Equivalence Partitioning</i>	
	Partisi nilai invalid	Nilai 0
	Partisi nilai valid	Nilai lebih dari 1 (10)
	<i>Boundary Value Analysis</i>	
	Nilai minimum, invalid (n-1)	Nilai 0
Kepada	Batas minimum yang valid	Nilai 1
	<i>Equivalence Partitioning</i>	
	Partisi nilai invalid	Dropdown kosong
	Partisi nilai valid	Dropdown terisi dengan salah satu opsi

Pada tabel diatas merupakan penggunaan metode *Equivalence Partitioning* dan *Boundary Value Analysis* pada setiap data input, yang akan digunakan dalam skenario pengujian.

Tabel 10. Hasil pengujian disposisi permohonan identifikasi

Test ID	Skenario	Expected Result	Result
SP011	Mengajukan permohonan dengan mengisi form yang ada	Dapat memberikan disposisi kepada inspektur atau pegawai lainnya	Sesuai
SP012	Mengisi salah satu kolom untuk mengajukan permohonan	Gagal memberikan disposisi kepada inspektur dan muncul peringatan kolom yang harus diisi	Sesuai
SP013	Mengosongkan semua kolom untuk mengajukan permohonan	Gagal memberikan disposisi kepada inspektur dan muncul peringatan kolom yang harus diisi	Sesuai

Tabel di atas menunjukkan hasil tes pada disposisi permohonan identifikasi. Dari tiga skenario tes yang dirancang, tiga di antaranya sesuai dengan hasilnya.



Gambar 5. Disposisi permohonan identifikasi

Pada gambar diatas adalah tampilan form disposisi permohonan identifikasi.

e. Penghapusbukuan Aset Tanah dan Bangunan

Tabel 11. Data uji settlement penghapusbukuan tanah dan bangunan

Data Input	Data Uji	
No Document Settlement	Equivalence Partitioning	
	Partisi nilai invalid	Nilai 0
		Nilai lebih dari 128 (200)
	Partisi nilai valid	Nilai 1-128 karakter (20)
	Boundary Value Analysis	
	Nilai minimum, invalid (n-1)	Nilai 0

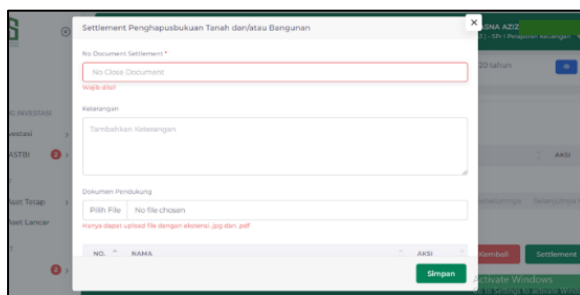
Data Input	Data Uji	
	Batas minimum yang valid	Nilai 1 dan 2
	Batas maksimum yang valid	Nilai 127 dan 128
	Nilai di atas batas maksimum, invalid (n+1)	Nilai 129

Pada tabel diatas merupakan penggunaan metode *Equivalence Partitioning* dan *Boundary Value Analysis* pada setiap data input, yang akan digunakan dalam skenario pengujian.

Tabel 12. Hasil pengujian settlement penghapus bukuan tanah dan bangunan

Test ID	Skenario	Expected Result	Result
SP014	Proses penghapusbukuan tanah dan bangunan dilakukan dengan melengkapi formulir yang ada dan bernilai valid	Mampu mensettlement penghapusbukuan tanah dan bangunan	Sesuai
SP015	Mengkosongkan form yang ada	Gagal menyelesaikan penghapusbukuan tanah dan bangunan, dan muncul peringatan tentang formulir yang harus diisi	Sesuai
SP016	Menghapusbukuan tanah dan bangunan dengan menambahkan nilai yang lebih besar dari nilai yang valid	Settlement penghapusbukuan tanah dan bangunan gagal, dan muncul notifikasi error	Sesuai
SP017	Melihat daftar settlement tanah dan bangunan	1. mampu melihat daftar yang harus disettlement 2. mampu melihat daftar penghapusbukuan Tanah dan Bangunan yang disettlement	Sesuai
SP018	Melihat detail settlement tanah dan bangunan	Mampu melihat detail hasil settlement tanah dan bangunan	Sesuai
SP019	Melakukan Logout	Sukses melakukan logout	Sesuai

Tabel di atas menunjukkan hasil pengujian pada halaman settlement penghapusbukuan tanah dan bangunan, serta daftar dan detail settlement penghapusbukuan tanah dan bangunan. 6 dari 6 skenario pengujian dirancang sesuai dengan hasilnya.



Gambar 6. Settlement penghapusbukuan tanah dan bangunan

Pada gambar diatas adalah tampilan form settlement penghapusbukuan tanah dan bangunan.

4.3. Nilai efektifitas

Pada tahap ini, nilai efektifitas aplikasi dihitung dengan mengetahui jumlah tabel yang diuji, jumlah skenario pengujian, dan hasil skenario pengujian. Hasil dari ketiga faktor ini dapat dihitung dengan

menggunakan rumus. Hasil pengujian dapat disimpulkan sebagai berikut:

Tabel 13. Kesimpulan data uji

No	Modul pengujian	Hasil tidak sesuai	Hasil sesuai	Total skenario
1	Pengajuan Investasi	1	36	37 skenario
2	Penciptaan BASTBI	0	14	14 skenario
3	Pengelolaan Aset Tetap	0	15	15 skenario
4	Penghapusbukuan Aset BTT	0	129	129 skenario
5	Penghapusbukuan Aset Tanah dan Bangunan	0	32	32 skenario
Total		1	226	227

Tabel di atas berisi informasi tentang data uji, dan perhitungan nilai efektifitas penggunaan manajemen aset PT. Petrokimia Gresik adalah sebagai berikut:

- Menghitung efektifitas tiap tabel
Perhitungan nilai efektifitas menggunakan persamaan (1) sebagai berikut :

Tabel 14. Perhitungan nilai efektifitas

Keterangan	Modul Pengujian	Hasil Perhitungan
Hasil sesuai	Pengajuan Investasi	$\left(\frac{36}{37}\right)$ $\times 100\%$ $= 97\%$
	Penciptaan BASTBI	$\left(\frac{14}{14}\right)$ $\times 100\%$ $= 100\%$
	Pengelolaan Aset Tetap	$\left(\frac{15}{15}\right)$ $\times 100\%$ $= 100\%$
	Penghapusbukuan Aset BTT	$\left(\frac{129}{129}\right)$ $\times 100\%$ $= 100\%$
	Penghapusbukuan Aset Tanah dan Bangunan	$\left(\frac{32}{32}\right)$ $\times 100\%$ $= 100\%$
Hasil tidak sesuai	Pengajuan Investasi	$\left(\frac{1}{37}\right)$ $\times 100\%$ $= 3\%$

Tabel di atas menunjukkan hasil perhitungan skenario yang sesuai dan tidak sesuai untuk setiap modul. Satu hasil yang tidak sesuai terjadi untuk modul pengajuan investasi, sementara hasil untuk modul lainnya sesuai sebesar 100%.

- b. Menghitung efektifitas keseluruhan
Perhitungan nilai efektifitas keseluruhan menggunakan persamaan (2) sebagai berikut:

Nilai efektifitas :

$$\left(\frac{100\% + 100\% + 100\% + 100\% + 97\%}{5} \right) \times 100\% = 99\%$$

Hasil perhitungan di atas menunjukkan bahwa satu skenario pengujian yang tidak sesuai terjadi pada modul pengajuan investasi, sehingga aplikasi manajemen aset PT Petrokimia Gresik memiliki tingkat capaian yang sangat efektif dengan rasio efektifitas di atas 80%. Dengan demikian, nilai efektifitas total aplikasi adalah 99% dari total efektifitas setiap modul dan dibagi oleh banyak modul. Dengan demikian, aplikasi sudah dapat berjalan sesuai mekanisme dan berfungsi dengan baik.

4.4. Daftar temuan kesalahan aplikasi

Setelah menguji aplikasi manajemen aset PT Petrokimia Gresik, berikut adalah kelemahan yang ditemukan penguji. Kelemahan tersebut disertai dengan saran untuk perbaikan.

Tabel 15. Hasil scenario pengujian yang tidak sesuai

[illegible]

Hasil pengujian yang tidak sesuai ditunjukkan pada table di atas. Ini terjadi saat memasukkan data yang salah pada halaman edit investasi.

Tabel 16. Daftar temuan kelemahan sistem

No	Kelemahan sistem	Rekomendasi perbaikan
1	Jika pengguna mengisi nilai yang lebih besar dari nilai valid pada formulir edit investasi melalui pada kolom lokasi dan keterangan, sistem tidak akan menampilkan peringatan error.	Menambahkan pemberitahuan kesalahan pada formulir jika pengguna menginputkan nilai lebih dari nilai validnya untuk memberi tahu pengguna tentang kesalahan yang terjadi.

Tabel di atas menunjukkan kelemahan sistem aplikasi manajemen aset. Hasil pengujian menunjukkan satu kelemahan dan rekomendasi untuk perbaikan.

4.5. Analisis prioritas perbaikan

Setelah melakukan pengujian pada aplikasi manajemen aset, ditemukan bahwa salah satu kelemahan sistem yang signifikan adalah tidak dapat menampilkan notifikasi error saat pengguna memasukkan nilai yang tidak valid pada form edit investasi di kolom lokasi dan keterangan. Karena pengguna mungkin tidak menyadari kesalahan yang terjadi, kelemahan ini dapat mengganggu integritas data dan keakuratan informasi sistem. Salah satu perbaikan yang disarankan adalah menambahkan peringatan kesalahan ke formulir edit investasi. Ini akan memungkinkan pengguna menemukan dan memperbaiki kesalahan dengan cepat, meningkatkan keamanan data, dan

meningkatkan pengalaman pengguna secara keseluruhan.

Diharapkan bahwa implementasi perbaikan ini akan meningkatkan efektivitas aplikasi secara keseluruhan dengan memberikan feedback yang tepat kepada pengguna dan meningkatkan kepercayaan pengguna terhadap sistem. Selain itu, perbaikan ini akan memastikan bahwa data dan informasi yang disimpan dalam sistem adalah akurat, sehingga pengguna dapat lebih mengandalkan aplikasi saat mengelola aset perusahaan.

5. KESIMPULAN DAN SARAN

Hasil pengujian yang dilakukan pada aplikasi manajemen aset menggunakan teknik *Equivalence Partitioning* dan *Boundary Value Analysis* menunjukkan bahwa dari 227 skenario pengujian, 226 sesuai dan 1 tidak sesuai. Ini menunjukkan bahwa aplikasi menghasilkan nilai efektivitas sebesar 99% dan dapat dianggap sangat efektif karena berjalan sesuai dengan mekanisme dan fungsionalitasnya. Kesalahan yang muncul tersebut harus diperbaiki segera. Berikut ini adalah saran untuk perbaikan dan pengembangan: Segera terapkan rekomendasi perbaikan untuk meningkatkan penggunaan aplikasi. Secara teratur setiap enam bulan, mengevaluasi kinerja dan fungsionalitas aplikasi untuk menemukan area perbaikan dan mengikuti perubahan kebutuhan pengguna untuk memastikan relevansi dan efisiensi sistem. Melakukan uji pengguna langsung untuk mendapatkan umpan balik penting tentang pengalaman pengguna sehari-hari. Melanjutkan penelitian untuk pengembangan lebih lanjut dengan menggunakan alat pengujian seperti IDE Selenium untuk hasil yang dapat dibandingkan dan saling melengkapi.

DAFTAR PUSTAKA

- [1] M. Ridwan and S. Ramadhani, "Rancangan Sistem Informasi Manajemen Aset di PT. Sentral Tukang Indonesia," *Jurnal CoreIT*, vol. 3, no. 2, 2017.
- [2] W. Wibisono and F. Baskoro, "PENGUJIAN PERANGKAT LUNAK DENGAN MENGGUNAKAN MODEL BEHAVIOUR UML," 2002.
- [3] M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, "PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)," 2015.
- [4] A. Sukmawati and G. Agiyani, "Cause Effect Graph dan Functional Testing dalam Pengujian Sistem Informasi Perjalanan Dinas," 2022.
- [5] D. I. Putri, "Teknik Equivalence Partitions untuk Pengujian Aplikasi Manajemen Kas dan Inventaris Berbasis Web," *INFORMATION MANAGEMENT FOR EDUCATORS AND PROFESSIONALS*, vol. 6, no. 2, pp. 193–202, 2022.
- [6] P. A. Desi, A. Santi, R. Afwani, and M. A. Albar, "PENGUJIAN BLACK BOX DENGAN METODE EQUIVALENCE PARTITIONING DAN BOUNDARY VALUE ANALYSIS (STUDI KASUS: SISTEM INFORMASI AKADEMIK UNIVERSITAS MATARAM)," 2022.
- [7] M. Sholeh, I. Gisfas, Cahiman, and M. A. Fauzi, "Black Box Testing on ukmbantul.com Page with Boundary Value Analysis and Equivalence Partitioning Methods," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Mar. 2021. doi: 10.1088/1742-6596/1823/1/012029.
- [8] H. Albert, S. Hutapea, Y. Priyadi, and E. Darwiyanto, "Analisis dan Pengujian dengan Menggunakan Metode Boundary Value Analysis dan Metode Equivalence Partitioning (Studi Kasus: Aplikasi Homelab)," Apr. 2022.
- [9] Vikasari, "Pengujian Sistem Informasi Magang Industri dengan Metode Blackbox Testing Boundary Value Analysis," 2018.
- [10] Y. Dwi Wijaya and M. Wardah Astuti, "PENGUJIAN BLACKBOX SISTEM INFORMASI PENILAIAN KINERJA KARYAWAN PT INKA (PERSERO) BERBASIS EQUIVALENCE PARTITIONS," *Jurnal Digital Teknologi Informasi*, vol. 4, p. 22, 2021.
- [11] V. Febrian, M. R. Ramadhan, M. Faisal, and A. Saifudin, "Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox," vol. 5, no. 1, pp. 2622–4615, 2020.
- [12] R. Parlita, T. Ardhan Nisaa', S. M. Ningrum, and B. A. Haque, "LITERATURE STUDY OF THE LACK AND EXCESS OF TESTING THE BLACK BOX," *TEKNOMATIKA*, vol. 10, no. 02, pp. 1–5, 2020.
- [13] A. Richardson, "Boundary Value Analysis Explained," eviltester.com.
- [14] A. Rifqi, Y. Arfani, P. Kasih, and D. P. Pamungkas, "Pengujian Aplikasi Presensi dengan Black box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," May 2021.
- [15] Autodika, "Cara Membuat Skenarion Testing Untuk Melakukan Pengujian Aplikasi," Autodika.com.
- [16] H. Dumoharis Lumbanraja, "PERANCANGAN SISTEM INFORMASI AKADEMIK ONLINE MENGGUNAKAN BLACK BOX TESTING PADA SEKOLAH TINGGI ILMU EKONOMI SURYA NUSANTARA," Oct. 2018.