

Analisis Kinerja Pengujian Black Box Terhadap *Website Nestify* Setelah Implementasi *Equivalence Partitioning* dan *Boundary Value Analysis*

Black Box Testing Performance Analysis of Nestify Website After Implementing Equivalence Partitioning and Boundary Value Analysis

Andika Raka Wicaksono*¹

³Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia

*e-mail: 21523146@students.uui.ac.id¹

Received: 12.04.2025	Revised: 06.05.2025	Accepted: 19.05.2025	Available online: 30.05.2025
-------------------------	------------------------	-------------------------	---------------------------------

Abstract: *Nestify is a web-based application for managing boarding house properties. To ensure functional conformance based on the use case diagram, Black Box testing with Equivalence Partitioning (EP) and Boundary Value Analysis (BVA) techniques was conducted to optimize test case selection and focus on boundary value testing. Careful test case design is the key to effective fault finding. Test results on the functionality of Boarding House Owner Account Login, Boarding House Owner Account Registration, and Manage Property prove the effectiveness of this combination of methods. Of the 33 total test cases executed, 29 test cases (87.88%) were successful, indicating a very positive performance and approaching the "Sangat Baik" category. However, several test cases (12.12%) were identified that did not pass, where the failures were spread over aspects of input validation and handling of certain conditions, including email length limit (Login), unique email validation (Registration), as well as photo file format validation and mandatory field handling (Manage Property).*

Keywords: *Black Box Testing, Equivalence Partitioning, Boundary Value Analysis, Software Testing, Nestify*

Abstrak: Aplikasi Nestify merupakan aplikasi berbasis *website* untuk pengelolaan properti indekos. Untuk memastikan kesesuaian fungsional berdasarkan *use case diagram*, dilakukan pengujian *Black Box* dengan teknik *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) guna optimalisasi pemilihan kasus uji dan fokus pada pengujian nilai batas. Desain kasus uji yang cermat menjadi kunci penemuan kesalahan secara efektif. Hasil pengujian pada fungsionalitas Login Akun Pemilik Kost, Registrasi Akun Pemilik Kost, dan Kelola Properti membuktikan efektivitas kombinasi metode ini. Dari 33 total kasus uji yang dieksekusi, 29 kasus uji (87.88%) dinyatakan berhasil, menunjukkan kinerja yang sangat positif dan mendekati kategori "Sangat Baik". Meskipun demikian, teridentifikasi beberapa kasus uji (12.12%) yang tidak lolos, di mana kegagalan tersebar pada aspek validasi *input* dan penanganan kondisi tertentu, meliputi batas panjang email (Login), validasi email unik (Registrasi), serta validasi format *file* foto dan penanganan *field* wajib (Kelola Properti).

Kata kunci: Pengujian Black Box, *Equivalence Partitioning*, *Boundary Value Analysis*, Pengujian Perangkat Lunak, Nestify

1. PENDAHULUAN

Pengujian perangkat lunak adalah tahap fundamental dalam siklus hidup pengembangan perangkat lunak (SDLC) yang krusial untuk mendeteksi *bug*, memastikan aplikasi berfungsi sesuai spesifikasi, andal, dan memenuhi kebutuhan pengguna akhir. Praktik pengujian yang efektif membangun kepercayaan pengguna dan mencegah peluncuran produk berkualitas rendah (Srivastava et al., 2024; Bajjouk et al., 2021). Pengabaian fase ini dapat menimbulkan masalah fungsional dan keandalan, di mana keandalan probabilitas sistem beroperasi tanpa kegagalan sangat dipengaruhi oleh komprehensivitas pengujian.

Black Box Testing adalah pendekatan umum yang mengevaluasi fungsionalitas perangkat lunak dari perspektif eksternal tanpa pengetahuan kode internal (Totade et al., 2023; Hu et al., 2017). Fokusnya adalah validasi perilaku sistem terhadap spesifikasi, efektif untuk mengungkap berbagai *bug* melalui simulasi interaksi pengguna. Namun, cakupannya bisa terbatas tanpa perancangan kasus uji yang cermat.

Untuk meningkatkan efektivitas *Black Box Testing*, teknik *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) sering digunakan. EP mengelompokkan data *input* ke dalam partisi ekuivalen, mengurangi redundansi kasus uji dengan menguji satu sampel per partisi (Santi et al., 2022), serta membedakan *input valid* dan *invalid*. BVA memfokuskan pengujian pada nilai-nilai batas antar

partisi, di mana kesalahan sering terjadi (Liew, 2024). Kombinasi EP dan BVA menghasilkan kasus uji *Black Box* yang lebih terarah dan efisien.

Pengembangan aplikasi *website* Nestify menjadi konteks penerapan prinsip pengujian ini. Proyek ini melanjutkan konsep aplikasi Nestify *mobile* yang dirintis melalui pendekatan *User Centered Design* (UCD) (Fauzi, 2024), dan kini berfokus pada implementasi *website* menggunakan metodologi *Research and Development* (R&D) dengan model *Waterfall*. Model *Waterfall* dipilih karena kesesuaiannya dengan persyaratan fungsional aplikasi manajemen indeks Nestify yang telah dipetakan secara komprehensif dalam studi UCD. Hasil analisis UCD tersebut, yang merinci konteks penggunaan dan kebutuhan pengguna, menjadi landasan penting untuk tahap Analisis Kebutuhan pada proyek *website* ini. Dengan dasar kebutuhan pengguna yang kuat, pengujian *Black Box Testing* pada *website* Nestify menjadi esensial untuk memverifikasi kesesuaian fungsionalitas dengan persyaratan, memastikan solusi efektif, pengalaman pengguna optimal, serta kompatibilitas lintas *browser* dan *platform*.

Penelitian ini berfokus pada analisis kinerja pengujian *Black Box* terhadap aplikasi *website* Nestify, setelah perancangan dan implementasi kasus uji menggunakan kombinasi teknik EP dan BVA. Tujuan utamanya adalah mengevaluasi efektivitas dan efisiensi gabungan kedua teknik tersebut dalam mengidentifikasi potensi kegagalan fungsional pada *website* Nestify dan memastikan kesesuaian fungsionalitas dengan kebutuhan pengguna yang telah dirumuskan. Pengujian ini ditekankan pada validasi *output* sistem berdasarkan variasi *input* representatif untuk memberikan keyakinan terhadap kualitas dan kelayakan fungsional aplikasi *website* Nestify.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menerapkan pengujian *Black Box* pada aplikasi *website* Nestify menggunakan teknik *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) untuk menganalisis kinerja pengujian. Tahapan penelitian yang dilakukan adalah sebagai berikut:

1. Analisis Kebutuhan Fungsional:

Meninjau kebutuhan fungsional dan alur pengguna (*user flow*) *website* Nestify yang berasal dari studi UCD sebelumnya (Fauzi, 2024). Tahap ini bertujuan memahami fungsionalitas yang akan diuji sebagai dasar perancangan skenario pengujian.

2. Perancangan Kasus Uji (*Test Case Design*):

Tahap ini meliputi pembuatan skenario pengujian dari kebutuhan fungsional, pembuatan *test case matrix*, serta perancangan nilai *test case* menggunakan *Boundary Value Analysis* (BVA) dan *Equivalence Partitioning* (EP). Batas pada pengelompokan data *test case* pada BVA adalah sebagai berikut:

- BLB (*Below the Lower Bound*): Nilai di bawah batas minimum.
- LB (*Lower Bound*): Nilai batas minimum.
- ALB (*Above the Lower Bound*): Nilai di atas batas minimum.
- BUB (*Below the Upper Bound*): Nilai di bawah batas maksimum.
- UB (*Upper Bound*): Nilai batas maksimum.
- AUB (*Above the Upper Bound*): Nilai di atas batas maksimum.

Sedangkan untuk *Equivalence Partitioning* (EP), *test case* dibagi menjadi kelas data *input valid* dan kelas data *input invalid* sesuai kriteria fungsionalitas.

3. Pelaksanaan Pengujian dan Analisis Kinerja:

Tahap selanjutnya adalah eksekusi seluruh kasus uji yang telah dirancang pada aplikasi *website* Nestify. Hasil setiap kasus uji didokumentasikan sebagai 'Berhasil' (*Pass*) jika *output* sesuai ekspektasi, atau 'Gagal' (*Fail*) jika terjadi penyimpangan. Kinerja pengujian dianalisis dengan menghitung persentase keberhasilan total menggunakan formula:

$$N = \frac{SP}{SM} \times 100$$

(N = Persentase keberhasilan, SP = Jumlah kasus uji berhasil, SM = Total kasus uji dieksekusi) (Lintang Ardias et al., 2024).

Kriteria penilaiannya adalah: Sangat Baik (89-100%), Baik (70-80%), Kurang (31-69%), dan Sangat Kurang (0-30%). Persentase ini menjadi indikator utama untuk mengevaluasi kesesuaian fungsionalitas *website* Nestify terhadap kebutuhan, setelah diuji dengan metode *Black Box* berbasis teknik *Equivalence Partitioning* dan *Boundary Value Analysis*.

2.2 Penelitian Terdahulu

Penelitian terdahulu dalam penelitian ini merupakan referensi penulisan dalam melakukan penelitian Pengujian Black Box Menggunakan Metode *Equivalence Partitioning* dan *Boundary Value Analysis*, diantaranya adalah:

1. Wardah Gracillaria Suharyono et al., (2024), menunjukkan bahwa penelitian dengan menerapkan secara eksplisit metode *Boundary Value Analysis* (BVA) dan *Equivalence Partitioning* (EP) untuk pengujian *Black Box* pada aplikasi SIADITA (Sistem Informasi Akademik dan Tugas Akhir), menunjukkan penggunaan gabungan kedua teknik pada sistem akademik spesifik.
2. Menurut Jimi Asmara (2019, dikutip dalam Selviana et al., 2023), penelitian pengujian aplikasi Kantor Digital di Politeknik Negeri Lampung menggunakan metode *Black Box* dengan teknik *Boundary Value Analysis* (BVA). Penelitian ini bertujuan menemukan tingkat kesalahan dan melaporkan pencapaian persentase keberhasilan 91.67%, mengindikasikan efektivitas BVA dalam menangani data normal maupun abnormal pada perangkat lunak administratif.
3. Menurut (Putu et al., 2020), Rosma (2019) melakukan penelitian pengujian sistem informasi inventaris di sebuah perusahaan farmasi menggunakan metode *Boundary Value Analysis* (BVA). Pengujian tersebut berhasil mencakup semua fitur atau menu dalam sistem, yang menunjukkan kegunaan BVA untuk aplikasi bisnis pengelola kuantitas.
4. Reid, C. S. (1997), mereferensikan studi eksperimental penting dari Ntafos (1984) yang membandingkan efektivitas *Equivalence Partitioning* (EP), *Boundary Value Analysis* (BVA), dan pengujian acak pada sistem avionik. Ditemukan dalam studi tersebut bahwa BVA secara signifikan lebih efektif daripada EP dalam mendeteksi kesalahan (faults) pada sistem yang diuji.
5. Aryandana et al. (2020), membandingkan EP dan BVA pada modul "tambah obat" dan menyimpulkan bahwa EP lebih unggul berdasarkan metrik yang mereka gunakan (persentase kelulusan kasus uji lebih tinggi, kebocoran cacat lebih rendah), meskipun EP menghasilkan lebih banyak kasus uji dalam skenario tersebut.

2.3 Pengujian Aplikasi

Pengujian aplikasi adalah proses krusial dalam siklus pengembangan perangkat lunak yang bertujuan mengimplementasikan program untuk menemukan kesalahan atau cacat (*faults*) dan memastikan kualitas serta kesesuaian perangkat lunak. Terdapat fokus signifikan pada pengujian aplikasi seluler karena tantangan unik yang dihadapkannya (Zein et al., 2016).

2.4 Blackbox Testing

Black Box Testing adalah pendekatan pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dan perilaku eksternal aplikasi, tanpa memperhatikan struktur internal kode. Pendekatan ini sering dioperasionalkan melalui teknik spesifik seperti *Boundary Value Analysis* (BVA) dan *Equivalence Partitioning* (EP) untuk merancang kasus uji (Mustaqbal et al., 2015; Jaya, 2018).

2.5 Boundary Value Analysis (BVA)

Boundary Value Analysis (BVA) adalah teknik pengujian kotak hitam spesifik yang berfokus pada pengujian nilai-nilai yang berada di tepi atau batas dari partisi domain *input/output*, karena pengalaman menunjukkan kesalahan sering terjadi di sekitar batas-batas ini. Teknik ini secara sistematis memeriksa batas dengan memilih nilai tepat pada, di bawah, dan di atas batas tersebut (Jaya, 2018; Dobslaw et al., 2023).

2.6 Equivalence Partitioning (EP)

Equivalence Partitioning adalah teknik fundamental dalam *Black Box Testing* yang membagi domain *input* menjadi beberapa kelas kesetaraan, dengan asumsi semua nilai dalam satu kelas diproses serupa, untuk mengurangi jumlah total kasus uji. Teknik ini hampir selalu direkomendasikan untuk

digunakan bersama dengan *Boundary Value Analysis* (BVA) untuk efektivitas (Bhat & Quadri, 2015; Mechtaev, 2017).

2.7 Use Case Diagram

Use Case Diagram adalah jenis diagram perilaku dalam UML yang digunakan dalam rekayasa kebutuhan untuk memodelkan interaksi antara pengguna eksternal (Aktor) dan sistem (*Use Case*), guna menangkap persyaratan fungsional dan lingkup sistem. Penelitian terkait berfokus pada area seperti penjaminan kualitas melalui pemeriksaan konsistensi antar diagram dan penilaian semantik otomatis untuk mengukur kemiripan diagram (Fauzan et al., 2021; Yuwantoko et al., 2017).

3. HASIL PENELITIAN DAN PEMBAHASAN

Bagian ini menyajikan hasil dari pelaksanaan pengujian *Black Box* pada aplikasi *website* Nestify menggunakan kombinasi teknik *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA), serta pembahasan awal terhadap temuan tersebut. Pengujian dilakukan berdasarkan kasus uji yang dirancang pada tahap sebelumnya. Penelitian ini memiliki batasan dalam hal jenis pengujian yang dilakukan. Fokus utama adalah pada pengujian fungsional untuk memastikan kesesuaian fitur dengan kebutuhan pengguna. Aspek non-fungsional seperti pengujian kinerja, pengujian keamanan, dan pengujian usability belum menjadi cakupan dalam studi ini.

3.1 Desain Kasus Uji (*Test Case*)

Perancangan kasus uji yang efektif dan efisien meliputi identifikasi fungsionalitas utama untuk Pemilik Kost, penyusunan skenario dan matriks uji, serta perancangan nilai uji dengan teknik *Equivalence Partitioning* dan *Boundary Value Analysis*.

1. Identifikasi User

Pengguna utama (aktor) dalam pengujian *website* Nestify adalah Pemilik Kost, yaitu individu atau entitas yang mengelola properti indekos melalui *platform* Nestify. Tugas utamanya mencakup manajemen akun, properti, informasi kamar (seperti harga, fasilitas, ketersediaan), dan pemantauan aktivitas properti, termasuk melihat daftar penyewa yang akan masuk.

2. Identifikasi Kasus Uji (*Use Case*)

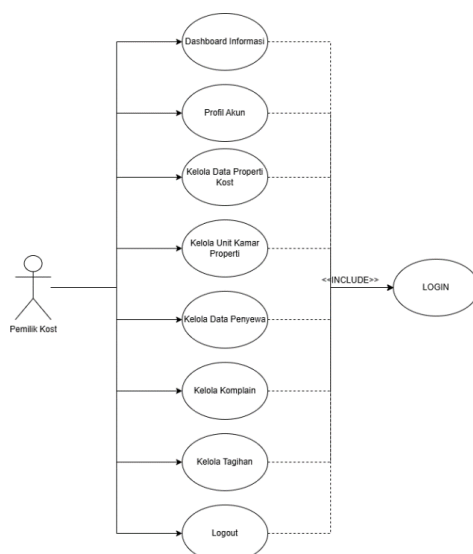
Use case memodelkan serangkaian tindakan yang dilakukan pengguna saat menggunakan aplikasi.

Tabel 1. Daftar *Use Case*

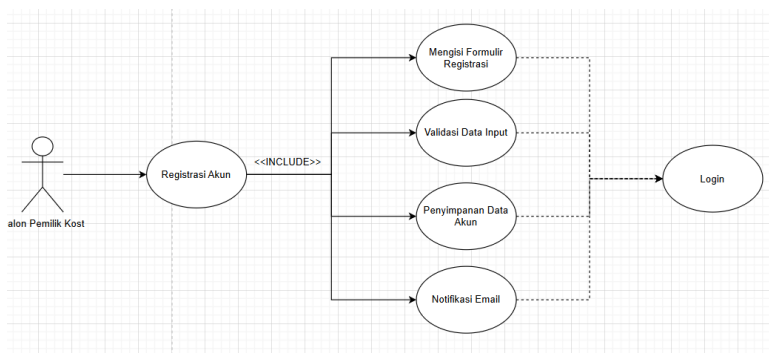
No.	Nama <i>Use Case</i>	Deskripsi Singkat
1	Registrasi Akun Pemilik	Pemilik Kost yang belum mempunyai akun dapat membuat akun dahulu menggunakan email dan membuat kata sandi yang baru.
2	Login Akun Pemilik Kost	Pemilik Kost masuk ke dalam sistem menggunakan email dan kata sandi yang telah terdaftar.
3	Dashboard Informasi	Pemilik Kost yang berhasil masuk ke dalam sistem akan di arahkan langsung ke <i>dashboard</i> informasi dan properti
4	Mengelola Properti Indekos	Pemilik Kost dapat menambah, melihat daftar, mengubah detail, dan menghapus data properti indekos miliknya.
5	Mengelola Kamar pada Properti	Untuk setiap properti, Pemilik Kost dapat menambah, melihat daftar, mengubah detail, dan menghapus data kamar.
6	Mengelola Data Penyewa	Pemilik Kost dapat melihat siapa saja penyewa yang aktif, sedang menunggu pengajuan penerimaan kost, atau yang sudah tidak aktif.
7	Mengelola Tagihan	Pemilik Kost dapat melihat jumlah pendapatan dari grafik dan dapat mengecek daftar penyewa yang belum lunas atau sudah lunas membayar kost.
8	Kelola Komplain	Pemilik Kost dapat menerima dan melakukan perbaikan melalui daftar komplain yang di ajukan oleh penyewa
9	Logout	Pemilik Kost keluar dari sistem untuk mengakhiri sesi.

3.1.1 Use Case Diagram

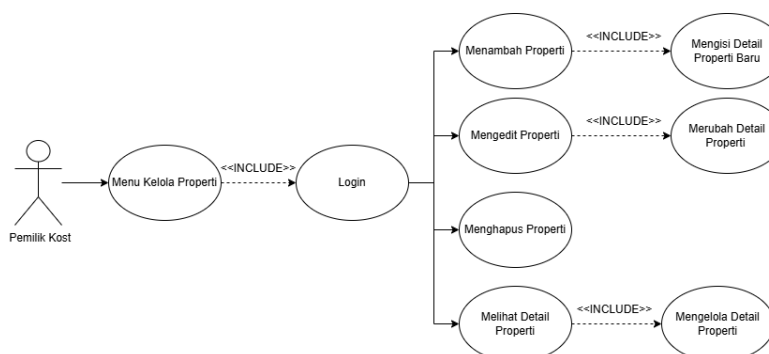
Gambar 1,2, dan 3 menyajikan diagram kasus penggunaan yang disusun berdasarkan detail yang terdapat pada Tabel 1.



Gambar 1. Use Case Diagram Login Pemilik Kost



Gambar 2. Use Case Diagram Registrasi Pemilik Kost



Gambar 3. Use Case Diagram Kelola Properti Pemilik Kost

3.1.2 Penyusunan Skenario Uji (Test Case)

Skenario uji mendeskripsikan alur interaksi pengguna (Pemilik Kost) dengan sistem secara lebih rinci, termasuk kondisi awal, langkah-langkah aksi, dan kondisi akhir yang diharapkan.

1. Skenario pengujian dikembangkan dari diagram kasus penggunaan.

Rincian skenario untuk fitur *Login* disajikan pada Tabel 2.

Tabel 2. Skenario Use Case Login

Nama Use Case	Login Akun Pemilik Kost
Aktor	Pemilik Kost Untuk memverifikasi bahwa Pemilik Kost yang terdaftar dapat berhasil masuk ke dalam sistem menggunakan kredensial (email dan kata sandi) yang <i>valid</i> , dan akses ditolak untuk kredensial yang tidak <i>valid</i> .
Tujuan	

Kondisi Awal	Pemilik Kost berada di halaman <i>Login</i> aplikasi Nestify. Sistem siap menerima <i>input</i> email dan kata sandi.
Skenario Utama (Normal Flow)	
1	Membuka halaman <i>Login</i> (jika belum terbuka).
2	Memasukkan alamat email yang <i>valid</i> dan telah terdaftar pada field "Email".
3	Memasukkan kata sandi yang <i>valid</i> dan sesuai dengan email yang dimasukkan pada field "Kata Sandi".
4	Menekan/klik tombol " <i>Login</i> ".
Kondisi Akhir (Normal)	Pemilik Kost berhasil <i>Login</i> dan berada di halaman dashboard atau halaman utama Pemilik Kost. Sesi <i>Login</i> Pemilik Kost aktif.
Skenario Alternatif (Exception Flows)	
1	Membuka halaman <i>Login</i> (jika belum terbuka).
2	Memasukkan alamat email yang salah atau tidak terdaftar.
3	Memasukkan kata sandi yang salah.
4	Menekan/klik tombol " <i>Login</i> ".
Kondisi Akhir (Alternatif)	Pemilik Kost gagal <i>Login</i> dan pesan kesalahan ditampilkan. Pemilik Kost tetap berada di halaman <i>Login</i> .

2. Skenario pengujian dikembangkan dari diagram kasus penggunaan.
Rincian skenario untuk fitur Registrasi disajikan pada Tabel 3.

Table 3. Skenario *Use Case* Registrasi

Nama Use Case	Registrasi Akun Pemilik Kost
Aktor	Calon Pemilik Kost Untuk memungkinkan Calon Pemilik Kost membuat akun baru di sistem Nestify dengan menyediakan informasi yang diperlukan dan valid, sehingga dapat menggunakan fitur-fitur yang tersedia untuk pemilik kost. Sistem juga akan menolak pembuatan akun jika informasi yang diberikan tidak valid atau tidak memenuhi kriteria.
Tujuan	Calon Pemilik Kost berada di halaman Registrasi aplikasi Nestify. Sistem siap menerima <i>input</i> data untuk pembuatan akun baru.
Kondisi Awal	
Skenario Utama (Normal Flow)	
1	Membuka halaman Registrasi (jika belum terbuka).
2	Memasukkan nama lengkap yang valid pada <i>field</i> "Nama Lengkap".
3	Memasukkan alamat email yang valid dan belum terdaftar di sistem pada <i>field</i> "Email".
4	Memasukkan kata sandi yang valid dan memenuhi kriteria keamanan pada <i>field</i> "Kata Sandi".
5	Memasukkan ulang kata sandi yang sama pada <i>field</i> "Konfirmasi Kata Sandi".
6	Memilih register sebagai <i>role</i> pemilik.
7	Menekan/klik tombol "Daftar" atau "Registrasi".
Kondisi Akhir (Normal)	Akun Calon Pemilik Kost berhasil dibuat. Sistem menampilkan pesan sukses registrasi dan mengirimkan email verifikasi. Data akun baru tersimpan di basis data sistem Nestify.
Skenario Alternatif (Exception Flows)	
1	Calon Pemilik Kost membuka halaman Registrasi (jika belum terbuka).
2	Calon Pemilik Kost mencoba melakukan registrasi dengan salah satu atau kombinasi kondisi ; Memasukkan alamat email yang sudah terdaftar di sistem.
3	Calon Pemilik Kost mencoba melakukan registrasi dengan salah satu atau kombinasi kondisi ; Tidak mengisi salah satu atau beberapa <i>field</i> wajib.
4	Calon Pemilik Kost mencoba melakukan registrasi dengan salah satu atau kombinasi kondisi ; Memasukkan data dengan format yang tidak valid pada satu atau beberapa <i>field</i> .

5	Calon Pemilik Kost mencoba melakukan registrasi dengan salah satu atau kombinasi kondisi ; <i>Input</i> pada <i>field</i> "Konfirmasi Kata Sandi" berbeda dengan <i>input</i> pada <i>field</i> "Kata Sandi".
6	Calon Pemilik Kost menekan/klik tombol "Daftar" atau "Registrasi". Pembuatan akun gagal. Sistem menampilkan pesan kesalahan yang relevan dengan kondisi kegagalan yang terjadi. Calon Pemilik Kost tetap berada di halaman Registrasi.
Kondisi Akhir (Alternatif)	

3. Skenario pengujian dikembangkan dari diagram kasus penggunaan.
Rincian skenario untuk fitur Kelola Properti disajikan pada Tabel 4.

Table 4. Skenario *Use Case* Menambah Kelola Properti

Nama Use Case	Menambah Properti Pemilik Kost
Aktor	Pemilik Kost Untuk memungkinkan Pemilik Kost menambahkan data properti indekos baru ke dalam sistem Nestify, termasuk detail nama, jenis, alamat, foto, dan informasi relevan lainnya.
Tujuan	Pemilik Kost telah login dan berada di halaman "Kelola Properti" atau dashboard utama. Pemilik Kost telah menginisiasi aksi untuk menambah properti baru.
Kondisi Awal	Sistem siap menerima <i>input</i> data properti baru.
Skenario Utama (Normal Flow)	
1	Pemilik Kost mengakses formulir penambahan properti baru.
2	Memasukkan nama properti yang valid pada <i>field</i> "Nama Properti".
3	Memilih jenis property pada <i>field</i> "Jenis Properti".
4	Memasukkan detail alamat properti yang lengkap pada <i>field</i> "Alamat Properti".
5	Mengunggah satu atau lebih foto properti yang valid dan sesuai format.
6	Memasukkan informasi tambahan properti.
7	Menekan/klik tombol "Simpan" atau "Tambah Properti". Data properti baru berhasil disimpan ke dalam sistem Nestify. Properti baru tersebut muncul dalam daftar properti milik Pemilik Kost. Sistem menampilkan pesan konfirmasi keberhasilan penambahan properti.
Kondisi Akhir (Normal)	
Skenario Alternatif (Exception Flows)	
1	Pemilik Kost mengakses formulir penambahan properti baru. Pemilik Kost mengisi sebagian data properti namun mengabaikan satu atau lebih <i>field</i> wajib atau memasukkan data dengan format yang salah.
2	
3	Pemilik Kost mengunggah foto properti.
4	Pemilik Kost menekan/klik tombol "Simpan" atau "Tambah Properti".
5	Sistem melakukan validasi terhadap semua <i>input</i> data yang diterima. Sistem mendeteksi bahwa <i>field</i> "Nama Properti" kosong, yang merupakan <i>field</i> wajib
6	Penambahan properti baru gagal. Sistem secara jelas mengindikasikan <i>field</i> mana yang memerlukan koreksi dan mengapa. Pemilik Kost tetap berada di formulir penambahan properti dengan semua data yang telah diisi sebelumnya tetap ada, sehingga Pemilik Kost hanya perlu memperbaiki data yang salah atau melengkapi yang kurang.
Kondisi Akhir (Alternatif)	

Table 5. Skenario *Use Case* Mengedit Kelola Properti

Nama Use Case	Mengedit Properti Pemilik Kost
Aktor	Pemilik Kost Untuk memungkinkan Pemilik Kost mengubah atau memperbarui informasi detail dari properti indekos miliknya yang sudah terdaftar di sistem Nestify.
Tujuan	Pemilik Kost telah login dan berada di halaman "Kelola Properti". Pemilik Kost telah memilih salah satu properti dari daftar dan menginisiasi aksi untuk mengedit properti tersebut. Formulir edit properti sudah terisi dengan data properti saat ini.
Kondisi Awal	

Skenario Utama (Normal Flow)

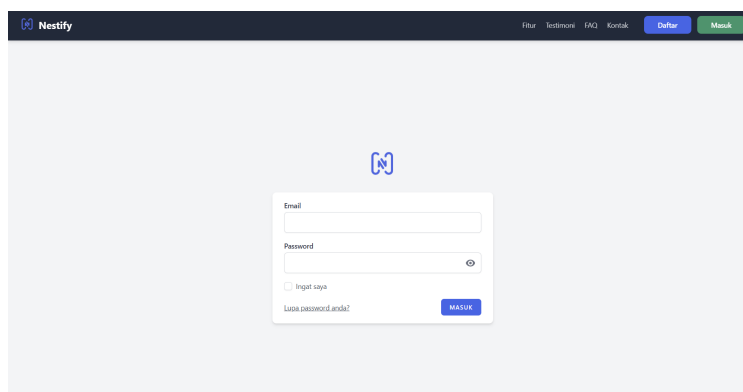
1	Pemilik Kost mengakses formulir pengeditan properti yang sudah terisi data properti yang dipilih.
2	Mengubah satu atau beberapa informasi pada <i>field</i> yang tersedia.
3	Jika ada perubahan foto, Pemilik Kost mengunggah foto baru atau menghapus foto lama.
4	Menekan/klik tombol "Simpan".
Perubahan data properti berhasil disimpan ke dalam sistem Nestify. Informasi properti yang diperbarui akan tampil di daftar properti dan halaman detail properti. Sistem menampilkan pesan konfirmasi keberhasilan pengeditan.	
Kondisi Akhir (Normal)	

Skenario Alternatif (Exception Flows)

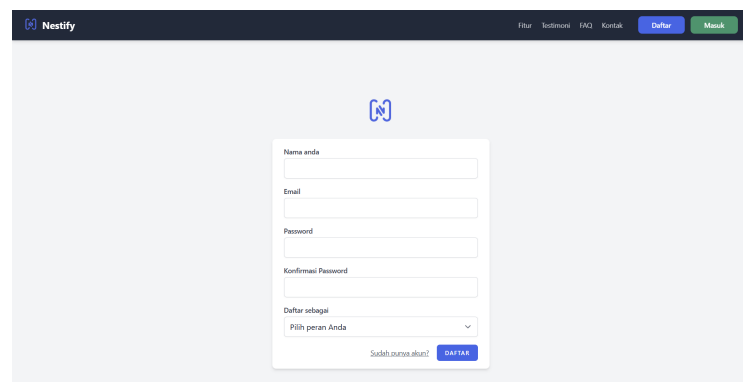
1	Pemilik Kost mengakses formulir pengeditan properti.
2	Pemilik Kost mengubah "Nama Properti" menjadi kosong.
3	Pemilik Kost menekan/klik tombol "Simpan".
4	Sistem melakukan validasi terhadap semua <i>input</i> data yang diubah.
5	Sistem mendeteksi bahwa "Nama Properti" kosong.
6	Sistem menampilkan pesan kesalahan <i>field</i> "Nama Properti": "Nama Properti tidak boleh kosong."
7	Data pada formulir kembali ke kondisi sebelum Pemilik Kost menekan "Simpan". Perubahan data properti gagal disimpan. Sistem secara jelas mengindikasikan <i>field</i> mana saja yang datanya tidak valid beserta alasannya. Pemilik Kost tetap berada di formulir pengeditan dan dapat langsung memperbaiki <i>input</i> yang salah.
Kondisi Akhir (Alternatif)	

3.1.3 Pembuatan Matriks Kasus Uji

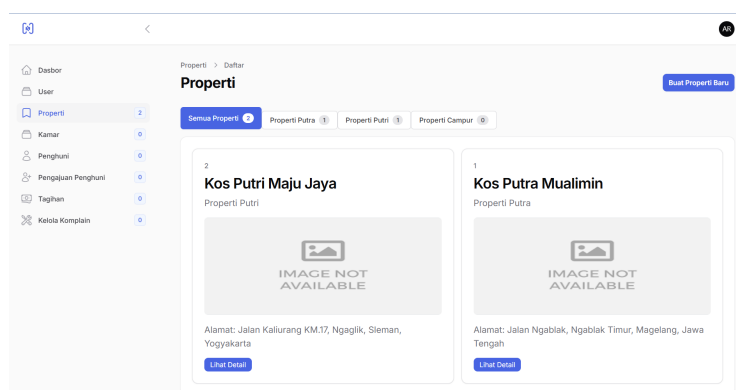
Kasus uji (*test case*) sendiri merupakan prosedur atau skenario yang dirancang untuk menguji fitur aplikasi, terdiri atas data uji dan hasil yang diharapkan. Ilustrasi antarmuka untuk halaman Login dapat dirujuk pada Gambar 4, halaman Registrasi dapat dilihat pada Gambar 5. Tampilan Halaman Registrasi Aplikasi Website Nestify, dan halaman Kelola Properti diilustrasikan pada Gambar 6. Tampilan Halaman Kelola Properti Aplikasi Website Nestify.



Gambar 4. Tampilan Halaman *Login* Aplikasi *Website* Nestify



Gambar 5. Tampilan Halaman Registrasi Aplikasi *Website* Nestify



Gambar 6. Tampilan Halaman Kelola Properti Aplikasi Website Nestify

Matriks kasus uji dikembangkan dari skenario pada Tabel 2. Rincian matriks kasus uji untuk skenario *Login* kemudian dipaparkan pada Tabel 6, rincian matriks kasus uji untuk skenario Registrasi kemudian dipaparkan pada Tabel 7, dan rincian matriks kasus uji untuk skenario Tambah Kelola Properti Baru dan Mengedit Properti kemudian dipaparkan pada Tabel 8 dan Tabel 9.

Table 6. Test Case Matrix Login

ID	Skenario	Input Email	Input Password	Hasil
1	Skenario Utama	Valid	Valid	Berhasil Login
2	Skenario Alternatif	Invalid	Valid	Menampilkan Warning fail
3	Skenario Alternatif	Valid	Invalid	Menampilkan Warning fail
4	Skenario Alternatif	Invalid	Invalid	Menampilkan Warning fail

Table 7. Test Case Matrix Registrasi

ID	Skenario	Input Nama Lengkap	Input Email	Input Kata Sandi	Input Kata Sandi Konfirmasi	Hasil yang Diharapkan
1	Skenario Utama	Valid	Valid (unik)	Valid (sesuai kriteria)	Valid (sama dengan Kata Sandi)	Berhasil Registrasi, Akun baru dibuat
2	Skenario Alternatif (Email sudah terdaftar)	Valid	Valid (sudah ada)	Valid (sesuai kriteria)	Valid (sama dengan Kata Sandi)	Gagal Registrasi, Menampilkan pesan "Email sudah terdaftar"
3	Skenario Alternatif (Format Email salah)	Valid	Invalid (format salah)	Valid (sesuai kriteria)	Valid (sama dengan Kata Sandi)	Gagal Registrasi, Menampilkan pesan "Format email tidak valid"
4	Skenario Alternatif (Kata Sandi lemah)	Valid	Valid (unik)	Invalid (tidak sesuai kriteria)	Invalid (tidak sesuai kriteria)	Gagal Registrasi, Menampilkan pesan "Kata sandi lemah/tidak sesuai"
5	Skenario Alternatif (Konfirmasi Kata Sandi salah)	Valid	Valid (unik)	Valid (sesuai kriteria)	Invalid (tidak sama)	Gagal Registrasi, Menampilkan pesan "Konfirmasi kata sandi tidak cocok"
6	Skenario Alternatif (Field wajib kosong)	Invalid (kosong)	Valid (unik)	Valid (sesuai kriteria)	Valid (sama dengan Kata Sandi)	Gagal Registrasi, Menampilkan pesan "Nama Lengkap tidak boleh kosong"
7	Skenario Alternatif (Beberapa field invalid)	Invalid (kosong)	Invalid (format salah)	Invalid (tidak sesuai kriteria)	Invalid (sama)	Gagal Registrasi, Menampilkan beberapa pesan kesalahan yang relevan

Table 8. Test Case Matrix Tambah Properti Baru

ID	Skenario	Input Nama Properti	Input Jenis Properti	Input Alamat Properti	Input Foto Properti	Input Informasi Tambahan	Aksi Pengguna	Hasil yang Diharapkan
1	Skenario Utama	Valid	Valid	Valid	Valid	Valid	Klik "Simpan Properti"	Properti baru berhasil disimpan,

2	Skenario Alternatif (Nama Properti Kosong)	Invalid (Kosong)	Valid	Valid	Valid	Valid	Klik "Simpan Properti"	tampil di daftar, pesan sukses. Gagal tambah, pesan "Nama Properti tidak boleh kosong", tetap di form. Gagal tambah, pesan "Alamat Properti tidak boleh kosong", tetap di form. Gagal tambah atau foto tidak terunggah, pesan "Format foto tidak didukung", tetap di form. Gagal tambah, menampilkan beberapa pesan kesalahan validasi, tetap di form.
3	Skenario Alternatif (Alamat Kosong)	Valid	Valid	Invalid (Kosong)	Valid	Valid	Klik "Simpan Properti"	
4	Skenario Alternatif (Format Foto Salah)	Valid	Valid	Valid	Invalid (Format Salah)	Valid	Klik "Simpan Properti"	
5	Skenario Alternatif (Field Wajib Kosong)	Invalid (Kosong)	Valid (Default)	Invalid (Kosong)	Tidak Diunggah	Opsional (Kosong)	Klik "Simpan Properti"	

Table 9. Test Case Matrix Edit Properti

ID	Skenario	Pilihan Properti	Input Nama Properti (Baru)	Input Jenis Properti (Baru)	Input Alamat (Baru)	Input Foto (Baru)	Aksi Pengguna	Hasil yang Diharapkan
1	Skenario Utama (Edit semua data valid)	Properti Valid	Valid	Valid	Valid	Valid	Klik "Simpan"	Perubahan properti berhasil disimpan, data terupdate di daftar, pesan sukses.
2	Skenario Alternatif (Nama Properti dikosongkan)	Properti Valid	Invalid (Kosong)	(Data Lama)	(Data Lama)	(Data Lama)	Klik "Simpan"	Gagal edit, pesan "Nama Properti tidak boleh kosong", tetap di form edit. Gagal edit foto atau semua, pesan "Format foto tidak didukung", tetap di form edit.
3	Skenario Alternatif (Format Foto Baru Salah)	Properti Valid	(Data Lama)	(Data Lama)	(Data Lama)	Invalid (Format Salah)	Klik "Simpan"	
4	Skenario Utama (Edit tanpa mengubah apapun)	Properti Valid	(Data Lama)	(Data Lama)	(Data Lama)	(Data Lama)	Klik "Simpan"	Tidak ada perubahan, kembali ke daftar atau pesan "Tidak ada data yang diubah".
5	Skenario Alternatif (Membatalkan Edit)	Properti Valid	(Ada perubahan di form)	(Ada perubahan di form)	(Ada perubahan di form)	(Ada perubahan di form)	Klik "Batal" (jika ada)	Perubahan tidak disimpan, kembali ke daftar properti atau detail properti (data lama).

3.1.4 Desain Nilai Uji dengan Equivalence Partitioning (EP) dan Boundary Value Analysis (BVA)

Penentuan nilai *input* yang dianggap *valid* oleh sistem akan mempermudah dalam mengidentifikasi dan merancang data uji untuk kondisi yang tidak *valid* (*invalid*). *Validasi check* untuk data *input* pada fungsionalitas *Login* Pemilik Kost di website Nestify dijelaskan pada Tabel 10, Tabel 11 untuk Registrasi, dan Tabel 12 untuk Kelola Properti.

Table 10. Cek Validasi Login

ID Validasi	Data Input	Deskripsi Kriteria Validasi
VAL_LOGIN_PK_001	Email	Email tidak boleh kosong.
VAL_LOGIN_PK_002	Email	Email harus memiliki format yang <i>valid</i>
VAL_LOGIN PK_003	Email	Panjang email memiliki batas minimal 6 dan maksimal 100 karakter.

VAL_LOGIN_PK_004	Email	Email yang dimasukkan harus sudah terdaftar di dalam basis data sistem Nestify.
VAL_LOGIN_PK_005	Kata Sandi	Kata Sandi tidak boleh kosong.
VAL_LOGIN_PK_006	Kata Sandi	Kata Sandi memiliki batas minimal 8 dan maksimal 32 karakter.
VAL_LOGIN_PK_007	Kata Sandi	Kata Sandi yang dimasukkan harus sesuai dengan kata sandi yang terasosiasi dengan email di basis data.

Table 11. Cek Validasi Registrasi

ID Validasi	Data Input	Deskripsi Kriteria Validasi
VAL_REG_PK_001	Nama Lengkap	Nama Lengkap tidak boleh kosong.
VAL_REG_PK_002	Nama Lengkap	Panjang Nama Lengkap memiliki batas.
VAL_REG_PK_003	Nama Lengkap	Nama Lengkap hanya boleh mengandung huruf dan spasi.
VAL_REG_PK_004	Email	Email tidak boleh kosong.
VAL_REG_PK_005	Email	Email harus memiliki format yang valid
VAL_REG_PK_006	Email	Panjang email memiliki batas
VAL_REG_PK_007	Email	Email yang dimasukkan harus unik dan belum pernah dibuat di database.
VAL_REG_PK_008	Kata Sandi	Kata Sandi tidak boleh kosong.
VAL_REG_PK_009	Kata Sandi	Kata Sandi memiliki batas minimal karakter
VAL_REG_PK_010	Kata Sandi	Kata Sandi memiliki batas maksimal karakter
VAL_REG_PK_011	Kata Sandi	Kata Sandi harus memenuhi kriteria kompleksita
VAL_REG_PK_012	Konfirmasi Kata Sandi	Konfirmasi Kata Sandi tidak boleh kosong.
VAL_REG_PK_013	Konfirmasi Kata Sandi	Isi Konfirmasi Kata Sandi harus sama dengan isi Kata Sandi.
VAL_REG_PK_014	Pilihan Peran (Role)	Pemilihan peran (sebagai Pemilik Kost) harus dilakukan jika ingin masuk ke bagian Pemilik Kost.

Table 12. Cek Validasi Kelola Properti

ID Validasi	Data Input	Deskripsi Kriteria Validasi
VAL_PROPERTI_PK_001	Nama Properti	Nama Properti tidak boleh kosong.
VAL_PROPERTI_PK_002	Nama Properti	Panjang Nama Properti memiliki batas.
VAL_PROPERTI_PK_003	Jenis Properti	Jenis Properti harus dipilih dari opsi yang valid.
VAL_PROPERTI_PK_004	Alamat Properti	Alamat Properti tidak boleh kosong.
VAL_PROPERTI_PK_005	Alamat Properti	Detail Alamat harus valid jika ada pemisahan field.
VAL_PROPERTI_PK_006	Foto Properti	Minimal satu foto properti harus diunggah.
VAL_PROPERTI_PK_007	Foto Properti	Format file foto harus sesuai.
VAL_PROPERTI_PK_008	Foto Properti	Ukuran file foto tidak boleh melebihi batas maksimal yang ditentukan.
VAL_PROPERTI_PK_009	Jumlah Foto	Jumlah maksimal foto yang dapat diunggah per properti dibatasi 5.
VAL_PROPERTI_PK_010	Informasi Tambahan	Jika field Informasi Tambahan memiliki batasan panjang, maka harus divalidasi.
VAL_PROPERTI_PK_011	Harga Sewa	Harga Sewa harus berupa angka positif dan tidak boleh kosong.
VAL_PROPERTI_PK_012	Jumlah Kamar	Jumlah Kamar harus berupa angka bulat positif.

Tabel 13 menyajikan penerapan kelas-kelas *Boundary Value Analysis* untuk *field input* pada halaman Login Pemilik Kost di *website* Nestify, berdasarkan kriteria validasi yang telah ditetapkan pada Tabel 10. Selanjutnya, Tabel 14 mengilustrasikan penerapan BVA untuk fitur Registrasi Akun Pemilik Kost, yang didasarkan pada kriteria validasi dari Tabel 11, dan Tabel 15 merincikan penerapan BVA untuk fitur Kelola Properti dengan mengacu pada kriteria validasi yang terdapat pada Tabel 12.

Table 13. Kelas *Boundary Value Analysis* Login

No.	Data Input	Aturan Batas Terkait	Nilai Uji	Valid / Invalid	Keterangan
1	Email	Minimal 6 karakter	andik@gmail.com	Invalid	BLB (Below Lower Bound)
2	Email	Minimal 6 karakter	andika@gmail.com	Valid	LB (Lower Bound)
3	Email	Minimal 6 karakter	andikar@gmail.com	Valid	ALB (Above Lower Bound)

Table 14. Kelas *Boundary Value Analysis* Registrasi

Table 15. Kelas *Boundary Value Analysis* Kelola Properti

89

18	Harga Sewa	Maksimal Rp10.000.000	10000001	Invalid	AUB (Above Upper Bound)
19	Jumlah Foto	Minimal 1 foto (untuk Tambah Baru)	0	Invalid	BLB (Below Lower Bound)
20	Jumlah Foto	Minimal 1 foto	1	Valid	LB (Lower Bound)
21	Jumlah Foto	Minimal 1 foto	2	Valid	ALB (Above Lower Bound)
22	Jumlah Foto	Maksimal 10 foto	9	Valid	BUB (Below Upper Bound)
23	Jumlah Foto	Maksimal 10 foto	10	Valid	UB (Upper Bound)
24	Jumlah Foto	Maksimal 10 foto	11	Invalid	AUB (Above Upper Bound)

Tabel 16 mendemonstrasikan langkah krusial dalam pengujian untuk fitur Login, yaitu pembagian kelas-kelas ekuivalen untuk data *input* email dan kata sandi. Dengan pendekatan serupa, Tabel 17 menyajikan pembagian kelas ekuivalen untuk data *input* pada fitur Registrasi Akun Pemilik Kost, dan Tabel 18 merincikan kelas-kelas ekuivalen untuk berbagai data *input* pada fitur Kelola Properti. Melalui ketiga tabel ini, peneliti mengidentifikasi berbagai kemungkinan *input* untuk masing-masing fungsionalitas, baik yang diharapkan diterima oleh sistem (*valid*) maupun yang seharusnya ditolak (*invalid*). Contohnya meliputi *input* yang kosong, format yang tidak sesuai, atau nilai yang berada di luar batasan yang ditentukan, guna memastikan bahwa setiap skenario pengujian yang dirancang bersifat relevan dan tidak redundan.

Table 16. Kelas Partioning Equivalence Login

No.	Data Input	Nilai Uji	Valid / Invalid	Keterangan
1	Email	andikaraka25@gmail.com	Valid	Memenuhi semua kriteria: tidak kosong, format valid, panjang sesuai, terdaftar.
2	Email	(kosong)	Invalid	Email kosong.
3	Email	Andikaraka.gmail.com	Invalid	Format email tidak valid.
4	Email	andik@gmail.com	Invalid	Panjang email di luar batas (kurang dari 6 karakter).
5	Email	dikaraka123@gmail.com	Invalid	Email tidak terdaftar di basis data.
6	Kata Sandi	Coba1234	Valid	Memenuhi semua kriteria: tidak kosong, panjang sesuai, cocok dengan email.
7	Kata Sandi	(kosong)	Invalid	Kata Sandi kosong.
8	Kata Sandi	Coba123	Invalid	Panjang Kata Sandi di luar batas (kurang dari 8 karakter).
9	Kata Sandi	Cobaaja1234	Invalid	Kata Sandi tidak sesuai dengan email di basis data.

Table 17. Kelas Partioning Equivalence Registrasi

No.	Data Input	Nilai Uji	Valid / Invalid	Keterangan
1	Nama Lengkap	Andika Raka Wicaksono	Valid	Memenuhi kriteria: tidak kosong, panjang sesuai, format valid.
2	Nama Lengkap	(kosong)	Invalid	Nama Lengkap kosong.
3	Nama Lengkap	A	Invalid	Panjang Nama Lengkap di luar batas
4	Nama Lengkap	AndikaRaka123!@#	Invalid	Format Nama Lengkap tidak valid
5	Email	Andikaraka25@gmail.com	Valid	Memenuhi kriteria: tidak kosong, format valid, panjang sesuai, unik.
6	Email	(kosong)	Invalid	Email kosong.
7	Email	https://www.google.com/	Invalid	Format email tidak valid
8	Email	andik@gmail.com	Invalid	Format email tidak valid dan/atau panjang di luar batas.
9	Email	Andikaraka65@gmail.com	Invalid	Email sudah terdaftar di basis data.
10	Kata Sandi	Coba1234	Valid	Memenuhi kriteria: tidak kosong, panjang sesuai, kompleksitas terpenuhi.
11	Kata Sandi	(kosong)	Invalid	Kata Sandi kosong.
12	Kata Sandi	coba	Invalid	Panjang Kata Sandi di luar batas (kurang dari batas minimal, misal <8 karakter).
13	Kata Sandi	Coba1234	Invalid	Tidak memenuhi kriteria kompleksitas

14	Konfirmasi Kata Sandi	Coba@t123	Valid	Sama dengan Kata Sandi yang valid.
15	Konfirmasi Kata Sandi	(kosong)	Invalid	Konfirmasi Kata Sandi kosong (jika Kata Sandi diisi).
16	Konfirmasi Kata Sandi	coba1234678	Invalid	Tidak sama dengan Kata Sandi yang dimasukkan.
17	Pilihan Peran	Pemilik Kost	Valid	Peran dipilih sesuai opsi yang valid.
18	Pilihan Peran	(tidak dipilih)	Invalid	Peran tidak dipilih (jika merupakan field wajib).

Table 18. Kelas *Partitioning Equivalence* Kelola Properti

No.	Data Input	Nilai Uji (Contoh)	Valid / Invalid	Keterangan
1	Nama Properti	Kost Putra Keluarga Sejahtera	Valid	Memenuhi kriteria: tidak kosong, panjang sesuai.
2	Nama Properti	(kosong)	Invalid	Nama Properti kosong.
3	Nama Properti	Kst	Invalid	Panjang Nama Properti di luar batas.
4	Jenis Properti	Laki-laki	Valid	Dipilih dari opsi yang valid.
5	Jenis Properti	(tidak dipilih)	Invalid	Jenis Properti tidak dipilih atau opsi default tidak valid.
6	Alamat Properti	Jl. Merdeka No. 17, Yogyakarta	Valid	Memenuhi kriteria: tidak kosong, format alamat dianggap valid.
7	Alamat Properti	(kosong)	Invalid	Alamat Properti kosong.
8	Foto Properti	Kost_kireina_dapur.jpg (1MB)	Valid	Format dan ukuran file sesuai, jumlah foto dalam batas.
9	Foto Properti	dokumen.pdf	Invalid	Format file foto tidak didukung.
10	Foto Properti	Kost_campur.jpg (5MB)	Invalid	Ukuran file foto melebihi batas maksimal.
11	Foto Properti	(Tidak ada foto diunggah)	Invalid	Minimal satu foto tidak terpenuhi.
12	Jumlah Kamar	10	Valid	Angka bulat positif dalam rentang yang wajar.
13	Jumlah Kamar	0	Invalid	Jumlah kamar tidak positif.
14	Jumlah Kamar	ABC	Invalid	Bukan angka.
15	Harga Sewa	750000	Valid	Angka positif dalam rentang yang wajar.
16	Harga Sewa	(kosong)	Invalid	Harga sewa kosong.
17	Harga Sewa	-100000	Invalid	Harga sewa negatif.
18	Harga Sewa	Lima Ratus Ribu	Invalid	Bukan angka.

3.2 Implementasi dan Hasil Pengujian Kasus Uji

3.2.1 Lingkungan pengujian

Pengujian dilaksanakan pada laptop *ASUS Vivobook S14 M5406UA* yang menjalankan *Microsoft Windows 11 Home Single Language* (10.0.22631) dan menggunakan *Google Chrome* 136.0.7103.93 (64-bit) sebagai peramban.

3.2.2 Pelaksanaan Pengujian

Pengujian dilakukan dengan mengeksekusi setiap *test case* menggunakan data *input* yang telah ditentukan, kemudian hasil observasi *output* dan perilaku sistem dicatat untuk verifikasi fungsionalitas. Rincian *test case* beserta ekspektasi hasil untuk fungsionalitas Login Akun Pemilik Kost disajikan pada Tabel 19, dan rangkuman hasil pengujiannya terdapat pada Tabel 22. Untuk fungsionalitas Registrasi Akun Pemilik Kost, desain kasus uji dipaparkan pada Tabel 20, dengan hasil pengujiannya dirangkum dalam Tabel 23. Selanjutnya, untuk fungsionalitas Kelola Properti, rincian desain kasus uji terdapat pada Tabel 21, dan hasil pengujiannya disajikan pada Tabel 24.

Table 19. Desain Kasus Uji Fungsionalitas *Login*

ID Test Case	Email (Input)	Kata Sandi (Input)	Hasil yang Diharapkan	Keterangan/Dasar Desain
TC_LOGIN _PK_R01	andikaraka25@gmail.com	coba1234	Berhasil <i>Login</i> . Gagal <i>Login</i> , pesan "Email atau Kata Sandi salah."	Input <i>valid</i> .
TC_LOGIN PK_R02	andikaraka25@gmail.com	cobaaja1234		Email <i>valid</i> , Kata Sandi tidak cocok.

Table 20. Desain Kasus Uji Fungsionalitas Registrasi

92

TC_REG_PK_R08	Andika Raka	Andikaraka71@gmail.com	lemah	lemah	Pemilik Kost	pesan "Email sudah terdaftar." Gagal Registrasi, pesan validasi panjang atau kompleksitas Kata Sandi. Gagal Registrasi, pesan "Konfirmasi kata sandi tidak cocok." Gagal Registrasi, pesan validasi panjang Nama Lengkap. Gagal Registrasi, pesan validasi panjang Email. Gagal Registrasi, pesan validasi panjang Kata Sandi. Gagal Registrasi, pesan validasi "Pilihan Peran harus dipilih."	Kata Sandi tidak memenuhi kriteria panjang/kompleksitas Konfirmasi Kata Sandi tidak sama dengan Kata Sandi. Panjang Nama Lengkap di bawah batas Panjang Email di bawah batas Panjang Kata Sandi di atas batas Pilihan Peran tidak dipilih.
TC_REG_PK_R09	Andika Raka	Andikaraka72@gmail.com	PassValid123	Coba12345	Pemilik Kost		
TC_REG_PK_R10	A	Andikaraka73@gmail.com	PassValid123	Coba12345	Pemilik Kost		
TC_REG_PK_R11	Andika Raka	andik@	PassValid123	Coba12345	Pemilik Kost		
TC_REG_PK_R12	Andika Raka	Andikaraka74@gmail.com	Coba12345678101112131415161718191	Coba12345678101112131415161718191	Pemilik Kost		
TC_REG_PK_R13	Andika Raka	Andikaraka75@gmail.com	Coba12345	Coba12345	(tidak dipilih)		

Table 21. Desain Kasus Uji Fungsionalitas Kelola Properti

ID Test Case	Fitur Kelola Properti	Input Data Utama / Kondisi Awal	Aksi Pengguna	Hasil yang Diharapkan	Keterangan/Dasar Desain
TC_KP_PK_R01	Tambah Properti	Semua field diisi dengan data valid	Klik tombol "Simpan Properti".	Properti baru berhasil disimpan, pesan sukses, tampil di daftar. Gagal menambah properti, pesan validasi "Nama Properti tidak boleh kosong". Gagal menambah properti, pesan validasi "Alamat Properti tidak boleh kosong". Gagal menambah properti, pesan validasi "Format file foto tidak didukung". Berhasil menampilkan halaman detail "Kost Idaman" dengan semua informasi yang benar. Perubahan properti berhasil disimpan, data	Skenario utama tambah properti dengan semua input valid. Tambah properti, Nama Properti kosong. Tambah properti, Alamat Properti kosong. Tambah properti, format foto salah. Skenario utama lihat detail. Skenario utama edit properti dengan semua input valid.
TC_KP_PK_R02	Tambah Properti	Nama Properti: (kosong). Input lain valid.	Klik tombol "Simpan Properti".		
TC_KP_PK_R03	Tambah Properti	Alamat Properti: (kosong). Input lain valid.	Klik tombol "Simpan Properti".		
TC_KP_PK_R04	Tambah Properti	Foto dokumen.pdf Properti: (format salah). Input lain valid.	Klik tombol "Simpan Properti".		
TC_KP_PK_R05	Lihat Detail Properti	Pilih properti "Kost Idaman" (properti valid dan ada di daftar). Pilih properti "Kost Lama".	Klik nama properti / tombol "Detail".		
TC_KP_PK_R06	Edit Properti	Input baru: Nama Properti="Kost Update"	Pilih properti, ubah data, klik		

		Keren", Jenis="Perempuan", Alamat, Foto semua valid.	"Simpan Perubahan".	terupdate, pesan sukses.	
TC_KP_PK_R07	Edit Properti	Pilih properti "Kost Lama". Input baru: Nama Properti="(kosong)". Input lain tidak diubah.	Pilih properti, kosongkan Nama, klik "Simpan Perubahan". Pilih properti,	Gagal mengedit, pesan validasi "Nama Properti tidak boleh kosong".	Skenario alternatif edit, Nama Properti dikosongkan.
TC_KP_PK_R08	Edit Properti	Pilih properti "Kost Lama". Input baru: Harga Sewa="ABCDE". Input lain tidak diubah.	ubah Harga Sewa, klik "Simpan Perubahan". Klik tombol "Hapus", lalu	Gagal mengedit, pesan validasi "Format Harga Sewa tidak valid". Properti berhasil dihapus, pesan	Skenario alternatif edit, Harga Sewa format salah.
TC_KP_PK_R09	Hapus Properti	Pilih properti "Kost Akan Dihapus" (properti valid).	konfirmasi "Ya, Hapus". Klik tombol "Hapus", lalu	sukses, hilang dari daftar.	Skenario utama hapus properti.
TC_KP_PK_R10	Hapus Properti	Pilih properti "Kost Akan Dihapus" (properti valid).	konfirmasi "Batal".	Penghapusan dibatalkan, properti tetap ada.	Skenario alternatif hapus, pembatalan.

Table 22. Hasil Pengujian Test Case Login

ID Test Case	Hasil yang Diharapkan	Hasil yang Didapatkan (Aktual)	Pass / Fail	Kondisi/Catatan
TC_LOGIN_PK_R01	Berhasil Login.	Berhasil Login	Pass	Login dengan kredensial valid.
TC_LOGIN_PK_R02	Gagal Login, pesan "Email atau Kata Sandi salah."	Invalid password	Pass	Email valid, Kata Sandi salah.
TC_LOGIN_PK_R03	Gagal Login, pesan "Email atau Kata Sandi salah."	Invalid email	Pass	Email tidak terdaftar.
TC_LOGIN_PK_R04	Gagal Login, pesan validasi "Email tidak boleh kosong."	Muncul Warning	Pass	Email dikosongkan.
TC_LOGIN_PK_R05	Gagal Login, pesan validasi "Kata Sandi tidak boleh kosong."	Muncul Warning	Pass	Kata Sandi dikosongkan.
TC_LOGIN_PK_R06	Gagal Login, pesan validasi "Format email tidak valid."	Muncul Warning	Pass	Format email salah.
TC_LOGIN_PK_R07	Gagal Login, pesan validasi panjang email (kurang).	Muncul Warning	Pass	Panjang email < batas minimal.
TC_LOGIN_PK_R08	Gagal Login, pesan validasi panjang Kata Sandi (kurang).	Muncul Warning	Pass	Panjang Kata Sandi < batas minimal.
TC_LOGIN_PK_R09	Gagal Login, pesan validasi panjang email (lebih).	Tidak muncul Warning	Fail	Panjang email > batas maksimal.
TC_LOGIN PK R10	Gagal Login, pesan validasi panjang Kata Sandi (lebih).	Muncul Warning	Pass	Panjang Kata Sandi > batas maksimal.

Table 23. Hasil Pengujian Test Case Registrasi

ID Test Case	Hasil yang Diharapkan	Hasil yang Didapatkan (Aktual)	Pass / Fail	Kondisi/Catatan
TC_REG_PK_R01	Berhasil Registrasi, pesan sukses, email verifikasi terkirim.	Berhasil Registrasi, pesan sukses muncul, email verifikasi terkirim.	Pass	Semua input valid.
TC_REG_PK_R02	Gagal Registrasi, pesan validasi "Nama Lengkap tidak boleh kosong."	Gagal Registrasi, pesan validasi "Nama Lengkap tidak boleh kosong" muncul.	Pass	Nama Lengkap kosong.
TC_REG_PK_R03	Gagal Registrasi, pesan validasi "Email tidak boleh kosong."	Gagal Registrasi, pesan validasi "Email tidak boleh kosong" muncul.	Pass	Email kosong.
TC_REG_PK_R04	Gagal Registrasi, pesan validasi "Kata Sandi tidak boleh kosong."	Gagal Registrasi, pesan validasi "Kata Sandi tidak boleh kosong" muncul.	Pass	Kata Sandi kosong.
TC REG PK R05	Gagal Registrasi, pesan validasi "Konfirmasi Kata Sandi tidak boleh kosong."	Gagal Registrasi, pesan validasi "Konfirmasi Kata Sandi tidak boleh kosong" muncul.	Pass	Konfirmasi Kata Sandi kosong.

TC_REG_PK_R06	Gagal Registrasi, pesan validasi "Format email tidak valid."	Gagal Registrasi, pesan validasi "Format email tidak valid" muncul.	Pass	Format email salah.
TC_REG_PK_R07	Gagal Registrasi, pesan "Email sudah terdaftar."	Registrasi berhasil meskipun email sudah terdaftar. Tidak ada pesan error.	Fail	Email sudah terdaftar, sistem gagal melakukan validasi duplikasi email.
TC_REG_PK_R08	Gagal Registrasi, pesan validasi panjang atau kompleksitas Kata Sandi.	Gagal Registrasi, pesan validasi "Kata sandi minimal 8 karakter" muncul.	Pass	Kata Sandi tidak memenuhi kriteria panjang/kompleksitas
TC_REG_PK_R09	Gagal Registrasi, pesan "Konfirmasi kata sandi tidak cocok."	Gagal Registrasi, pesan "Konfirmasi kata sandi tidak cocok" muncul.	Pass	Konfirmasi Kata Sandi tidak sama dengan Kata Sandi.
TC_REG_PK_R10	Gagal Registrasi, pesan validasi panjang Nama Lengkap.	Gagal Registrasi, pesan validasi "Nama Lengkap minimal 3 karakter" muncul.	Pass	Panjang Nama Lengkap di bawah batas.
TC_REG_PK_R11	Gagal Registrasi, pesan validasi panjang Email.	Gagal Registrasi, pesan validasi "Email minimal 6 karakter" muncul.	Pass	Panjang Email di bawah batas.
TC_REG_PK_R12	Gagal Registrasi, pesan validasi panjang Kata Sandi.	Gagal Registrasi, pesan validasi "Kata Sandi maksimal 32 karakter" muncul.	Pass	Panjang Kata Sandi di atas batas.
TC_REG_PK_R13	Gagal Registrasi, pesan validasi "Pilihan Peran harus dipilih."	Gagal Registrasi, pesan validasi "Silakan pilih peran Anda" muncul.	Pass	Pilihan Peran tidak dipilih.

Table 24. Hasil Pengujian Test Case Kelola Properti

ID Test Case	Hasil yang Diharapkan	Hasil yang Didapatkan (Aktual)	Pass / Fail	Kondisi/Catatan
TC_KP_PK_R01	Properti baru berhasil disimpan, pesan sukses, tampil di daftar.	Properti baru berhasil disimpan, pesan sukses muncul, properti tampil di daftar.	Pass	Skenario utama tambah properti dengan semua input valid.
TC_KP_PK_R02	Gagal menambah properti, pesan validasi "Nama Properti tidak boleh kosong."	Gagal menambah properti, pesan validasi "Nama Properti tidak boleh kosong" muncul.	Pass	Tambah properti, Nama Properti kosong.
TC_KP_PK_R03	Gagal menambah properti, pesan validasi "Alamat Properti tidak boleh kosong."	Gagal menambah properti, pesan validasi "Alamat Properti tidak boleh kosong" muncul.	Pass	Tambah properti, Alamat Properti kosong
TC_KP_PK_R04	Gagal menambah properti, pesan validasi "Format file foto tidak didukung."	Sistem menerima file dokumen.pdf, properti tersimpan namun foto tidak tampil/rusak. Tidak ada pesan error spesifik saat unggah.	Fail	Tambah properti, format foto salah. Sistem gagal memvalidasi format foto dengan benar.
TC_KP_PK_R05	Berhasil menampilkan halaman detail "Kost Idaman" dengan semua informasi yang benar.	Berhasil menampilkan halaman detail "Kost Idaman" dengan semua informasi yang benar.	Pass	Skenario utama lihat detail.
TC_KP_PK_R06	Perubahan properti berhasil disimpan, data terupdate, pesan sukses.	Perubahan properti berhasil disimpan, data terupdate di daftar, pesan sukses muncul.	Pass	Skenario utama edit properti dengan semua input valid.
TC_KP_PK_R07	Gagal mengedit, pesan validasi "Nama Properti tidak boleh kosong."	Perubahan properti berhasil disimpan meskipun Nama Properti dikosongkan.	Fail	Skenario alternatif edit, Nama Properti dikosongkan. Sistem gagal validasi field wajib.
TC_KP_PK_R08	Gagal mengedit, pesan validasi "Format Harga Sewa tidak valid."	Gagal mengedit, pesan "Masukkan angka yang valid untuk Harga Sewa" muncul.	Pass	Skenario alternatif edit, Harga Sewa format salah
TC_KP_PK_R09	Properti berhasil dihapus, pesan sukses, hilang dari daftar.	Properti berhasil dihapus, pesan sukses muncul, properti hilang dari daftar.	Pass	Skenario utama hapus properti.
TC_KP_PK_R10	Penghapusan dibatalkan, properti tetap ada.	Penghapusan dibatalkan, properti tetap ada di daftar.	Pass	Skenario alternatif hapus, pembatalan.

3.2.3 Analisis Terhadap Hasil Pengujian

Kinerja pengujian fungsionalitas diukur menggunakan formula persentase keberhasilan $N=(SP/SM) \times 100\%$, di mana SP adalah jumlah kasus uji berhasil dan SM adalah total kasus uji yang dieksekusi. Berdasarkan pengujian pada keseluruhan fungsionalitas yang dievaluasi (Login Akun Pemilik Kost, Registrasi Akun Pemilik Kost, dan Kelola Properti), dari total 33 kasus uji yang dieksekusi ($SM=33$) yang mencakup 10 kasus uji untuk Login, 13 kasus uji untuk Registrasi, dan 10 kasus uji untuk Kelola Properti, tercatat 29 kasus uji berhasil ($SP=29$). Jumlah kasus uji yang berhasil terdiri dari 9 kasus uji Login, 12 kasus uji Registrasi, dan 8 kasus uji Kelola Properti. Hal ini menghasilkan persentase keberhasilan (N) secara keseluruhan sebesar $(29/33) \times 100\% = 87.88\%$.

Mengacu pada kriteria penilaian yang ditetapkan (Sangat Baik (89-100%), Baik (70-80%), Kurang (31-69%), dan Sangat Kurang (0-30%)), persentase keberhasilan 87.88% untuk keseluruhan fungsionalitas yang diuji ini menunjukkan kinerja yang sangat positif, melampaui batas atas kategori 'Baik' dan sangat mendekati standar untuk kategori 'Sangat Baik'.

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Kombinasi metode *Black Box Testing* yang memanfaatkan teknik *Equivalence Partitioning* (EP) dan *Boundary Value Analysis* (BVA) terbukti efektif dalam mengevaluasi fungsionalitas utama pada website Nestify, yang mencakup modul Login Akun Pemilik Kost, Registrasi Akun Pemilik Kost, dan Kelola Properti. Pengujian ini menunjukkan bahwa sebagian besar aspek dari fungsionalitas yang diuji telah beroperasi sesuai ekspektasi dengan tingkat keberhasilan keseluruhan mencapai 87.88%. Kinerja ini melampaui batas atas kategori 'Baik' (70-80%) dan sangat mendekati standar untuk kategori 'Sangat Baik' (89-100%), mengindikasikan bahwa fungsionalitas inti website Nestify telah solid. Meskipun demikian, masih teridentifikasi beberapa area pada berbagai modul yang memerlukan penyempurnaan untuk mencapai kualitas optimal.

Untuk, kegagalan-kegagalan yang ditemukan selama pengujian tersebar pada beberapa aspek validasi *input* dan penanganan kondisi tertentu. Pada fungsionalitas Login Akun Pemilik Kost, kegagalan terpusat pada validasi *input* terkait penanganan sistem terhadap batasan panjang karakter maksimal untuk *field* email. Sistem teramati belum memberikan respons atau pesan peringatan yang akurat ketika menerima data *input* email yang panjangnya melebihi batas maksimal, seperti yang terdokumentasi pada kasus uji TC_LOGIN_PK_R09. Untuk fungsionalitas Registrasi Akun Pemilik Kost, teridentifikasi kekurangan dalam validasi unik email, di mana sistem masih mengizinkan registrasi menggunakan alamat email yang seharusnya sudah terdeteksi terdaftar, seperti pada kasus uji TC_REG_PK_R07. Selanjutnya, pada fungsionalitas Kelola Properti, ditemukan dua kegagalan utama: pertama, pada fitur Tambah Properti, sistem gagal melakukan validasi format *file* foto dengan benar dan menerima *file* dengan format yang tidak seharusnya didukung tanpa pesan error yang memadai. Kedua, pada fitur Edit Properti, sistem memperbolehkan penyimpanan data properti meskipun *field* wajib seperti Nama Properti dikosongkan, yang seharusnya tidak diizinkan. Kekurangan-kekurangan dalam validasi *input*, penanganan batas, dan validasi kondisi spesifik ini merupakan jenis kesalahan yang teridentifikasi pada modul-modul yang diuji dan memerlukan perhatian untuk perbaikan lebih lanjut.

4.2 Saran

Menindaklanjuti temuan-temuan dan kesimpulan tersebut, beberapa saran diajukan untuk perbaikan aplikasi website Nestify serta sebagai arahan untuk penelitian selanjutnya:

1. Untuk Tim Pengembang Aplikasi Nestify:
 - Disarankan untuk segera menindaklanjuti dan memperbaiki temuan kegagalan fungsional. Perbaikan utama mencakup implementasi validasi yang lebih akurat terhadap batas panjang email pada fitur Login (TC_LOGIN_PK_R09), penguatan mekanisme validasi email unik pada fitur Registrasi (TC_REG_PK_R07), serta perbaikan pada validasi format *file* foto dan penanganan *field* wajib pada fitur Kelola Properti (TC_KP_PK_R04, TC_KP_PK_R07).
 - Setelah proses perbaikan *bug* dilakukan, sangat penting untuk melaksanakan siklus pengujian ulang (*re-testing*) pada kasus-kasus uji yang sebelumnya gagal dan pengujian regresi pada fungsionalitas terkait. Langkah ini krusial untuk memastikan semua kesalahan teratasi dan tidak menimbulkan masalah baru.
2. Untuk Penelitian Selanjutnya:

- Dianjurkan untuk memperluas cakupan pengujian fungsional ke fitur-fitur lain dalam *website* Nestify yang belum diuji dalam penelitian ini, seperti pengelolaan detail kamar, manajemen data penyewa, fungsionalitas *dashboard* informasi, pengelolaan tagihan, dan penanganan komplain, guna mendapatkan validasi sistem yang lebih komprehensif.
- Selain pengujian fungsional, disarankan untuk melibatkan pengujian non-fungsional. Pengujian kinerja untuk mengukur waktu respons dan skalabilitas sistem, serta pengujian keamanan untuk mengidentifikasi dan mengatasi potensi kerentanan pada validasi *input* pengguna, dapat memberikan evaluasi kualitas perangkat lunak yang lebih menyeluruh.

DAFTAR PUSTAKA

- Aryandana, I. G. S., Permanasari, A. E., & Adji, T. B. (2020). Comparing method equivalence class partitioning and boundary value analysis with study case add medicine module. *IOP Conference Series: Materials Science and Engineering*, 732(1), 012072. <https://doi.org/10.1088/1757-899X/732/1/012072>
- Bajjouk, M., Ehsan Rana, M., Reka Ramachandiran, C., & Chelliah, S. (2021). Software testing for reliability and quality improvement. In *Journal of Applied Technology and Innovation* (Vol. 5, Issue 2).
- Bhat, A., & Quadri, S. M. K. (2015). Equivalence class partitioning and boundary value analysis - A review. *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, 1557–1562.
- Dobslaw, F., Feldt, R., & Gomes de Oliveira Neto, F. (2023). Automated black-box boundary value detection. *PeerJ Computer Science*, 9, e1625. <https://doi.org/10.7717/peerj-cs.1625>
- Fauzan, R., Siahaan, D., Rochimah, S., & Triandini, E. (2021). A Different Approach on Automated Use Case Diagram Semantic Assessment. *International Journal of Intelligent Engineering and Systems*, 14(1), 496–505. <https://doi.org/10.22266/ijies2021.0228.46>
- Fauzi, N. F. (2024). *IMPLEMENTASI METODE USER CENTERED DESIGN PADA PERANCANGAN ANTARMUKA APLIKASI INDEKOS*.
- Hu, V. C., Kuhn, R., & Yaga, D. (2017). *Verification and test methods for access control policies models*. <https://doi.org/10.6028/NIST.SP.800-192>
- Jaya, T. S. (2018). Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung). *Jurnal Informatika: Jurnal Pengembangan IT*, 3(1), 45–48. <https://doi.org/10.30591/jpit.v3i1.647>
- Liew, A. (2024, July 14). *Equivalence Partitioning and Boundary Value Analysis*. <https://Alanliew.Medium.Com/Equivalence-Partitioning-and-Boundary-Value-Analysis-C940a0c120f5>.
- Lintang Ardias, A., Elsera Astrianty, L., Sains dan Teknologi, F., Teknologi Yogyakarta Jl Siliwangi, U., Lor, J., Mlati, K., & Istimewa Yogyakarta, D. (2024). *PENGEMBANGAN SISTEM INFORMASI DESA PAGELAK MADUKARA BERBASIS WEB DEVELOPMENT OF A WEB-BASED INFORMATION SYSTEM FOR PAGELAK MADUKARA VILLAGE*. 13(1).
- Mechtaev, S. (2017). Test-equivalence Analysis for Automatic Patch Generation. In *ACM Trans. Softw. Eng. Methodol.* 1, 1, Article (Vol. 1). <http://simplesystems.org/libtiff/>
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). In *Jurnal Ilmiah Teknologi Informasi Terapan: Vol. I* (Issue 3).
- Putu, I., Prayudha, A., Hartati, R. S., & Divayana, Y. (n.d.). I Boundary Value Analysis Testing Techniques on Learning Management System Applications. *International Journal of Engineering and Emerging Technology*, 4(2).
- Reid, S. C. (n.d.). *An Empirical Analysis of Equivalence Partitioning, Boundary Value Analysis and Random Testing*.
- Santi, P., Afwani, R., Albar, M., Anjarwani, S., & Mardiansyah, A. Z. (2022). *Black Box Testing with Equivalence Partitioning and Boundary Value Analysis Methods (Study Case: Academic Information System of Mataram University)* (pp. 207–219). https://doi.org/10.2991/978-94-6463-084-8_19
- Selviana, E. Z., Wahanani, H. E., & Aditiawan, F. P. (2023). Black Box Testing in Community Service Systems Using Boundary Value Analysis and Cause Effect Graph Methods (Case Study of Jombang District Community and Village Empowerment Service). *East Asian Journal of Multidisciplinary Research*, 2(10), 4161–4184. <https://doi.org/10.55927/eajmr.v2i10.6435>
- Srivastava, A., Saxena, D., & Dwivedi, N. (2024). A Review Paper on Software Testing. *International Research Journal of Modernization in Engineering Technology and Science*. <https://doi.org/10.56726/irjmets50146>
- Totade, S. K., Tayde, T., & Dhole, P. (2023). *1699492898_5a0d3671b9_7_irjiet*. <https://doi.org/https://doi.org/10.47001/IRJIET/2023.710089>
- Wardah Gracillaria Suharyono, F., Kartini, K., & Junaidi, A. (2024). *PENERAPAN METODE BOUNDARY VALUE ANALYSIS DAN EQUIVALENCE PARTITIONING DALAM PENGUJIAN BLACK BOX*

- UNTUK APLIKASI SIADITA. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(1), 1013–1020. <https://doi.org/10.36040/jati.v8i1.8921>
- Yuwantoko, A. M., Siahaan, D. O., & Ahmadiyah, A. S. (2017). *Pembuatan Kakas Bantu untuk Mendeteksi Ketidaksesuaian Diagram Urutan (Sequence Diagram) dengan Diagram Kasus Penggunaan (Use Case Diagram)*.
- Zein, S., Salleh, N., & Grundy, J. (2016). A systematic mapping study of mobile application testing techniques. *Journal of Systems and Software*, 117, 334–356. <https://doi.org/10.1016/j.jss.2016.03.065>