

**PENGUJIAN APLIKASI *PANIC BUTTON* UNTUK PANTI
LANSIA DEWANATA CILACAP MENGGUNAKAN METODE
ROBUTNESS TESTING DAN *EQUIVALENCE PARTITIONING***

SKRIPSI

**Diajukan untuk memenuhi sebagian persyaratan mendapatkan gelar Strata
Satu Program Studi Sistem Informasi**



Disusun oleh :

ARIF BIKI

202231003

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU KOMPUTER
UNIVERSITAS NAHDLATUL ULAMA AL GHAZALI
CILACAP
2026**

HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya menyatakan bahwa penulisan skripsi dengan judul “Pengujian Aplikasi *Panic Button* Untuk Panti Lansia Dewanata Cilacap Menggunakan Metode *Robutness Testing* Dan *Equivalence Partitioning*” adalah karya saya dengan arahan dari pembimbing dan belum diajukan kepada pihak manapun. Sumber yang dikutip dalam skripsi ini telah dicantumkan dalam daftar pustaka.

Dengan demikian pernyataan ini dibuat dengan sebenarnya. Apabila di kemudian hari terdapat ketidaksesuaian dengan pernyataan ini, maka saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Cilacap, 13 Agustus 2026



NIM. 202231003

LEMBAR PENGESAHAN

Skripsi saudara

Nama : Arif Biki
NIM : 202231003
Fakultas/Prodi : Fakultas MIKOM / Sistem Informasi
Judul : Pengujian Aplikasi *Panic Button* untuk Panti Lansia
Dewanata Cilacap menggunakan Metode *Robutness*
Testing dan Equivalence Partitioning

Telah disidangkan oleh Dewan Penguji Fakultas Matematika dan Ilmu Komputer
Universitas Nahdlatul Ulama AlGhazali (UNUGHA) Cilacap pada hari / tanggal :

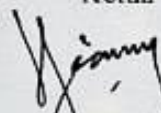
Rabu, 12 Agustus 2026

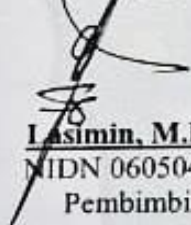
Dan dapat diterima sebagai pemenuhan tugas akhir mahasiswa Program Strata Satu
Program Studi Sistem Informasi (S.Kom) Fakultas Matematika dan Ilmu Komputer
(FMIKOM) pada Universitas Nahdlatul Ulama AlGhazali (UNUGHA) Cilacap.

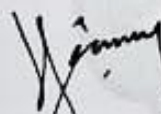
Cilacap, 15 Agustus 2026

Dewan Sidang

Ketua


Verry, S. T., M. Kom
NIDN 0628068001
Penguji I

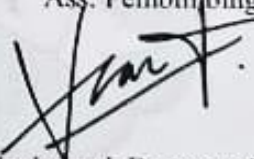

Lasimin, M. Kom
NIDN 0605048602
Pembimbing


Verry, S. T., M. Kom
NIDN 0628068001

Sekretaris


Lasimin, M. Kom
NIDN 0605048602
Penguji II


Eri Nur Shofi'i, S.H., M.A.
NIDN 0602029501
Ass. Pembimbing


M. Noviansyah Dasaprawira, M.Pd
NIDN 0610119301

Mengetahui

Dekan Fakultas Matematika dan Ilmu Komputer




Mochamad T. A. Aziz Zein, S.Si., M.Kom
NIDN 2125098601

HALAMAN NOTA KONSULTAN

Lasimin, M.Kom.

Dosen Fakultas Matematika dan Ilmu Komputer
Universitas Nahdlatul Ulama Al Ghazali Cilacap

NOTA KONSULTAN

Hal : Skripsi Saudara Arif Biki
Lampiran : -

Yth.

Dekan Fakultas Matematika dan Ilmu Komputer
Universitas Nahdlatul Ulama Al Ghazali Cilacap
Di Cilacap.

Assalamualaikum Wr. Wb.

Setelah membaca, memeriksa dan melakukan perbaikan seperlunya maka skripsi saudara :

Nama : Arif Biki
NIM : 202231003
Fakultas/Prodi : Fakultas MIKOM / Sistem Informasi
Judul : Pengujian Aplikasi *Panic Button* untuk Panti Lansia
Dewanata Cilacap menggunakan Metode *Robutness Testing* dan *Equivalence Partitioning*

Dapat diajukan ke Fakultas Matematika dan Ilmu Komputer (FMIKOM) Universitas Nahdlatul Ulama AlGhazali (UNUGHA) Cilacap untuk memenuhi syarat memperoleh gelar Strata Satu (S1).

Wassalamualaikum Wr. Wb.

Cilacap, 15 Agustus 2026
Konsultan

Lasimin, M.Kom.
NIDN 0605048602

NOTA PEMBIMBING

Cilacap, 15 Agustus 2026

Kepada Yth :

Fakultas Matematika dan Ilmu Komputer (FMIKOM)
Universitas Nahdlatul Ulama Al Ghazali (UNUGHA) Cilacap
Di Cilacap

Assalamualaikum Wr. Wb.

Setelah melakukan bimbingan, telaah, arahan, dan koreksi tahap penulisan skripsi saudara :

Nama : Arif Biki
NIM : 202231003
Fakultas/Prodi : Fakultas MIKOM / Sistem Informasi
Judul : Pengujian Aplikasi *Panic Button* untuk Panti Lansia
Dewanata Cilacap menggunakan Metode *Robutness Testing* dan *Equivalence Partitioning*

Kami berpendapat bahwa skripsi tersebut sudah dapat diajukan ke sidang skripsi. Bersamaan ini kami kirimkan skripsi tersebut, semoga dapat segera disidangkan. Atas perhatiannya kami ucapkan terima kasih.

Wassalamualaikum Wr. Wb.

Mengetahui

Pembimbing I



Verry, S. T., M.Kom
NIDN 0628068001

Pembimbing II



M. Noviansyah Dasaprawira, M.Pd
NIDN 0610119301

MOTTO

“ Bukan cuma jago di ***Land of Dawn***, di dunia nyata pun tugas tuntas. ***Mythic Imortal*** didapat, gelar **Sarjana** pun kelarr ”

-

“ Lulus ujian **rank** sampai ***Mythic Imortal***, lulus sidang skripsi dengan percaya diri ”

~ Namanya juga manusia ~

PERSEMBAHAN

Keberhasilan ini ku persembahkan untuk Ayah dan Ibu, dua malaikat tanpa sayap yang tak pernah lelah mendoakan dan percaya padaku saat dunia meragukan. Terima kasih telah menjadi rumah tempatku pulang. Tak lupa juga untuk kakak kakakku, terima kasih sudah menjadi pendengar yang baik dan pelindung untuk diriku. Dukunganmu sangat berarti bagiku.

KATA PENGANTAR

Puji Syukur kepada Alloh SWT atas segala limpahan nikmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi ini. Sholawat dan salam senantiasa penulis hanturkan kepada Rosulluloh SAW sebagai pembimbing seluruh umat manusia.

Selama penulisan Skripsi ini banyak menerima bantuan dan dukungan sehingga dapat menyelesaikan. Oleh karena itu, penulis mengucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Mochamad T. A. Aziz Zein, S.Si., M.Kom sebagai Dekan FMIKOM UNUGHA.
2. Bapak Verry, S. T., M.Kom sebagai pembimbing I.
3. Bapak M. Noviansyah Dasaprawira, M.Pd sebagai pembimbing II.
4. Ayah dan Ibu saya serta kakak - kakak saya yang telah men-support.
5. Bapak – ibu dosen Program Studi Sistem Informasi UNUGHA.
6. Semua pihak yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna karena adanya keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karena itu, semua kritik dan saran yang bersifat membangun akan penulis terima dengan senang hati. Penulis berharap, semoga skripsi ini dapat bermanfaat bagi semua pihak yang memerlukan.

Cilacap, 13 Agustus 2026

ABSTRAK

Sistem tanggap darurat yang andal sangat diperlukan bagi Pusat Pelayanan Sosial Lanjut Usia Dewanata Cilacap untuk melindungi para lansia dari potensi ancaman, mengingat keterlambatan respons dapat berakibat fatal. Sebuah aplikasi *Panic Button* berbasis Android telah dikembangkan sebagai sarana cepat untuk memberi tahu staf mengenai situasi darurat. Namun, pengguna sering kali mengalami kepanikan dalam situasi darurat, yang dapat menyebabkan kesalahan input, kesulitan interaksi, atau kegagalan program. Oleh karena itu, pengujian perangkat lunak yang menyeluruh sangat penting untuk memastikan bahwa program memiliki ketahanan tinggi, tidak mengalami *crash*, serta berkinerja optimal dalam berbagai kondisi.

Tujuan penelitian ini adalah untuk mengevaluasi dan menerapkan teknik *Equivalence Partitioning* dan *Robustness Testing* pada aplikasi *Panic Button* berbasis Android. Selain itu, penelitian ini bertujuan untuk menilai hasil pengujian guna mengukur efikasi, stabilitas, serta keandalan fungsional dan non-fungsional sistem saat menghadapi kondisi ekstrem, input yang valid, maupun input yang tidak valid.

Teknik pengujian yang digunakan, yaitu *Robustness Testing* (RT) dan *Equivalence Partitioning* (EP), diklasifikasikan sebagai *Black Box Testing*. Untuk mengevaluasi ketahanan sistem dalam menangani kesalahan, *robustness testing* dilakukan dengan membuat skenario input yang mencakup kondisi ekstrem dan melampaui batas-batas normal (*out-of-bounds*). Sementara itu, guna mengoptimalkan efektivitas pengujian dan cakupan fungsional, modul *Login* dan *Registration* menggunakan *Equivalence Partitioning*, yang membagi data input ke dalam kelas-kelas ekuivalensi (partisi valid dan tidak valid).

Berdasarkan hasil pengujian, 94 kasus uji berhasil dan 13 gagal dari total 107 kasus uji yang dijalankan pada 7 sub-modul di dalam 2 modul utama (*Login* dan *Register*). Pengujian ketahanan (*robustness testing*) pada sub-modul Kata Sandi Login menghasilkan tingkat keberhasilan terendah sebesar 50,00% (Tidak Efektif), sedangkan pengujian *Equivalency Partitioning* pada modul Register menghasilkan tingkat keberhasilan tertinggi sebesar 98,07% (Sangat Efektif). Efektivitas keseluruhan aplikasi *Panic Button* sebesar 75,79% menunjukkan tingkat kualitas

dan keandalan yang dikategorikan sebagai "Cukup Efektif", namun fitur validasi input tertentu masih perlu ditingkatkan.

Kata kunci : *Panic Button, Black Box Testing, Robustness Testing, Equivalence Partitioning, Panti Lansia Dewanata.*

ABSTRACT

A dependable emergency response system is necessary for the Dewanata Cilacap Social Service Center for the Elderly to protect senior citizens from potential threats, where a delayed reaction could be lethal. An Android-based Panic Button app was created as a quick way to alert employees of emergencies. However, users frequently experience panic in emergency situations, which may result in incorrect inputs, interaction difficulties, or program failures. Therefore, thorough software testing is essential to guaranty that the program has high resilience, doesn't crash, and performs at its best under a variety of circumstances.

The objective of this study is to evaluate and apply Equivalency Partitioning and Robustness Testing techniques to an Android-based Panic Button application. Additionally, it aims to assess the test results in order to gage the system's efficacy, stability, and functional and non-functional reliability when dealing with harsh conditions, valid inputs, and invalid inputs.

The testing techniques used, namely Robustness Testing (RT) and Equivalency Partitioning (EP), are classified as Black Box Testing. In order to evaluate the system's error-handling resilience, robustness testing is carried out by creating input scenarios that entail extreme conditions and go beyond typical boundaries (out-of-bounds). In the meantime, to optimize testing effectiveness and functional coverage, the Login and Registration modules employ Equivalence Partitioning, which divides input data into equivalence classes (valid and invalid partitions).

According to test findings, 94 test cases succeeded and 13 failed out of a total of 107 test cases that were run across 7 sub-modules inside 2 major modules (Login and Register). Robustness testing on the Login Password sub-module produced the lowest success rate at 50.00% (Ineffective), whereas Equivalency Partitioning testing on the Register module produced the highest success rate at 98.07% (Highly Effective). The Panic Button application's overall effectiveness of 75.79% indicates a quality and dependability level categorized as "Quite Effective," however certain input validation features still need to be improved.

Keywords: *Panic Button, Black Box Testing, Robustness Testing, Equivalence Partitioning, Dewanata Elderly Care Home.*

DAFTAR ISI

HALAMAN PERNYATAAN KEASLIAN SKRIPSI.....	ii
PENGESAHAN	iii
NOTA PEMBIMBING	v
MOTTO.....	vi
PERSEMBAHAN	vii
KATA PENGANTAR.....	viii
ABSTRAK	ix
ABSTRACT	xi
DAFTAR ISI	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
A. Latar Belakang.....	1
B. Rumusan Masalah.....	2
C. Batasan Masalah	3
D. Tujuan Penelitian.....	3
E. Manfaat Penelitian	3
BAB II KAJIAN PUSTAKA	5
A. Studi Pustaka	5
B. Landasan Teori	11
1. Konsep dasar aplikasi <i>Panic Button SOS</i>	11
2. Pengujian Perangkat Lunak (<i>Software Testing</i>)	13
3. Metode <i>Robustness Testing</i> (RT)	13
4. Metode <i>Equivalence Partitioning</i> (EP)	14

BAB III METODOLOGI PENELITIAN.....	16
A. Waktu Penelitian.....	16
B. Tempat Penelitian	16
C. Prosedur Penelitian	16
D. Jadwal Penelitian	44
BAB IV PEMBAHASAN.....	45
A. Hasil Pengujian Metode <i>Robustness Testing</i>	45
B. Hasil Pengujian Metode <i>Equivalence Partitioning</i>	51
C. Pencatatan Analisa Hasil Pengujian	55
BAB V PENUTUP	60
A. Kesimpulan.....	60
B. Saran	61
DAFTAR PUSTAKA.....	62

DAFTAR TABEL

Tabel 2 1 Penelitian Terkait	5
Tabel 3 1 Identifikasi User	21
Tabel 3 2 Validasi Check Modul Daftar	23
Tabel 3 3 Validasi Check Modul Login	23
Tabel 3 4 Skenario Robustness Testing Modul login (Email)	24
Tabel 3 5 Skenario Robustness Testing Modul Login (Password)	24
Tabel 3 6 Skenario Robustness Testing Modul Daftar (Nama)	25
Tabel 3 7 Skenario Robustness Testing Modul Daftar (Email)	26
Tabel 3 8 Skenario Robustness Testing Modul Daftar (Password)	27
<i>Tabel 3 9 Skenario Equivalence Partitioning Modul login</i>	<i>28</i>
Tabel 3 10 Skenario Equivalence Partitioning Modul Daftar	30
Tabel 3 11 Standar acuan efektivitas	43
Tabel 3 12 Jadwal Penelitian	44
Tabel 4 1 Hasil Pengujian Metode Robustness Testing Modul Daftar (Nama)....	45
Tabel 4 2 Hasil Pengujian Metode Robustness Testing Modul Daftar (Email)....	46
Tabel 4 3 Hasil Pengujian Metode Robustness Testing Modul Daftar (Password)	47
Tabel 4 4 Hasil Pengujian Metode Robustness Testing Modul login (Email).....	48
Tabel 4 5 Hasil Pengujian Metode Robustness Testing Modul Login (Password)49	
Tabel 4 6 Hasil Pengujian Metode Equivalence Partitioning Modul login	51
Tabel 4 7 Hasil Pengujian Metode Equivalence Partitioning Modul Daftar	52
Tabel 4 8 Hasil analisa modul daftar (Nama)	56
Tabel 4 9 Hasil analisa modul daftar (Email)	57
Tabel 4 10 Hasil analisa modul daftar (Password).....	57
Tabel 4 11 Hasil analisa modul login (Email)	58
Tabel 4 12 Hasil analisa modul login (Password).....	58
Tabel 4 13 Hasil analisa modul login.....	59
Tabel 4 14 Hasil analisa modul daftar.....	59

DAFTAR GAMBAR

Gambar 2 1 Diagram Block Client	12
Gambar 2 2 Diagram Block Server	12
Gambar 3 1 Tahapan Penelitian	17
Gambar 3 2 Tampilan Awal	19
Gambar 3 3 Modul Daftar	19
Gambar 3 4 Modul Login	19
Gambar 3 5 Tampilan Home	20
Gambar 3 6 Tampilan About	20
Gambar 3 7 Modul Profil	20
Gambar 3 8 Modul Tahapan Kegiatan	20
Gambar 3 9 Modul Landasan Hukum	21
Gambar 3 10 Modul Visi Misi	21

BAB I PENDAHULUAN

A. Latar Belakang

Pengujian aplikasi (*software*) merupakan sebuah tahapan pemeriksaan suatu program guna memastikan sistem beroperasi sesuai spesifikasinya dan jika terdapat kesalahan maka dilakukan diperbaiki agar program sesuai tujuan dan memiliki kualitas yang baik [1]. Pengujian ini bertujuan untuk menguji fungsional dan memastikan apakah aplikasi ini dapat bekerja sesuai pada kualitas aplikasi [2]. Mengingat pentingnya peran pengujian aplikasi ini, pengujian perangkat lunak dilakukan agar bisa memberikan jaminan kualitas dari perangkat lunak dan tidak ada kesalahan pada sistem [3]. Pengujian ini memastikan bahwa aplikasi berfungsi sesuai dengan yang diharapkan, bebas dari cacat, dan mampu bekerja dengan andal dalam berbagai kondisi [4].

Tahapan pengujian aplikasi ini sangat penting diterapkan pada aplikasi *panic button* yang dirancang untuk menangani kondisi darurat agar sistem beroperasi tanpa adanya hambatan [5]. Berdasarkan jurnal [6] aplikasi *panic button* digunakan untuk para lansia di Panti Pelayanan Sosial Lanjut Usia Dewanata Cilacap. Walaupun panti ini menyediakan lingkungan yang aman dan terawat, potensi terjadinya insiden darurat tetap ada. Keterlambatan penanganan hanya dalam hitungan menit dapat berakibat fatal kepada para lansia. Aplikasi *Panic Button* muncul sebagai solusi cepat dan efektif untuk para lansia meminta bantuan dalam pertolongan darurat [6].

Kebutuhan pertolongan untuk para lansia selaras dengan tujuan perancangan aplikasi ini yang mana aplikasi dirancang untuk mengirimkan informasi penting (keadaan darurat) kepada pihak perawat atau petugas yang bertugas. Guna untuk menjamin kebutuhan tujuan tersebut, digunakan metode *Robustness Testing* dan *Equivalence Partitioning* yang bertujuan untuk mengidentifikasi potensi cacat (*bug*) dan mengevaluasi sejauh mana aplikasi *Panic Button* mampu bertahan dan berfungsi secara optimal di bawah tekanan, sehingga dapat memberikan jaminan keselamatan bagi penggunaanya. Metode *Robustness Testing* adalah jenis pengujian non-fungsional yang krusial untuk mengevaluasi ketahanan dan keandalan suatu sistem perangkat lunak, terutama

saat dihadapkan pada data masukan yang tidak terduga, tidak valid, atau berada di luar batas normal sistem (*out of bounds input*) [7]. Tujuan utama pengujian ini adalah untuk menentukan sejauh mana aplikasi dapat terus beroperasi dengan benar [8]. Metode *Equivalence Partitioning* adalah teknik pengujian *Black Box* yang fokus pada efisiensi pengujian kualitas aplikasi dengan cara membagi data masukan menjadi partisi-partisi yang dianggap ekuivalen, sehingga hanya perlu mengambil satu kasus uji dari setiap partisi [9].

Kondisi darurat sering kali memicu kepanikan yang menyebabkan kesalahan interaksi pengguna terhadap perangkat. Pengujian metode *Robustness Testing*, aplikasi berisiko mengalami *crash* saat menerima input abnormal dari pengguna yang panik. Sementara itu, metode *Equivalence Partitioning* menjadi esensial untuk memvalidasi bahwa sistem memberikan respons yang tepat pada setiap kategori input, sehingga meminimalisir kegagalan fungsi di saat yang paling dibutuhkan. Berdasarkan jurnal [6] aplikasi *panic botton* dan jurnal penelitian [1], [2], [7], [10], [11] menerapkan metode *Robustness Testing* dan *Equivalence Partitioning* peneliti tertarik untuk melakukan penelitian pengujian *Robustness Testing* dan *Equivalence Partitioning* pada aplikasi *panic button*. Hasil penelitian ini diharapkan dapat digunakan sebagai *referensi* untuk pengembangan sistem pada masa depan yang dapat menyesuaikan kebutuhan dan ekspetasi pengguna, serta meningkatkan kualitas sistem.

B. Rumusan Masalah

Berdasarkan latar belakang yang disampaikan, maka dapat diidentifikasi rumusan masalah sebagai berikut:

1. Bagaimana penerapan *Robustness Testing* pada aplikasi *Panic Button*?
2. Bagaimana penerapan *Equivalence Partitioning* pada aplikasi *Panic Button*?
3. Bagaimana hasil pengujian *Robustness Testing* dan *Equivalence Partitioning* pada aplikasi *panic button*?

C. Batasan Masalah

Berdasarkan rumusan masalah diperoleh batasan masalah adalah objek pengujian hanya terbatas pada aplikasi *Panic Button* berbasis Android.

D. Tujuan Penelitian

Sehubungan dengan masalah yang telah dirumuskan, berikut adalah tujuan penelitian ini:

1. Mengetahui bagaimana penerapan *Robustness Testing* pada aplikasi *Panic Button*.
2. Mengetahui bagaimana penerapan *Equivalence Partitioning* pada aplikasi *Panic Button*.
3. Mengetahui hasil pengujian *Robustness Testing* dan *Equivalence Partitioning* pada aplikasi *panic button*.

E. Manfaat Penelitian

Berdasarkan latar belakang, rumusan masalah, dan tujuan penelitian, diharapkan bahwa penelitian ini akan bermanfaat untuk hal-hal berikut :

1. Manfaat Teoritis
 - a. Penelitian ini diharapkan dapat memberikan kontribusi pada literatur ilmu pengetahuan, khususnya dalam bidang pengujian perangkat lunak (*software testing*) dengan menerapkan metode *Robustness Testing* dan *Equivalence Partitioning* pada aplikasi *Panic Button*.
 - b. Penelitian ini dapat menjadi model referensi atau panduan bagi peneliti atau pengembang lain yang ingin melakukan pengujian *Robustness Testing* dan *Equivalence Partitioning* serupa, terutama pada aplikasi yang berkaitan dengan keselamatan dan keamanan.
2. Manfaat Praktis
 - I. Bagi Peneliti :
 - a. Peneliti akan mendapatkan pengalaman langsung dan pemahaman mendalam dalam mengimplementasikan pengujian *Equivalence Partitioning* dan *Robustness Testing*.

- b. Peneliti akan terlatih untuk berpikir secara sistematis dan analitis dalam mengidentifikasi cacat (*bug*) dan mengevaluasi keandalan sebuah sistem perangkat lunak.
- c. Hasil penelitian ini dapat menjadi bukti nyata kemampuan peneliti dalam bidang *Software Testing*, yang sangat relevan untuk karir di industri teknologi.

II. Bagi Pengembang Aplikasi:

- a. Hasil pengujian dari penelitian ini dapat membantu pengembang untuk mengidentifikasi cacat yang ada dan memperbaikinya, sehingga aplikasi *Panic Button* menjadi lebih andal dan stabil.
- b. Pengembang akan mendapatkan wawasan tentang bagaimana aplikasi mereka bereaksi terhadap input yang tidak valid dan tidak terduga, yang mungkin tidak terpikirkan dalam proses pengembangan.

III. Bagi Pengguna Aplikasi:

- a. Dengan meningkatnya kualitas dan keandalan aplikasi, pengguna akan merasa lebih aman dan percaya diri saat menggunakan aplikasi *Panic Button* dalam situasi darurat, karena mereka tahu aplikasi tersebut akan berfungsi sebagaimana mestinya.

BAB II KAJIAN PUSTAKA

A. Studi Pustaka

Pada penelitian ini mencari sumber atau *referensi* terhadap beberapa penelitian yang berkaitan dengan penelitian ini sebagai acuan dalam Pengujian *Robustness Testing* dan *Equivalence Partitioning* pada aplikasi *Panic Button*. Berikut ini adalah beberapa penelitian yang telah dilakukan oleh peneliti sebelumnya antara lain:

Tabel 2 1 Penelitian Terkait

No.	Peneliti	Keterangan Penelitian
1	Alvin Valerian, Delon Bernady, Ferry Wahyudi, Jonathan Pratama Dinatha, One Dhylan B. (2026)	Judul : <i>Implementasi Unit Testing, Integration Testing, System Testing, Dan Validation Testing Pada Aplikasi Berbasis Website.</i> Metode : <i>Unit Testing, Integration Testing, System Testing, Validation Testing, Black Box Testing, White Box Testing.</i> Hasil Penelitian : Berdasarkan hasil pengujian, sistem telah memenuhi standar kualitas melalui <i>Unit Testing, Integration Testing, System Testing, dan Validation Testing</i> . <i>Unit Testing</i> memastikan fitur CRUD berfungsi tanpa kesalahan, sementara <i>Integration Testing</i> membuktikan interaksi antar modul berjalan baik. <i>System Testing</i> menguji stabilitas sistem, dan <i>Validation Testing</i> menunjukkan kemudahan penggunaan serta keakuratan data. Hasil pengujian mengonfirmasi bahwa sistem memiliki integritas data yang baik dan beroperasi optimal[12].
2	Hasna Sarifatin M., Safiq Rosad, Edy Sulistyanto (2024)	Judul : <i>Implementasi Tombol Panik SOS Untuk Lansia Berbasis IoT Pada Panti Pelayanan Sosial Lanjut Usia Dewanata Cilacap</i> Metode : Berbasis Internet of Things Software : <i>Panic Button</i> Hasil Penelitian : Perancangan Tombol Panik SOS

No.	Peneliti	Keterangan Penelitian
		untuk Lansia Berbasis <i>Internet of Things</i> (IoT) memerlukan pertimbangan beberapa faktor, termasuk kemudahan penggunaan, keamanan, dan integrasi dengan teknologi lainnya. Tombol Panik SOS untuk Lansia Berbasis IoT dapat membantu meningkatkan keselamatan dan kesejahteraan lansia. Realibilitas perangkat Tombol Panik SOS terhadap lingkungan yang ada sangat penting untuk memastikan bahwa perangkat tersebut dapat berfungsi dengan baik dan efektif dalam situasi darurat[6].
3	Aufa Ardilla, Firza Prima Aditiawan, Yisti Vita Via (2024)	Judul : Penerapan Teknik Equivalence Partitioning Dan Boundary Value Analysis Dalam Pengujian Black Box Aplikasi Kediri Single Window For Investment (Studi Kasus : Dinas Penanaman Modal Dan Pelayanan Terpadu Satu Pintu Kota Kediri). Metode : Equivalence Partitioning Dan Boundary Value Analysis Hasil Penelitian : Hasil dari pengujian diperoleh nilai akhir efektivitas sistem yang dihitung dengan menjumlahkan nilai efektivitas dari masing-masing tabel dan dibagi dengan seluruh jumlah tabel sehingga hasil akhir diperoleh nilai sebesar 95,31%. Berdasarkan hasil perhitungan tersebut termasuk dalam kategori nilai sangat efektif. Dapat disimpulkan bahwa aplikasi Kediri Single Window for Investment milik Dinas Penanaman Modal dan Pelayanan Terpadu Satu Pintu Kota Kediri ini sudah berjalan sesuai fungsionalitas dan dapat dikatakan sudah sangat baik. Namun, di samping itu masih terdapat beberapa kesalahan dalam aplikasi sehingga diperlukan perbaikan agar aplikasi berfungsi lebih optimal lagi. Untuk penelitian selanjutnya, diharapkan dapat

No.	Peneliti	Keterangan Penelitian
		dilakukan pengujian menggunakan metode lain atau bisa dengan menggunakan tools khusus untuk melakukan pengujian. Dengan demikian, hasil penelitian dapat digunakan sebagai bahan pembandingan dan dapat saling melengkapi sehingga membantu DPMPTSP Kota Kediri dalam meningkatkan efektivitas aplikasi tersebut[10].
4	Farra Wardah Gracillaria Suharyono, Kartini, Achmad Junaidi Program (2024)	Judul : Penerapan Metode Boundary Value Analysis Dan Equivalence Partitioning Dalam Pengujian Black Box Untuk Aplikasi Siadita. Metode : Equivalence Partitioning Dan Boundary Value Analysis Hasil Penelitian : Hasil pengujian Black Box Testing menggunakan metode Boundary Value Analysis dan Equivalence Partitioning pada aplikasi SIADITA memberikan beberapa kesimpulan yang akan dijelaskan dalam bagian ini. Kesimpulan yang pertama yaitu, terdapat enam kelemahan sistem pada aplikasi SIADITA yang berpengaruh terhadap nilai efektivitas aplikasi di antaranya yaitu, pada halaman login tidak ada pemberitahuan/alert mengenai pengguna jika tidak sengaja tidak mengisi email, tidak ada sensitive case pada input ketika pengguna tidak sengaja menambahkan spasi dan karakter yang tidak sesuai seperti karakter huruf kecil dan besar, pada halaman input email tidak ada sistem validasi alamat email, halaman pusat bantuan tidak dapat melakukan autentikasi menuju aplikasi WhatsApp, tidak ada fungsi untuk membatasi input pada nama lengkap yang terdapat di halaman edit profile dan halaman pengisian survey, dan yang terakhir yaitu tidak ada fungsi yang membatasi input pada form kritik dan saran pada halaman kritik dan saran. Kesimpulan

No.	Peneliti	Keterangan Penelitian
		selanjutnya yaitu, aplikasi SIADITA memperoleh nilai efektivitas sebesar 91,66% dari total keseluruhan 144 test case dengan 126 test case hasil valid dan 18 test case hasil invalid yang mana nilai tersebut masuk ke dalam kategori sangat sangat efektif berdasarkan pada standart acuan Litbang Depdagri (1991), namun perlu adanya upaya perbaikan dan pengembangan lebih lanjut pada sistem untuk mencapai nilai efektivitas terbaik, sehingga fungsionalitasnya dapat beroperasi secara optimal[2].
5	Taufik Maulana, Apriade Voutama	Judul : Black Box Testing Dengan Teknik Equivalence Partitions Pada Website Dinas Pengendalian Penduduk Dan Keluarga Berencana Karawang Metode : Hasil Penelitian : kelebihan dan kelemahan Black Box Testing dengan Teknik Equivalence Partitions pada website kbinten.id. Kelebihannya yaitu Efisien, Dengan teknik Equivalence Partitions, pengujian dapat dilakukan dengan efisien dan efektif karena hanya perlu memilih beberapa data input yang mewakili setiap partisi. Kelemahannya yaitu Terbatas pada kondisi partisi yang diuji: Teknik Equivalence Partitions hanya menguji beberapa kondisi partisi yang mewakili input yang mungkin digunakan pengguna. Ini mungkin tidak cukup untuk menguji keseluruhan fungsionalitas perangkat lunak[13].
6	Syita Luginawati, Ari Purno Wahyu (2023)	Judul : Pengujian Perangkat Lunak Menggunakan Metode Regression Testing Pada Aplikasi Pembelajaran Matematika Untuk Siswa Tingkat Smp Berbasis Android Metode : <i>Black Box Testing</i> Hasil Penelitian : Dari hasil pengujian yang telah dilakukan pada aplikasi cerMath berbasis mobile android

No.	Peneliti	Keterangan Penelitian
		mendapatkan kesimpulan bahwa metode <i>black box</i> dengan teknik regression ini mempermudah tester dalam mengetahui kesesuaian alur fungsi dengan hasil yang diharapkan dan dapat melihat hasil pada setiap prosesnya dan sangat berpengaruh pada pembelajaran matematika berbasis android[14].
7	Andika Pratama, Aditya Gema Pratama, Hanifah Permatasari (2023)	Judul : Pengujian Perangkat Lunak Sistem Informasi Perpustakaan Menggunakan Metode <i>Black Box Testing</i> Metode : <i>Black Box Testing</i> Software : <i>Google Form</i> Hasil Penelitian : Pengujian perangkat lunak menggunakan Metode <i>Black Box Testing</i> dengan Teknik <i>Equivalence Partition</i> cocok bagi pemula karena lebih sederhana caranya. Hasil pengujian menunjukkan output beberapa form dari system informasi perpustakaan yang dibuat sudah tidak ada error, dan sudah sesuai dengan yang diharapkan, pada perancangan dengan validitas 100% [15].
8	I Dewa Made Widia, Sovia Rosalin, Salnan Ratih Asriningtias, Elta Sonalita (2022)	Judul : Black Box Testing Menggunakan Boundary Value Analysis dan Equivalence Partitioning pada Aplikasi Pengadaan Bahan Baku Batik dengan Pendekatan Use Case Metode : Black Box Testing Hasil Penelitian : Penggabungan metode Boundary Value Analysis dan Equivalence Partitioning dengan pendekatan use case untuk desain test case pada metode Black Box Testing sangat efisien dalam menemukan kesalahan aplikasi. Hal ini ditunjukkan nilai DRE 0.45 yang berarti 45% test case yang dibangun tidak lolos uji atau dapat menemukan kesalahan dari aplikasi. Saat ini pengujian masih dilakukan dari sisi fungsional aplikasi,

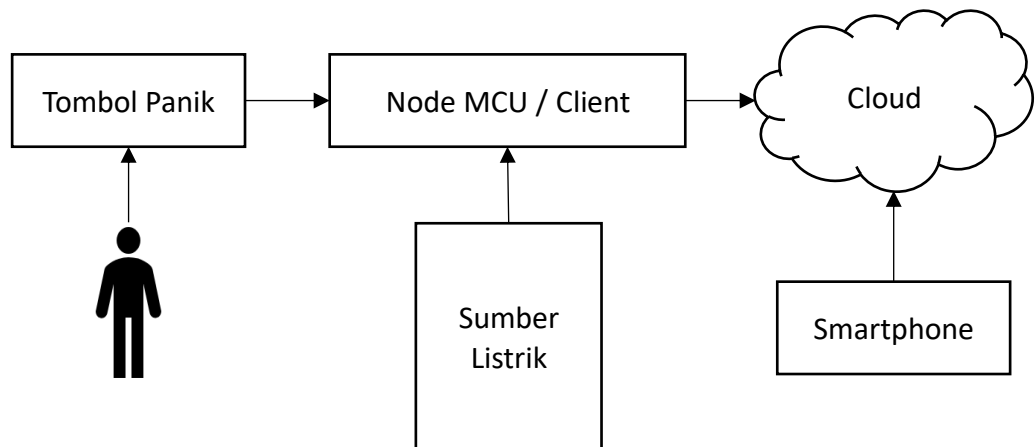
No.	Peneliti	Keterangan Penelitian
		maka untuk penelitian selanjutnya pengujian bisa dilakukan dari sisi non fungsional aplikasi. Selain itu bisa melakukan komparasi dengan metode Black Box Testing yang lainnya[16].
9	Nabil Fahlevi Abdi, S.R. Candra Nursari (2022)	Judul: Pengujian <i>Black Box</i> Pada Website Dengan Metode <i>Robustness Testing</i> Metode: <i>Robustness Testing</i> Hasil Penelitian : Berdasarkan hasil pengujian dengan menggunakan metode <i>Robustness Testing</i> pada website Eiger Adventure dapat disimpulkan dari 20 pengujian terdapat 19 pengujian yang berhasil dalam menangani kesalahan input atau gangguan pada sistem. Website ini dapat menangani kesalahan input yang tidak valid tersebut dan masih tetap berfungsi dengan baik, maka presentase keberhasilan pengujian ini adalah 95% [7].
10	Fadhila Tangguh Admojo1, M. Leo Adi Saputra (2022)	Judul : Analisis Sistem Keuangan Desa (SISKEUDes) di Kecamatan Muara Sugihan Menggunakan Metode <i>Black Box Testing</i> Metode : <i>Black Box Testing</i> Hasil Penelitian : Berdasarkan hasil pengujian black box dari 160 <i>test case</i> ditemukanya 24 defect, sehingga didapatkan nilai $DRE = \left(\frac{24}{160}\right) \times 100\% = 0,15 = 15\%$. Hal ini menunjukkan metode black bok dapat menemukan kesalahan atau defect pada aplikasi Siskeudes sebesar 15% [3].

B. Landasan Teori

1. Konsep dasar aplikasi *Panic Button SOS*

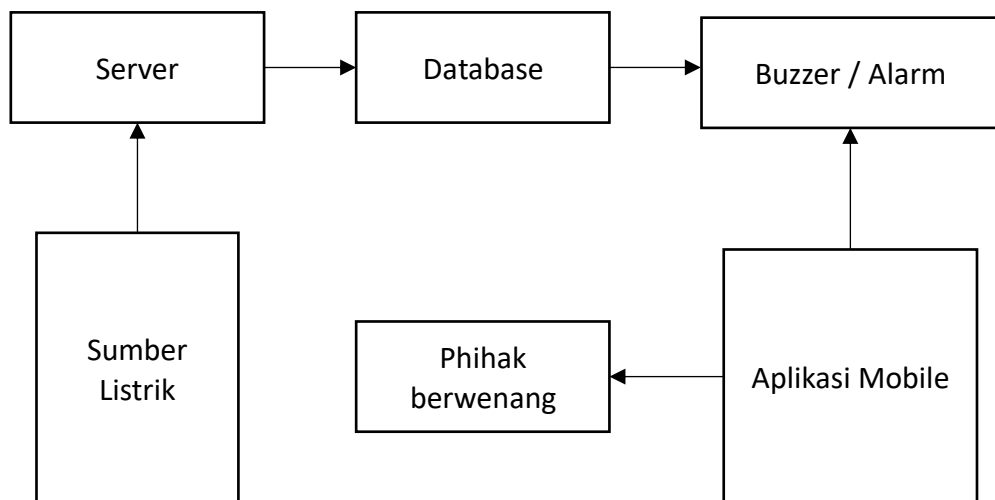
Panic Button SOS adalah sistem keamanan yang dirancang untuk memberikan tanggapan cepat dalam situasi darurat [17]. Fitur Panic Button SOS merepresentasikan lini pertahanan pertama dalam menghadapi situasi darurat. Dengan menekan tombol SOS, pengguna dapat langsung mengirimkan sinyal darurat ke pihak-pihak terkait yang telah ditentukan seperti admin atau pengguna lainnya [6]. Tujuan utamanya adalah untuk menjembatani waktu kritis antara insiden terjadi dan datangnya bantuan. Sistem ini bekerja dengan efisiensi, memungkinkan pengguna mengirim permintaan bantuan real-time tanpa perlu repot melakukan panggilan telepon.

Sistem keamanan pada jurnal [6] ini beroperasi secara end-to-end mulai dari perangkat keras hingga perangkat lunak, ketika pengguna mengaktifkan tombol panik, sinyal darurat segera ditangkap oleh unit mikrokontroler (Node MCU/Client) yang bertindak sebagai jembatan data. Selanjutnya, data tersebut diteruskan ke layanan Cloud untuk diproses dan diolah. Setelah diproses, Cloud bertanggung jawab mengirimkan notifikasi darurat secara real-time ke smartphone atau perangkat pihak berwenang atau pengguna yang telah ditentukan. Seluruh proses ini didukung oleh sistem baterai terpisah yang menyediakan daya kritis untuk tombol panik dan Node MCU, menjamin sistem tetap fungsional dan dapat diandalkan bahkan saat terjadi pemadaman daya utama. Secara keseluruhan, diagram ini memvisualisasikan alur sistem yang komprehensif, mulai dari inisiasi di perangkat keras hingga diterimanya peringatan darurat melalui infrastruktur cloud.



Gambar 2 1 Diagram Block Client

Pada bagian backend pada jurnal [6], Server berperan sentral dalam mengelola seluruh respons darurat. Setelah menerima sinyal dari client (Node MCU), Server bertindak cepat dengan melakukan tiga fungsi utama yaitu menyimpan data insiden secara aman ke dalam database, mengaktifkan buzzer atau alarm lokal sebagai peringatan dini, dan mengirimkan notifikasi push ke aplikasi mobile yang digunakan oleh admin, pihak berwenang atau tim tanggap darurat. Keberlanjutan operasional server ini sangat penting. Oleh karena itu, Powerbank (uninterruptible power supply) diintegrasikan untuk memastikan pasokan daya yang konstan, sehingga sistem tetap aktif dan responsif meskipun terjadi gangguan atau pemadaman listrik.



Gambar 2 2 Diagram Block Server

2. Pengujian Perangkat Lunak (*Software Testing*)

Pengujian Perangkat Lunak (*Software Testing*) adalah proses evaluasi sistematis yang krusial untuk memverifikasi dan memvalidasi apakah sebuah produk perangkat lunak berfungsi sesuai dengan persyaratan yang telah ditentukan, dan untuk mengidentifikasi adanya celah (*bug*) atau kekurangan [4]. Tujuannya melampaui sekadar menemukan kesalahan tetapi berfokus pada peningkatan kualitas, keandalan, dan kinerja aplikasi secara keseluruhan. *Software Testing* memastikan aplikasi mampu beroperasi dengan stabil dalam berbagai kondisi sehingga memenuhi ekspektasi pengguna dan mengurangi risiko kegagalan operasional saat digunakan [18].

Pengujian perangkat lunak dapat diklasifikasikan ke dalam beberapa dimensi untuk memberikan cakupan evaluasi yang komprehensif terhadap suatu system [12]. Secara umum, klasifikasi yang paling mendasar membaginya berdasarkan pengujian fungsional dan pengujian non-fungsional. Pertama, Pengujian Fungsional (*Functional Testing*) yang berfokus pada verifikasi apakah setiap fitur aplikasi bekerja sesuai dengan spesifikasi dan kebutuhan pengguna, Kedua, Pengujian Non-Fungsional (*Non-Functional Testing*) yang mengukur kualitas operasional sistem, termasuk aspek-aspek seperti kinerja, keamanan, kemudahan penggunaan, dan ketahanan (*robustness*). Selain itu, pengujian juga diklasifikasikan berdasarkan tahap implementasi (misalnya, *Unit Testing*, *Integration Testing*, *System Testing*, dan *Acceptance Testing*), memastikan kualitas aplikasi dievaluasi secara bertahap dari komponen terkecil hingga sistem secara keseluruhan [19].

3. Metode *Robustness Testing* (RT)

Robustness Testing adalah metode pengujian *black box* yang fundamental dan krusial yang berfokus pada evaluasi ketahanan dan stabilitas suatu sistem perangkat lunak ketika dihadapkan pada kondisi lingkungan yang tidak ideal atau masukan data yang tidak valid bahkan diluar batas kemampuan atau bahkan merusak (*malformed*) [7]. Tujuan utama pengujian ini adalah untuk memastikan bahwa aplikasi tidak *crash* atau gagal total melainkan mampu merespons *input* abnormal tersebut dengan cara yang anggun (*graceful*) dan

terkelola. Metode ini secara aktif mensimulasikan skenario terburuk seperti mencoba memasukkan karakter khusus yang tidak diharapkan ke dalam kolom angka atau menguji respons aplikasi saat koneksi jaringan terputus tiba-tiba, untuk mengidentifikasi titik kegagalan sistem [3].

Dalam konteks aplikasi *Panic Button*, *Robustness Testing* menjadi sangat penting karena sistem harus berfungsi dalam kondisi yang penuh tekanan dan tidak terduga. Pengujian ini berupaya memverifikasi fungsi penanganan kesalahan (*error handling*) sistem apakah aplikasi menampilkan pesan kesalahan yang jelas dan informatif, apakah aplikasi mampu kembali ke status normal setelah menerima *input* yang salah, dan apakah data sensitif (seperti lokasi) tetap terlindungi meskipun terjadi anomali *input*. Dengan memaksa aplikasi bekerja di luar batas spesifikasinya. Metode ini membantu menjamin bahwa sistem keamanan darurat tersebut akan tetap andal dan dapat diandalkan ketika pengguna paling membutuhkannya bahkan dalam situasi paling ekstrem.

4. Metode *Equivalence Partitioning (EP)*

Equivalence Partitioning (EP) adalah teknik desain kasus uji *black box* yang fundamental (segala sesuatu yang bersifat dasar atau pokok), dirancang untuk meningkatkan efisiensi dan cakupan pengujian dengan meminimalkan keadaan berlebih dari suatu hal (redundansi) [7]. Metode ini bekerja berdasarkan prinsip bahwa input data ke dalam sebuah aplikasi dapat dikelompokkan menjadi partisi atau kelas yang memiliki perilaku ekuivalen yang artinya jika satu kasus uji berhasil mendeteksi cacat dalam sebuah partisi, kemungkinan besar semua kasus uji lain dalam partisi yang sama juga akan menghasilkan hasil serupa. Oleh karena itu, *Equivalence Partitioning* membagi semua kondisi input yang mungkin menjadi dua kategori utama yaitu partisi valid (data yang diharapkan dan dapat diterima sistem) dan partisi tidak valid (data yang berada di luar batas atau format yang diharapkan) [2]. Prinsip kerja utama dari *Equivalence Partitioning* adalah mengasumsikan bahwa kondisi pemrosesan data oleh perangkat lunak akan identik untuk semua anggota yang berada dalam satu kelas masukan yang setara. Metode ini didasarkan pada premis statistik bahwa jika sebuah input yang diambil dari kelompok tertentu

(partisi) mengungkapkan adanya bug, maka semua input lain dari kelompok yang sama kemungkinan besar akan memicu bug yang sama. Sebaliknya, jika input tersebut berhasil, maka semua input lainnya juga akan berhasil [7].

Penerapan utama dari *Equivalence Partitioning* adalah mengurangi jumlah kasus uji secara drastis tanpa mengurangi efektivitasnya [13]. Sebagai contoh ketika menguji setiap kemungkinan input (misalnya, menguji seratus nomor telepon yang berbeda) penguji hanya perlu memilih satu atau dua nilai representatif dari setiap partisi yang telah ditentukan. Misalnya, untuk kolom yang menerima usia 18 hingga 65 tahun, partisi validnya adalah satu angka dalam rentang tersebut, dan partisi tidak validnya adalah angka di bawah 18 dan angka di atas 65. Dengan memfokuskan pengujian pada satu sampel dari setiap kelas *Equivalence Partitioning* memastikan bahwa pengujian menjadi lebih terorganisir, lebih cepat, dan lebih efektif dalam menemukan kesalahan yang berkaitan dengan pemrosesan data.

Selain contoh diatas penerapan *Equivalence Partitioning* dalam desain kasus uji dimulai dengan menganalisis setiap kolom input pada aplikasi untuk menetapkan kelas ekuivalen yang relevan [13]. Misalnya, pada aplikasi *Panic Button* jika kolom untuk nomor kontak darurat dispesifikasikan harus memiliki 10 hingga 13 digit maka penguji akan mendefinisikan tiga partisi utama yaitu satu partisi valid (misalnya, nomor 12 digit) dan dua partisi tidak valid (misalnya, nomor 9 digit sebagai *too short* dan nomor 14 digit sebagai *too long*). Kasus uji kemudian dirumuskan untuk mengambil satu nilai representatif dari masing-masing partisi tersebut. Pengujian ini bertujuan untuk menguji bahwa input valid diterima dan diproses, sementara input tidak valid ditolak dengan pesan kesalahan yang sesuai, sehingga memastikan validasi data pada antarmuka berfungsi efisien tanpa perlu menguji setiap kemungkinan kombinasi nomor telepon.

BAB III METODOLOGI PENELITIAN

A. Waktu Penelitian

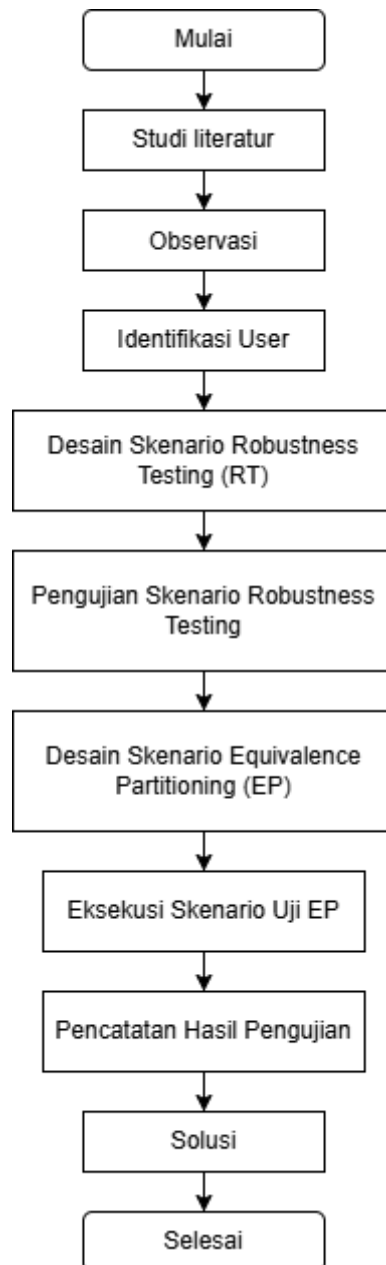
Penelitian ini dilaksanakan pada bulan September tahun 2025 sampai dengan bulan Agustus tahun 2026.

B. Tempat Penelitian

Penelitian ini dilakukan pada semester ganjil tahun ajaran 2025 / 2026 di Panti Lansia Dewanata Cilacap.

C. Prosedur Penelitian

Prosedur penelitian adalah tahapan sistematis yang diikuti oleh peneliti. Pada tahap penelitian ini merujuk pada Pengujian Aplikasi *Panic Button* Menggunakan metode *Robutness Testing* dan *Equivalence Partitioning*. Berikut tahapan penelitian yang dilakukan oleh peneliti dapat dilihat pada gambar berikut.



Gambar 3 1 Tahapan Penelitian

Berikut penjelasan tahapan prosedur penelitian :

1. Mulai

Awal dari semua tahapan adalah tahapan dimulai. Tahapan mulai adalah proses inisiasi atau mobilisasi yang mana sebagai momen sebuah ide atau rencana diubah menjadi kenyataan[17].

2. Studi Literatur

Tahapan ini memanfaatkan berbagai sumber tertulis seperti jurnal, skripsi, buku, dan lainnya yang berkaitan dengan penelitian. Dengan studi literatur, dapat mengetahui teori, konsep, dan metode yang akan digunakan dalam penelitian. Tujuannya untuk mendapatkan pemahaman yang lebih baik tentang topik-topik tersebut[2].

3. Observasi

Tahapan ini mencari sumber tentang spesifikasi aplikasi *Panic Button* yang nantinya digunakan sebagai panduan teknis yang merangkum seluruh fitur dan parameter sistem, di mana detail rinciannya diperoleh dan divalidasi melalui hasil wawancara mendalam dengan pembuat aplikasi *Panic Button*. Teknik observasi merupakan cara yang digunakan oleh peneliti untuk mendapatkan informasi terkait apa yang akan diteliti[9]. Peneliti melakukan kunjungan ke pengembang aplikasi untuk meminta izin mempelajari modul – modul dan melakukan pengujian terhadap aplikasi *panic button* serta menggali informasi tentang penjelasan aplikasi, rancangan, kebutuhan, fitur, tahapan pengembangan aplikasi, dan kendala perancangan kepada pengembang aplikasi.

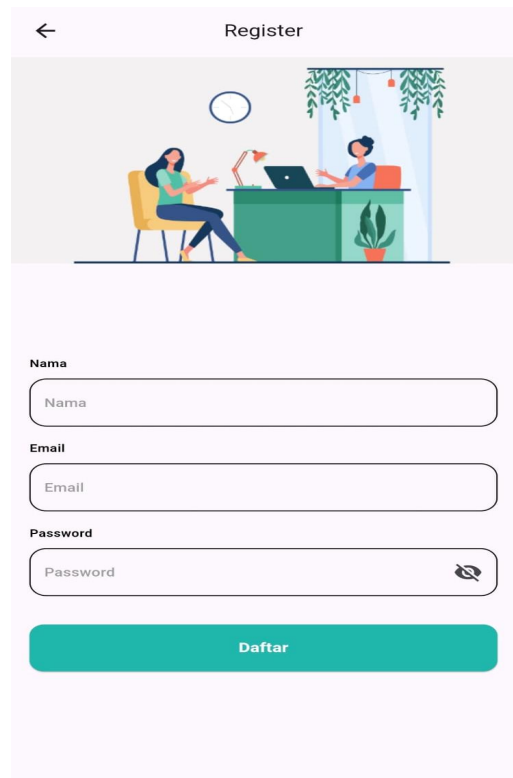
Dibawah ini merupakan beberapa tampilan dan Modul admin terdiri dari 6 modul pada aplikasi *panic button* berbasis android :

- a. Tampilan Awal Aplikasi *Panic button*
- b. Modul Daftar
- c. Modul Login
- d. Tampilan Home
- e. Tampilan About
- f. Modul Profil
- g. Modul Landasan Hukum
- h. Modul Visi Misi
- i. Modul Tahapan Kegiatan

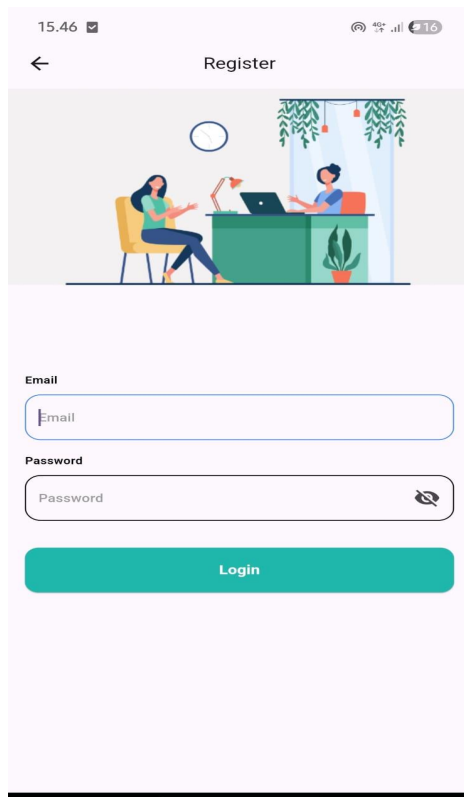
Dibawah ini merupakan tampilan dan modul pada aplikasi *panic button* yang bersumber dari developer, modul daftar dan modul login yang akan dilakukan pengujian, diantaranya adalah sebagai berikut :



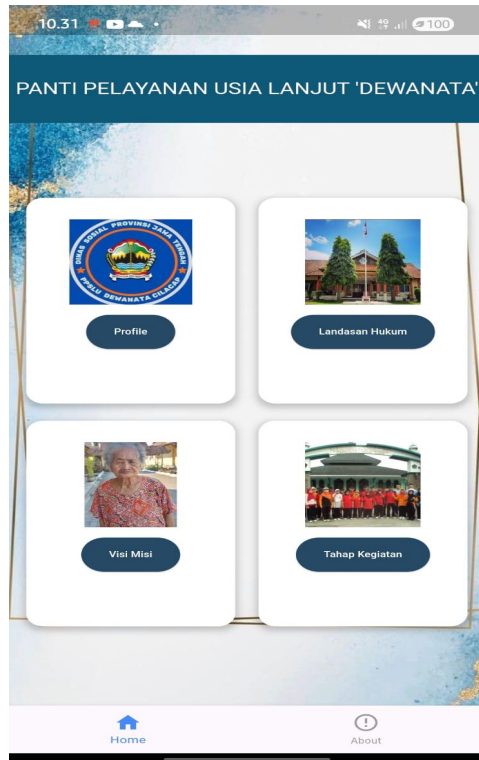
Gambar 3 2 Tampilan Awal



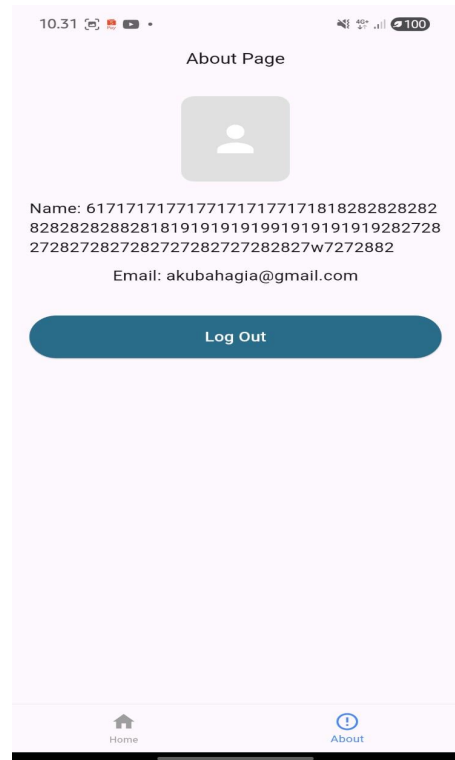
Gambar 3 3 Modul Daftar



Gambar 3 4 Modul Login



Gambar 3 5 Tampilan Home



Gambar 3 6 Tampilan About



Gambar 3 7 Modul Profil



Gambar 3 8 Modul Tahapan Kegiatan



Gambar 3 9 Modul Landasan Hukum



Gambar 3 10 Modul Visi Misi

4. Identifikasi User

Tahapan ini mengidentifikasi dan menggunakan fitur-fitur pada aplikasi *panic button* yang mana sebagai *user* agar nantinya dapat mendapatkan informasi dari pengujian tersebut (misalnya aktivasi tombol, pengiriman lokasi, validasi *input*). Identifikasi user bertujuan untuk mendapatkan informasi dan fungsi fitur aplikasi [2]. Pada tahapan Identifikasi User peneliti melakukan pendeskripsian mengenai pihak eksternal yang berinteraksi dengan aplikasi *panic button* dan fitur apa saja yang diberikan oleh pihak eksternal (user) tersebut yang dijelaskan dalam tabel 3.1 dibawah ini :

Tabel 3 1 Identifikasi User

No	User	Modul	Daftar Kebutuhan
1	Admin	Login	Admin dapat melakukan login
2	Admin	Home	Admin dapat melihat total Modul yang ada pada aplikasi <i>panic button</i>

3	Admin	Profil	Admin dapat melihat
4	Admin	Tahapan Kegiatan	Admin dapat melihat tentang profil panti pelayanan social dewanata cilacap.
5	Admin	Landasan Hukum	Admin dapat melihat tentang landasan hukum panti pelayanan social dewanata cilacap.
6	Admin	Visi Misi	Admin dapat melihat visi misi panti pelayanan social dewanata cilacap.
7	Admin	About	Admin dapat melihat akun yang login dan tombol log out akun

5. Desain Skenario *Robustness Testing* (RT)

Tahapan ini melakukan perancangan skenario yang memicu *input* di luar batas (*out-of-bounds*), atau gangguan ekstrem. Skenario *Robustness Testing* membantu dalam pemahaman yang mendalam tentang bagaimana sebuah sistem akan berperilaku dalam konteks penggunaan yang nyata. Penerapan desain skenario penerapan *Robustness Testing* pada sistem keuangan desa [3]. Pada tahapan perancangan *Robutness Testing* melakukan penjabaran batasan – batasan validasi check yang ada pada data input pada modul login dan daftar yang akan diuji coba. Batasan – batasan yang dijabarkan didapat dari pengembang aplikasi yang didapatkan saat observasi.

Tabel 3.2 menunjukan validasi check modul daftar yang bersumber dari developer yang bertujuan untuk mendeskripsikan batasan – batasan yang ada pada modul daftar yang memiliki tipe data input. Validasi ini berguna sebagai acuan sebelum melakukan pengujian nilai batas menggunakan metode *robustness testing*.

Tabel 3 2 Validasi Check Modul Daftar

No	Data Input	Deskripsi
A01	Nama	Nama tidak boleh kosong
A02	Nama	Nama minimal 1 dan maksimal 18
A03	Email	Email tidak boleh kosong
A04	Email	Username minimal 1 dan maksimal 18
A05	Email	Email dapat terdiri huruf, angka, dan spesial karakter
A06	Email	Email harus terdaftar di dalam data base
A07	Email	Email harus ada simbol “@” dan nama domain
A08	Email	Email tidak dapat mengandung spasi
A09	Password	Password tidak boleh kosong
A10	Password	Password minimal 6 dan maksimal 24
A11	Password	Password dapat terdiri angka, huruf, dan spesial karakter
A12	Password	Password harus ada di dalam data base
A13	Password	Password tidak dapat mengandung spasi

Tabel 3.3 menunjukkan validasi check modul login yang bersumber dari developer bertujuan untuk mendeskripsikan batasan – batasan yang ada pada modul login yang memiliki tipe data input. Validasi ini berguna sebagai acuan sebelum melakukan pengujian nilai batas menggunakan metode *robustness testing*.

Tabel 3 3 Validasi Check Modul Login

ID	Data Input	Deskripsi
B01	Email	Email tidak boleh kosong
B02	Email	Username minimal 1 dan maksimal 18 karakter
B03	Email	Email dapat terdiri huruf, angka, dan spesial karakter
B04	Email	Email harus terdaftar di dalam data base
B05	Email	Email harus ada symbol “@” dan nama domain
B06	Email	Email tidak dapat mengandung spasi
B07	Password	Password tidak boleh kosong
B08	Password	Password minimal 6 dan maksimal 24

B09	Password	Password dapat terdiri angka,huruf, dan spesial karakter
B10	Password	Password harus ada di dalam data base
B11	Password	Password tidak dapat mengandung spasi

Dibawah ini merupakan perancangan skenario *robustness testing* Modul login yang telah dibuat, diantaranya sebagai berikut [7] :

Tabel 3 4 Skenario Robustness Testing Modul login (Email)

ID	Input Email	Detail	Input Password
C01	Batas bawah valid	Persis batas minimal karakter	Valid
C02	Dibatas bawah batas valid	1 karakter kurang dari batas minimal	Valid
C03	Batas atas valid	Persis batas maksimal karakter	Valid
C04	Batas atas batas valid	1 karakter kurang dari batas maksimal	Valid
C05	Input ekstrem tidak valid	Nilai paling kecil yang mungkin	Valid
C06	Input ekstrem tidak valid	Nilai jauh melebihi batas maksimal	Valid
C07	Karakter tidak valid ekstrem	Mengandung banyak simbol tidak diizinkan	Valid
C08	Karakter spasi ekstrem	Banyak spasi diseluruh input	Valid

Tabel 3 5 Skenario Robustness Testing Modul Login (Password)

ID	Input Password	Detail	Input Email
D01	Batas bawah valid	Persis batas minimal karakter	Valid

D02	Dibatas bawah batas valid	1 karakter kurang dari batas minimal	Valid
D03	Batas atas valid	Persis batas maksimal karakter	Valid
D04	Batas atas batas valid	1 karakter kurang dari batas maksimal	Valid
D05	Input ekstrem tidak valid	Nilai paling kecil yang mungkin	Valid
D06	Input ekstrem tidak valid	Nilai jauh melebihi batas maksimal	Valid
D07	Kurang beberapa syarat valid	Tanpa huruf besar dan karakter khusus	Valid
D08	Karakter khusus tidak diizinkan	Simbol tidak termasuk dalam aturan	Valid

Dibawah ini merupakan perancangan skenario *robustness testing* Modul Daftar (Nama) yang telah dibuat, diantaranya sebagai berikut [7]:

Tabel 3 6 Skenario Robustness Testing Modul Daftar (Nama)

ID	Input Nama	Detail	Input Password	Input Email
E01	Batas bawah valid	Mengosongkan nama	Valid	Valid
E02	Batas atas valid	Persis batas maksimal karakter	Valid	Valid
E03	Batas atas batas valid	1 karakter kurang dari batas maksimal	Valid	Valid
E04	Karakter tidak valid	Mengandung banyak simbol	Valid	Valid

E05	Karakter spasi ekstrem	Banyak spasi diseluruh input	Valid	Valid
------------	------------------------------	---------------------------------	-------	-------

Dibawah ini merupakan perancangan skenario *robustness testing* Modul Daftar (Email) yang telah dibuat, diantaranya sebagai berikut [7]:

Tabel 3 7 Skenario Robustness Testing Modul Daftar (Email)

ID	Input Email	Detail	Input Nama	Input Password
F01	Batas bawah valid	Persis batas minimal karakter	Valid	Valid
F02	Dibatas bawah batas valid	1 karakter kurang dari batas minimal	Valid	Valid
F03	Atas bawah valid	Persis batas maksimal karakter	Valid	Valid
F04	Dibatas awah batas valid	1 karakter kurang dari batas maksimal	Valid	Valid
F05	Input ekstrem tidak valid	Nilai paling kecil yang mungkin	Valid	Valid
F06	Input ekstrem tidak valid	Nilai jauh melebihi batas maksimal	Valid	Valid
F07	Karakter tidak valid ekstrem	Mengandung banyak simbol tidak diizinkan	Valid	Valid
F08	Karakter spasi ekstrem	Banyak spasi diseluruh input	Valid	Valid

Dibawah ini merupakan perancangan skenario *robustness testing* Modul Daftar (Email) yang telah dibuat, diantaranya sebagai berikut [7]:

Tabel 3 8 Skenario Robustness Testing Modul Daftar (Password)

ID	Input Password	Detail	Input Nama	Input Password
G01	Batas bawah valid	Persis batas minimal karakter	Valid	Valid
G02	Dibatas bawah batas valid	1 karakter kurang dari batas minimal	Valid	Valid
G03	Atas bawah valid	Persis batas maksimal karakter	Valid	Valid
G04	Dibatas awah batas valid	1 karakter kurang dari batas maksimal	Valid	Valid
G05	Input ekstrem tidak valid	Nilai paling kecil yang mungkin	Valid	Valid
G06	Input ekstrem tidak valid	Nilai jauh melebihi batas maksimal	Valid	Valid
G07	Kurang beberapa syarat valid	Tanpa huruf besar dan karakter khusus	Valid	Valid
G08	Karakter khusus tidak diizinkan	Simbol tidak termasuk dalam aturan	Valid	Valid

6. Pengujian Skenario *Robustness Testing*

Pada tahapan pengujian *Robustness Testing* merupakan tahapan yang akan dilakukan pengujian langsung kedalam aplikasi *panic button*. Data uji yang telah disiapkan pada tahapan sebelumnya (desai skenario *Robustness Testing*) selanjutnya akan dieksekusi pada tahapan ini, Kemudian hasil yang telah didapatkan dan hasil yang diinginkan akan dicatat pada bagian ini sehingga akan mempermudah dalam melakukan analisa hasil pengujian. tahapan pertama dimulai dengan menyusun test case yang dibuat dan dokumentasikan hasil pengujian agar dapat dianalisa keefektifannya[7].

7. Desain Skenario *Equivalence Partitioning* (EP)

Tahapan ini melakukan pembuatan *testing cases* dengan membagi *input* menjadi partisi valid dan tidak valid untuk memaksimalkan cakupan *input* yang beragam secara efisien. Penerapan desain scenario metode *Equivalence Partitioning* pada Aplikasi Sistem Seleksi Pemenang Tender [18]. Tahap perancangan skenario *Equivalence Partitioning* melibatkan penyusunan narasi terstruktur yang merepresentasikan interaksi sistem dalam berbagai kondisi operasional. Proses perancangan ini mencakup identifikasi peran penguji serta formulasi langkah-langkah prosedural yang konkret untuk mencapai tujuan spesifik melalui fitur-fitur fungsional sistem. Dibawah ini merupakan perancangan skenario *Equivalence Partitioning* Modul login yang telah dibuat, diantaranya sebagai berikut :

Tabel 3 9 Skenario Equivalence Partitioning Modul login

Nama Skenario	:	Halaman Login
User	:	Admin
Tujuan	:	Masuk ke dalam aplikasi <i>panic button</i>
Kondisi awal	:	Menampilkan halaman login
Kondisi akhir	:	Masuk ke dashboard Home
ID Skenario		Tahapan Pengujian
H01 Skenario Utama 1	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan Email dan Password yang sudah terdaftar oleh sistem 3 User menekan tombol login
H02 Skenario Uji 1	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan Email yang sudah terdaftar dan input Password (< 6 karakter) 3 User menekan tombol login

H03 Skenario Uji 2	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email yang sudah terdaftar dan input Password (>18 Karakter)
		3	User menekan tombol login
H04 Skenario Uji 3	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email yang sudah terdaftar dan Password dikosongkan
		3	User menekan tombol login
H05 Skenario Uji 4	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email yang sudah terdaftar dan input Password tanpa angka dan simbol
		3	User menekan tombol login
H06 Skenario Uji 5	:	1	User diminta sistem untuk memasukan email dan password
		2	User mengosongkan Email dan Password
		3	User menekan tombol login
H07 Skenario Uji 6	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email tanpa simbol @ dan Password yang sudah terdaftar
		3	User menekan tombol login
H08 Skenario Uji 7	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email tanpa domain dan Password yang sudah terdaftar
		3	User menekan tombol login
H09 Skenario Uji 8	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan Email tanpa ekstensidan

			3	User menekan tombol login
H10	Skenario Uji 9	:	1	User diminta sistem untuk memasukan email dan password
			2	User memasukan Password yang sudah terdaftar dan mengosongkan Email
			3	User menekan tombol login

Tabel 3.10 menjelaskan mengenai tahapan – tahapan yang telah dirancang pada skenario *Equivalence Partitioning* modul daftar. Berdasarkan rancangan skenario modul daftar yang telah dibuat terdapat 1 skenario utama dan 51 skenario uji yang akan dilakukan terhadap modul daftar.

Tabel 3 10 Skenario Equivalence Partitioning Modul Daftar

Nama Skenario	:	Halaman Daftar
User	:	Admin
Tujuan	:	Masuk ke dalam aplikasi <i>panic button</i>
Kondisi awal	:	Menampilkan halaman Daftar
Kondisi akhir	:	Masuk ke dashboard Home

ID	Skenario		Tahapan Pengujian
I01	Skenario Utama 1	:	1 User diminta sistem untuk memasukan email dan password
			2 User memasukan Email, Password, dan Nama yang tidak terdaftar oleh sistem
			3 User menekan tombol daftar
I02	Skenario Uji 1	:	1 User diminta sistem untuk memasukan email dan password
			2 User memasukan
			- Email tanpa symbol @
			- Password yang tidak terdaftar
			- Nama yang tidak terdaftar

		3	User menekan tombol daftar
I03 Skenario Uji 2	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Email tanpa symbol @ - Password yang terdaftar - Nama yang tidak terdaftar
		3	User menekan tombol daftar
I04 Skenario Uji 3	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Email tanpa symbol @ - Password yang tidak terdaftar - Nama yang terdaftar
		3	User menekan tombol daftar
I05 Skenario Uji 4	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Email tanpa symbol @ - Password yang terdaftar - Nama yang terdaftar
		3	User menekan tombol daftar
I06 Skenario Uji 5	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Email tanpa domain - Password yang tidak terdaftar - Nama yang tidak terdaftar
		3	User menekan tombol daftar
I07 Skenario Uji 6	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan

			- Email tanpa domain - Password yang terdaftar - Nama yang tidak terdaftar
		3	User menekan tombol daftar
I08	Skenario Uji 7	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan - Email tanpa domain - Password yang tidak terdaftar - Nama yang terdaftar 3 User menekan tombol daftar
I09	Skenario Uji 8	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan - Email tanpa domain - Password yang terdaftar - Nama yang terdaftar 3 User menekan tombol daftar
I10	Skenario Uji 9	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan - Email tanpa ekstensi - Password yang tidak terdaftar - Nama yang tidak terdaftar 3 User menekan tombol daftar
I11	Skenario Uji 10	:	1 User diminta sistem untuk memasukan email dan password 2 User memasukan - Email tanpa ekstensi - Password yang terdaftar - Nama yang tidak terdaftar 3 User menekan tombol daftar

- I12 Skenario Uji 11** : 1 User diminta sistem untuk memasukan email dan password
2 User memasukan
- Email tanpa ekstensi
- Password yang tidak terdaftar
- Nama yang terdaftar
3 User menekan tombol daftar
-
- I13 Skenario Uji 12** : 1 User diminta sistem untuk memasukan email dan password
2 User memasukan
- Email tanpa ekstensi
- Password yang terdaftar
- Nama yang terdaftar
3 User menekan tombol daftar
-
- I14 Skenario Uji 13** : 1 User diminta sistem untuk memasukan email dan password
2 User memasukan
- Email yang sudah terdaftar
- Password yang tidak terdaftar
- Nama yang tidak terdaftar
3 User menekan tombol daftar
-
- I15 Skenario Uji 14** : 1 User diminta sistem untuk memasukan email dan password
2 User memasukan
- Email yang sudah terdaftar
- Password yang terdaftar
- Nama yang tidak terdaftar
3 User menekan tombol daftar
-
- I16 Skenario Uji 15** : 1 User diminta sistem untuk memasukan email dan password
2 User memasukan
- Email yang sudah terdaftar

			- Password yang tidak terdaftar - Nama yang terdaftar
		3	User menekan tombol daftar
I17 Skenario Uji 16	:	1	User diminta sistem untuk memasukan email dan password
		2	User melakukan - Input Email yang terdaftar - Input Password yang terdaftar - Input Nama yang terdaftar
		3	User menekan tombol daftar
I18 Skenario Uji 17	:	1	User diminta sistem untuk memasukan email dan password
		2	User melakukan - Mengosongkan Email - Input Password yang tidak terdaftar - Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I19 Skenario Uji 18	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Mengosongkan Email - Password yang terdaftar - Nama yang tidak terdaftar
		3	User menekan tombol daftar
I20 Skenario Uji 19	:	1	User diminta sistem untuk memasukan email dan password
		2	User memasukan - Mengosongkan Email - Password yang tidak terdaftar - Nama yang terdaftar
		3	User menekan tombol daftar
I21 Skenario Uji 20	:	1	User diminta sistem untuk memasukan email

			dan password
		2	User memasukan
			- Mengosongkan Email - Password yang terdaftar - Nama yang terdaftar
		3	User menekan tombol daftar
I22	Skenario Uji 21	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
			- Input Email yang tidak terdaftar - Input Password terlalu pendek (< 6 Karakter) - Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I23	Skenario Uji 22	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
			- Input Email yang terdaftar - Input Password terlalu pendek (< 6 Karakter) - Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I24	Skenario Uji 23	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
			- Input Email yang tidak terdaftar - Input Password terlalu pendek (< 6 Karakter) - Input Nama yang terdaftar
		3	User menekan tombol daftar
I25	Skenario Uji 24	:	1 User diminta sistem untuk memasukan email dan password

		2	User melakukan
		-	Input Email yang terdaftar
		-	Input Password terlalu pendek (< 6 Karakter)
		-	Input Nama yang terdaftar
		3	User menekan tombol daftar
I26	Skenario Uji 25	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang tidak terdaftar
		-	Input Password terlalu Panjang (>24 Karakter)
		-	Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I27	Skenario Uji 26	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang terdaftar
		-	Input Password terlalu Panjang (>24 Karakter)
		-	Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I28	Skenario Uji 27	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang tidak terdaftar
		-	Input Password terlalu Panjang (>24 Karakter)
		-	Input Nama yang terdaftar
		3	User menekan tombol daftar
I29	Skenario Uji 28	:	1 User diminta sistem untuk memasukan email dan password

		2	User melakukan
		-	Input Email yang terdaftar
		-	Input Password terlalu Panjang (>24 Karakter)
		-	Input Nama yang terdaftar
		3	User menekan tombol daftar
I30	Skenario Uji 29	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang tidak terdaftar
		-	Input Password hanya angka/huruf saja
		-	Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I31	Skenario Uji 30	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang terdaftar
		-	Input Password hanya angka/huruf saja
		-	Input Nama yang tidak terdaftar
		3	User menekan tombol daftar
I32	Skenario Uji 31	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang tidak terdaftar
		-	Input Password hanya angka/huruf saja
		-	Input Nama yang terdaftar
		3	User menekan tombol daftar
I33	Skenario Uji 32	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang terdaftar
		-	Input Password hanya angka/huruf saja

			- Input Nama yang terdaftar
		3	User menekan tombol daftar
I34	Skenario Uji 33	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Mengosongkan Password - Input Nama yang tidak terdaftar 3 User menekan tombol daftar
I35	Skenario Uji 34	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang terdaftar - Mengosongkan Password - Input Nama yang tidak terdaftar 3 User menekan tombol daftar
I36	Skenario Uji 35	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Mengosongkan Password - Input Nama yang terdaftar 3 User menekan tombol daftar
I37	Skenario Uji 36	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang terdaftar - Mengosongkan Password - Input Nama yang terdaftar 3 User menekan tombol daftar
I38	Skenario Uji 37	:	1 User diminta sistem untuk memasukan email dan password

		2	User
		-	Input Email yang tidak terdaftar
		-	Input Password yang tidak terdaftar
		-	Mengkosongkan Nama
		3	User menekan tombol daftar
I39	Skenario Uji 38	:	1 User diminta sistem untuk memasukan email dan password
		2	User
		-	Input Email yang terdaftar
		-	Input Password yang tidak terdaftar
		-	Mengkosongkan Nama
		3	User menekan tombol daftar
I40	Skenario Uji 39	:	1 User diminta sistem untuk memasukan email dan password
		2	User
		-	Input Email yang tidak terdaftar
		-	Input Password yang terdaftar
		-	Mengkosongkan Nama
		3	User menekan tombol daftar
I41	Skenario Uji 40	:	1 User diminta sistem untuk memasukan email dan password
		2	User
		-	Input Email yang terdaftar
		-	Input Password yang terdaftar
		-	Mengkosongkan Nama
		3	User menekan tombol daftar
I42	Skenario Uji 41	:	1 User diminta sistem untuk memasukan email dan password
		2	User melakukan
		-	Input Email yang tidak terdaftar
		-	Input Password yang tidak terdaftar
		-	Input Nama hanya simbol/angka

		3	User menekan tombol daftar
I43	Skenario Uji 42	:	1 User diminta sistem untuk memasukan email dan password 2 User - Input Email yang terdaftar - Input Password yang tidak terdaftar - Input Nama hanya simbol/angka 3 User menekan tombol daftar
I44	Skenario Uji 43	:	1 User diminta sistem untuk memasukan email dan password 2 User - Input Email yang tidak terdaftar - Input Password yang terdaftar - Input Nama hanya simbol/angka 3 User menekan tombol daftar
I45	Skenario Uji 44	:	1 User diminta sistem untuk memasukan email dan password 2 User - Input Email yang terdaftar - Input Password yang terdaftar - Input Nama hanya simbol/angka 3 User menekan tombol daftar
I46	Skenario Uji 45	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Input Password yang tidak terdaftar - Input Nama sangat Panjang(>18 Karakter) 3 User menekan tombol daftar
I47	Skenario Uji 46	:	1 User diminta sistem untuk memasukan email dan password 2 User

			- Input Email yang tidak terdaftar - Input Password yang terdaftar - Input Nama sangat Panjang(>18 Karakter)
		3	User menekan tombol daftar
I48	Skenario Uji 47	:	1 User diminta sistem untuk memasukan email dan password 2 User - Input Email yang terdaftar - Input Password yang tidak terdaftar - Input Nama sangat Panjang(>18 Karakter) 3 User menekan tombol daftar
I49	Skenario Uji 48	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang terdaftar - Input Password yang terdaftar - Input Nama sangat Panjang(>18 Karakter) 3 User menekan tombol daftar
I50	Skenario Uji 49	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Input Password yang tidak terdaftar - Input Nama yang terdaftar 3 User menekan tombol daftar
I51	Skenario Uji 50	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Input Password yang terdaftar - Input Nama yang terdaftar 3 User menekan tombol daftar

I52 Skenario Uji 51	:	1 User diminta sistem untuk memasukan email dan password 2 User melakukan - Input Email yang tidak terdaftar - Input Password yang terdaftar - Input Nama yang tidak terdaftar 3 User menekan tombol daftar
----------------------------	---	--

8. Eksekusi Skenario Uji EP

Tahapan ini menjalankan pengujian fungsionalitas dasar menggunakan data *representatif* dari setiap partisi (valid dan tidak valid). Pada tahapan pengujian *Equivalence Partitioning* merupakan tahapan yang akan dilakukan pengujian langsung kedalam aplikasi *panic button*. Data uji yang telah disiapkan pada tahapan sebelumnya (desai skenario *Equivalence Partitioning*) selanjutnya akan dieksekusi pada tahapan ini, Kemudian hasil yang telah didapatkan dan hasil yang diinginkan akan dicatat pada bagian ini sehingga akan mempermudah dalam melakukan analisa hasil pengujian. tahapan pertama dimulai dengan menyusun test case yang dibuat dan dokumentasikan hasil pengujian agar dapat dianalisa keefektifannya[18]

9. Pencatatan Hasil Pengujian

Tahapan ini melakukan pencatatan dan mendokumentasikan hasil pengujian dari metode *Robutness Testing* dan *Equivalence Partitioning* untuk menarik kesimpulan tentang kualitas dan keandalan aplikasi *panic button*. Sebelum mengetahui kualitas dari sebuah aplikasi maka perlu dilakukan penghitungan presentase dari total kasus uji (Test Case) yang sudah dilakukan pada tahapan pengujian *Robutness Testing* dan *Equivalence Partitioning*. Penghitungan presentase Test Case menggunakan rumus dibawah ini[16].

$$Test\ case\ pass = \left(\frac{Test\ case\ passed}{Total\ test\ case} \right) \times 100\%$$

$$Test\ case\ failed = \left(\frac{Find\ test\ case\ failed}{Total\ test\ case} \right) \times 100\%$$

Berdasarkan hasil perhitungan presentase test case maka untuk penilaian kualitas (tingkat capaian efektivitas pengujian) dibagi menjadi 4 kategori. Adapun rincian kriteria rasio efektivitas dapat dilihat pada tabel dibawah [2].

Tabel 3 11 Standar acuan efektivitas

No	Tingkat Capaian	Rasio Efektivitas (%)
1	Sangat efektif	$\geq 80,00$
2	Cukup efektif	60,00 – 79,99
3	Tidak efektif	40,00 – 59,99
4	Sangat tidak efektif	$< 40,00$

10. Solusi

Hasil pencatatan pengujian *Robustness Testing* dan *Equivalence Partitioning* yang diperoleh akan dianalisis agar bisa mendapatkan solusi untuk pengembangan aplikasi *panic button*. Tim pengembang akan diberikan informasi hasil pengujiannya dan solusi untuk pengembangan aplikasi *panic button* untuk memperkuat stabilitas aplikasi sehingga sesuai dengan yang diinginkan.

11. Selesai

Tahapan ini adalah berakhirnya penelitian dan telah selesai.

D. Jadwal Penelitian

Tabel 3 12 Jadwal Penelitian

No	Kegiatan	2025																2026											
		September				Oktober				November				Desember				Januari				Agustus							
		Minggu ke-																											
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4				
1. Studi Literatur																													
2. Observasi																													
3. Desain Skenario EP & RT																													
4. Persiapan Pengujian																													
5. Pengujian EP & RT																													
6. Analisa Pengujian																													
7. Evaluasi																													
8. Penulisan Proposal Skripsi																													
9. Penulisan Skripsi																													
10. Publikasi Ilmiah																													

BAB IV PEMBAHASAN

A. Hasil Pengujian Metode *Robustness Testing*

Hasil pengujian perangkat lunak menggunakan metode *robustness testing* pada Modul Daftar disajikan secara sistematis untuk menunjukkan ketahanan sistem terhadap berbagai variasi input. Pengujian ini berfokus pada validasi perilaku modul saat menerima data di luar batas normal guna memastikan stabilitas performa. Rincian mengenai parameter uji, skenario, serta luaran dari proses tersebut dirangkum dalam Tabel Hasil Pengujian Metode *Robustness Testing* Modul Daftar di bawah ini.

Tabel 4 1 Hasil Pengujian Metode Robustness Testing Modul Daftar (Nama)

ID Test case	Kriteria Nama	Nilai Input Nama	Pass / Failed	Hasil yang diharapkan
E01	Batas bawah valid	U (1 karakter)	Pass	Input diterima
E02	Batas atas valid	“ “ (0 karakter)	Failed	Registration failed: “Name must be fiiled in”
E03	Batas atas batas valid	UserEmailTesting12 (18 karakter)	Pass	Input diterima
E04	Karakter tidak valid	UserEmailTesting123 (19 karakter)	Failed	Registration failed: “Name must be less than 18 character”
E05	Karakter khusus dan spasi ekstrem	“ % -#“	Pass	Input diterima

Paparan berikut menyajikan hasil pengujianm modul daftar, khususnya pada parameter input email, dengan menerapkan metode *robustness testing*. Pengujian ini bertujuan untuk mengukur sejauh mana sistem mampu menangani variasi data input, baik yang bersifat valid maupun ekstrem, guna menjamin integritas data dan stabilitas aplikasi. Detail mengenai skenario uji, respons sistem, serta status validasi dari pengujian tersebut dirangkum secara komprehensif dalam tabel di bawah ini.

Tabel 4 2 Hasil Pengujian Metode Robustness Testing Modul Daftar (Email)

ID Test case	Kriteria Email	Nilai Input Email	Pass / Failed	Hasil yang diharapkan
F01	Batas bawah valid	1@gmail.com (1 karakter username)	Pass	Input diterima
F02	Dibatas bawah batas valid	@gmail.com (0 karakter username)	Pass	Registration failed: “username must be fiiled in”
F03	Atas bawah valid	UserEmailTesting12@gmail.com (18 karakter username)	Pass	Input diterima
F04	Dibatas awah batas valid	UserEmailTesting123@gmail.com (19 karakter username)	Failed	Registration failed: “User name must be less than 18 character”
F05	Input ekstrem tidak valid	“ “@gmail.com	Pass	Registration failed: “username must be fiiled in”

F06	Input ekstrem tidak valid	Userrrrrrr 12345678 @gmail.com (20 karakter username dan spasi)	<i>Failed</i>	Registration failed: "Name must be less than 18 character"
F07	Karakter tidak valid ekstrem	User @#\$%\$#%~* @gmail.com	<i>Pass</i>	Registration failed: "The email address is badly formatted"
F08	Karakter spasi ekstrem	U s e r 1 2 3 @gmail.com	<i>Pass</i>	Registration failed: "The email address is badly formatted"

Berikut hasil dari pengujian Metode *Robustness Testing* Modul Daftar (Password) disajikan pada tabel berikut.

Tabel 4 3 Hasil Pengujian Metode Robustness Testing Modul Daftar (Password)

ID Test case	Kriteria Password	Nilai Input Password	<i>Pass / Failed</i>	Hasil yang diharapkan
G01	Batas bawah valid	Us1234 (6 karakter)	<i>Pass</i>	Input diterima
G02	Dibatas bawah batas valid	Us123 (5 karakter)	<i>Pass</i>	Registration failed: "password should be at least 6 character"
G03	Atas bawah valid	UserEmailTesting12 (18 karakter)	<i>Pass</i>	Input diterima
G04	Dibatas awah	UserEmailTesting123 (19 karakter)	<i>Failed</i>	Registration failed: "User

	batas valid			name must be less than 18 character”
G05	Input ekstrem tidak valid	“ “ (kosong)	<i>Pass</i>	Registration failed: “Password must be fiiled in”
G06	Input ekstrem	Userrrrrrrr 12 - 345678 (20 karakter)	<i>Failed</i>	Registration failed: “User name must be less than 18 character”
G07	Kurang beberapa syarat valid	\$#%~*	<i>Pass</i>	Registration failed: “password should be at least 6 character”
G08	Karakter khusus	U % # * 1) 3	<i>Pass</i>	Input diterima

Berikut hasil dari pengujian Metode *Robustness Testing* Modul Login (Email) disajikan pada tabel berikut.

Tabel 4 4 Hasil Pengujian Metode Robustness Testing Modul login (Email)

ID Test case	Kriteria Email	Nilai Input Email	<i>Pass / Failed</i>	Hasil yang diharapkan
C01	Batas bawah valid	1@gmail.com (1 karakter)	<i>Pass</i>	Input diterima
C02	Dibatas bawah batas valid	@gmail.com (0 karakter)	<i>Pass</i>	Registration failed: “Username must be fiiled in”

C03	Batas atas valid	UserEmailTesting12@gmail.com (18 karakter)	<i>Pass</i>	Input diterima
C04	Batas atas batas valid	UserEmailTesting123@gmail.com (19 karakter)	<i>Failed</i>	Registration failed: “Username must be less than 18 character”
C05	Input ekstrem tidak valid	“ “@gmail.com (kosong)	<i>Pass</i>	Registration failed: “username must be filed in”
C06	Input ekstrem tidak valid	Userrrrrrr\$123456 -@gmail.com	<i>Pass</i>	Registration failed: “The email address is badly formatted”
C07	Karakter tidak valid ekstrem	User@#\$\$%\$#%~*	<i>Pass</i>	Registration failed: “The email address is badly formatted”
C08	Karakter spasi ekstrem	U s e r 1 2 3	<i>Pass</i>	Registration failed: “The email address is badly formatted”

Berikut hasil dari pengujian Metode *Robustness Testing* Modul Login (Password) disajikan pada tabel berikut.

Tabel 4 5 Hasil Pengujian Metode *Robustness Testing* Modul Login (Password)

ID Test case	Kriteria Password	Nilai Input Password	<i>Pass / Failed</i>	Hasil yang diharapkan
---------------------	--------------------------	-----------------------------	-----------------------------	------------------------------

D01	Input Password	Us1234 (6 karakter)	<i>Pass</i>	Input diterima
D02	Batas bawah valid	Us (2 karakter)	<i>Pass</i>	Registration failed: “password should be at least 6 character”
D03	Dibatas bawah batas valid	UserEmailTesting12 (18 karakter)	<i>Pass</i>	Input diterima
D04	Batas atas valid	UserEmailTesting123 (19 karakter)		Registration failed: “Username must be less than 18 character”
D05	Batas atas batas valid	“ “ (kosong)	<i>Pass</i>	Registration failed: “password should be at least 6 character”
D06	Input ekstrem tidak valid	Use (rrrr&rr) 12345678	<i>Failed</i>	Registration failed: “password must not contain space character”
D07	Input ekstrem tidak valid	User @#\$\$%\$#%~*	<i>Failed</i>	Registration failed: “password must not contain

				space character”
D08	Kurang beberapa syarat valid	U s e r 1 2 3	<i>Failed</i>	Registration failed: “password must not contain space character”

B. Hasil Pengujian Metode *Equivalence Partitioning*

Pemaparan mengenai hasil pengujian fungsionalitas pada Modul Login dengan mengimplementasikan metode *Equivalence Partitioning*. Pengujian ini difokuskan pada pengelompokan kondisi input ke dalam beberapa kelas equivalensi untuk memvalidasi respons sistem terhadap data valid dan tidak valid secara efisien. Melalui pendekatan ini, efektivitas sistem dalam mengelola hak akses pengguna dapat diukur secara komprehensif, sebagaimana yang didokumentasikan dalam Tabel berikut.

Tabel 4 6 Hasil Pengujian Metode *Equivalence Partitioning* Modul login

ID Test case	Nilai Input	Hasil yang diinginkan	Hasil yang didapat	Pass / Failed
H01	Email : User1@gmail.com Password : User123	Input diterima	Input diterima	<i>Pass</i>
H02	Email : User1@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
H03	Email : User1@gmail.com Password : User11111111111111	Registration failed	Input diterima	<i>Failed</i>
H04	Email : User1@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
H05	Email : User1@gmail.com Password : @#34@#4	Input diterima	Input diterima	
H06	Email : Password :	Registration failed	Registration failed	<i>Pass</i>
H07	Email : Usergmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
H08	Email : User@.com Password : User123	Registration failed	Registration failed	<i>Pass</i>

H09	Email : User@gmail Password : User123	Registration failed	Registration failed	<i>Pass</i>
H10	Email : Password : User123	Registration failed	Registration failed	<i>Pass</i>

Berikut hasil dari pengujian Metode *Robustness Testing* Modul daftar disajikan pada tabel berikut.

Tabel 4 7 Hasil Pengujian Metode *Equivalence Partitioning* Modul Daftar

ID	Nilai Input	Hasil yang diinginkan	Hasil yang didapat	Pass / Failed
I01	Nama : User1 Email : User1@gmail.com Password : User123	Input diterima	Input diterima	<i>Pass</i>
I02	Nama : User2 Email : User1@gmail.com Password : User12	Registration failed	Registration failed	<i>Pass</i>
I03	Nama : User2 Email : User1@gmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
I04	Nama : User1 Email : User1@gmail.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I05	Nama : User1 Email : User1@gmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
I06	Nama : User2 Email : User1@.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I07	Nama : User2 Email : User1@.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I08	Nama : User2 Email : User1@.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I09	Nama : User1 Email : User1@.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
I10	Nama : User2 Email : User1@gmail Password : User34	Registration failed	Registration failed	<i>Pass</i>
I11	Nama : User2 Email : User1@gmail Password : User123	Registration failed	Registration failed	<i>Pass</i>

I12	Nama : User1 Email : User1@gmail Password : User34	Registration failed	Registration failed	<i>Pass</i>
I13	Nama : User1 Email : User1@gmail Password : User123	Registration failed	Registration failed	<i>Pass</i>
I14	Nama : User2 Email : User1@gmail.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I15	Nama : User2 Email : User1@gmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
I16	Nama : User1 Email : User1@gmail.com Password : User34	Registration failed	Registration failed	<i>Pass</i>
I17	Nama : User1 Email : User1@gmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>
I18	Nama : User2 Email : Password : User34	Registration failed	Registration failed	<i>Pass</i>
I19	Nama : User2 Email : Password : User123	Registration failed	Registration failed	<i>Pass</i>
I20	Nama : User1 Email : Password : User34	Registration failed	Registration failed	<i>Pass</i>
I21	Nama : User1 Email : Password : User123	Registration failed	Registration failed	<i>Pass</i>
I22	Nama : User3 Email : User2@gmail.com Password : User3	Registration failed	Registration failed	<i>Pass</i>
I23	Nama : User2 Email : User2@gmail.com Password : User3	Registration failed	Registration failed	<i>Pass</i>
I24	Nama : User2 Email : User2@gmail.com Password : User3	Registration failed	Registration failed	<i>Pass</i>
I25	Nama : User2 Email : User2@gmail.com Password : User12345678 9us123456789	Input diterima	Input diterima	<i>Pass</i>
I26	Nama : User2 Email : User2@gmail.com Password : User12345678 9us123456789	Input diterima	Input diterima	<i>Pass</i>

I27	Nama : User3 Email : User3@gmail.com Password : User12345678 9us123456789	Registration failed	Input diterima	<i>Failed</i>
I28	Nama : User2 Email : User2@gmail.com Password : User12345678 9us123456789	Registration failed	Registration failed	<i>Pass</i>
I29	Nama : User2 Email : User3@gmail.com Password : User12345678 9us123456789	Registration failed	Registration failed	<i>Pass</i>
I30	Nama : User4 Email : User4@gmail.com Password : 123456	Input diterima	Input diterima	<i>Pass</i>
I31	Nama : User5 Email : User4@gmail.com Password : 123456	Registration failed	Registration failed	<i>Pass</i>
I32	Nama : User4 Email : User5@gmail.com Password : 123456	Input diterima	Input diterima	<i>Pass</i>
I33	Nama : User4 Email : User5@gmail.com Password : 123456	Registration failed	Registration failed	<i>Pass</i>
I34	Nama : User5 Email : User5@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
I35	Nama : User5 Email : User4@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
I36	Nama : User4 Email : User5@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
I37	Nama : User4 Email : User4@gmail.com Password :	Registration failed	Registration failed	<i>Pass</i>
I38	Nama : Email : User6@gmail.com Password : User34	Input diterima	Input diterima	<i>Pass</i>
I39	Nama : Email : User6@gmail.com Password : User56	Registration failed	Registration failed	<i>Pass</i>
I40	Nama : Email : User7@gmail.com Password : User34	Input diterima	Input diterima	<i>Pass</i>
I41	Nama : Email : User7@gmail.com Password : User123	Registration failed	Registration failed	<i>Pass</i>

I42	Nama : 123 Email : User8@gmail.com Password : User56	Input diterima	Input diterima	<i>Pass</i>
I43	Nama :123 Email : User8@gmail.com Password : User78	Registration failed	Registration failed	<i>Pass</i>
I44	Nama :123 Email : User9@gmail.com Password : User56	Input diterima	Input diterima	<i>Pass</i>
I45	Nama :123 Email : User9@gmail.com Password : User56	Registration failed	Registration failed	<i>Pass</i>
I46	Nama : User123456789us 123 Email : User9@gmail.com Password : User78	Input diterima	Input diterima	<i>Pass</i>
I47	Nama : User123456789us 123 Email : User10@gmail.com Password : User78	Input diterima	Input diterima	<i>Pass</i>
I48	Nama : User123456789us 123 Email : <u>User10@gmail.com</u> Password : User10	Registration failed	Registration failed	<i>Pass</i>
I49	Nama : User123456789us 123 Email : <u>User10@gmail.com</u> Password : User78	Registration failed	Registration failed	<i>Pass</i>
I50	Nama : User4 Email : <u>User11@gmail.com</u> Password : User10	Input diterima	Input diterima	<i>Pass</i>
I51	Nama : User4 Email : <u>User12@gmail.com</u> Password : User78	Input diterima	Input diterima	<i>Pass</i>
I52	Nama : User5 Email : <u>User13@gmail.com</u> Password : User11	Input diterima	Input diterima	<i>Pass</i>

C. Pencatatan Analisa Hasil Pengujian

Pada tahapan pencatatan hasil pengujian akan melakukan analisa hasil pengujian dari data yang telah didapatkan pada tahapan pengujian pengujian *Robustness Testing* dan *Equivalence Partitioning*. Kemudian dari hasil yang didapatkan penguji akan melakukan perhitungan kualitas

menggunakan rumus presentase nilai test case yang ada pada tabel 3.11.

Dibawah ini merupakan hasil analisa pengujian yang sudah dilakukan terhadap aplikasi *panic button* menggunakan metode pengujian *Robustness Testing* dan *Equivalence Partitioning*, diantaranya adalah sebagai berikut :

1. Pengujian *Robustness Testing*

- a. Tabel 4.8 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul daftar (Nama) menggunakan metode *Robustness Testing*.

$$Test\ case\ pass = \left(\frac{3}{5}\right) \times 100\% = 60\%$$

$$Test\ case\ failed = \left(\frac{2}{5}\right) \times 100\% = 40\%$$

Tabel 4 8 Hasil analisa modul daftar (Nama)

No	Modul	SubModul	Pass	Failed	Kategori
1	Daftar	Nama	60%	40%	Cukup Efektif

Berdasarkan hasil pengujian pada modul daftar (Nama) menggunakan metode *Robustness Testing*, telah menghasilkan 3 data uji yang lolos dan 2 data uji yang tidak lolos dari total 5 test case yang sudah diuji sehingga pada modul daftar (Nama) memiliki kategori kualitas yang cukup efektif.

- b. Tabel 4.9 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul daftar (Email) menggunakan metode *Robustness Testing*.

$$Test\ case\ pass = \left(\frac{6}{8}\right) \times 100\% = 75\%$$

$$Test\ case\ failed = \left(\frac{2}{8}\right) \times 100\% = 25\%$$

Tabel 4 9 Hasil analisa modul daftar (Email)

No	Modul	SubModul	Pass	Failed	Kategori
1	Daftar	Email	75%	25%	Cukup Efektif

Berdasarkan hasil pengujian pada modul daftar (Email) menggunakan metode *Robustness Testing*, telah menghasilkan 6 data uji yang lolos dan 2 data uji yang tidak lolos dari total 8 test case yang sudah diuji sehingga pada modul daftar (Email) memiliki kategori kualitas yang cukup efektif.

- c. Tabel 4.10 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul daftar (Password) menggunakan metode *Robustness Testing*.

$$Test\ case\ pass = \left(\frac{6}{8}\right) \times 100\% = 75\%$$

$$Test\ case\ failed = \left(\frac{2}{8}\right) \times 100\% = 25\%$$

Tabel 4 10 Hasil analisa modul daftar (Password)

No	Modul	SubModul	Pass	Failed	Kategori
1	Daftar	Password	75%	25%	Cukup Efektif

Berdasarkan hasil pengujian pada modul daftar (Password) menggunakan metode *Robustness Testing*, telah menghasilkan 6 data uji yang lolos dan 2 data uji yang tidak lolos dari total 8 test case yang sudah diuji sehingga pada modul daftar (Password) memiliki kategori kualitas yang cukup efektif.

- d. Tabel 4.11 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul login (Email) menggunakan metode *Robustness Testing*.

$$Test\ case\ pass = \left(\frac{7}{8}\right) \times 100\% = 82,5\%$$

$$Test\ case\ failed = \left(\frac{1}{8}\right) \times 100\% = 12,5\%$$

Tabel 4 11 Hasil analisa modul login (Email)

No	Modul	SubModul	Pass	Failed	Kategori
1	Login	Email	82,5%	12,5%	Sangat Efektif

Berdasarkan hasil pengujian pada modul login (Email) menggunakan metode *Robustness Testing*, telah menghasilkan 7 data uji yang lolos dan 1 data uji yang tidak lolos dari total 8 test case yang sudah diuji sehingga pada modul login (Email) memiliki kategori kualitas yang cukup efektif.

- e. Tabel 4.12 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul login (Password) menggunakan metode *Robustness Testing*.

$$Test\ case\ pass = \left(\frac{4}{8}\right) \times 100\% = 50\%$$

$$Test\ case\ failed = \left(\frac{4}{8}\right) \times 100\% = 50\%$$

Tabel 4 12 Hasil analisa modul login (Password)

No	Modul	SubModul	Pass	Failed	Kategori
1	Login	Password	50%	50%	Tidak Efektif

Berdasarkan hasil pengujian pada modul login (Password) menggunakan metode *Robustness Testing*, telah menghasilkan 4 data uji yang lolos dan 4 data uji yang tidak lolos dari total 8 test case yang sudah diuji sehingga pada modul login (Password) memiliki kategori kualitas yang tidak efektif.

2. Pengujian *Equivalence Partitioning*

- a. Tabel 4.13 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul login menggunakan metode *Equivalence*

Partitioning.

$$Test\ case\ pass = \left(\frac{9}{8}\right) \times 100\% = 90\%$$

$$Test\ case\ failed = \left(\frac{1}{8}\right) \times 100\% = 10\%$$

Tabel 4 13 Hasil analisa modul login

No	Modul	Pass	Failed	Kategori
1	Login	90%	10%	Sangat Efektif

Berdasarkan hasil pengujian pada modul login daftar (Password) menggunakan metode *Equivalence Partitioning*, telah menghasilkan 9 data uji yang lolos dan 1 data uji yang tidak lolos dari total 10 test case yang sudah diuji sehingga pada modul login memiliki kategori kualitas yang cukup efektif.

- b. Tabel 4.14 merupakan hasil analisa dari hasil pengujian yang sudah dilakukan kepada modul daftar menggunakan metode *Equivalence Partitioning*.

$$Test\ case\ pass = \left(\frac{51}{52}\right) \times 100\% = 98,07\%$$

$$Test\ case\ failed = \left(\frac{1}{52}\right) \times 100\% = 1,93\%$$

Tabel 4 14 Hasil analisa modul daftar

No	Modul	Pass	Failed	Kategori
1	Daftar	98,07%	1,93%	Sangat Efektif

Berdasarkan hasil pengujian pada modul daftar menggunakan metode *Equivalence Partitioning*, telah menghasilkan 51 data uji yang lolos dan 1 data uji yang tidak lolos dari total 52 test case yang sudah diuji sehingga pada modul daftar memiliki kategori kualitas yang sangat efektif.

BAB V PENUTUP

A. Kesimpulan

Bedasarkan pengujian yang telah dilakukan terhadap aplikasi *panic button* menggunakan metode *Robustness Testing* dan *Equivalence Partitioning* telah diujikan, maka dapat ditarik kesimpulan diantaranya seperti :

1. Metode *Robustness Testing* (RT) yang diterapkan untuk menentukan batas atas dan batas bawah nilai input guna mereduksi jumlah *test case* sehingga proses pengujian menjadi lebih efisien. Tahapan pengujian dimulai dengan penjabaran aturan (*rules*) pada data input bertipe *string* melalui mekanisme *validation check*. Selanjutnya, pengujian nilai batas dilakukan untuk memvalidasi kesesuaian implementasi sistem terhadap aturan yang telah ditetapkan.
2. Metode *Equivalence Partitioning* (EP) yang diterapkan untuk mengidentifikasi efek fungsional pada aplikasi *Panic Button*. Prosedur pengujian dimulai dengan identifikasi profil pengguna dan penyusunan skenario uji. Selanjutnya, dilakukan pemetaan kelas ekivalensi data input berdasarkan skenario tersebut, yang mencakup kombinasi data valid dan tidak valid menggunakan *test case matrix*. Tahap akhir melibatkan eksekusi *test case* untuk memvalidasi pemenuhan standar kelulusan uji pada aplikasi.
3. Pengujian yang sudah dilakukan terhadap aplikasi *panic button* menggunakan metode *Robustness Testing* dan *Equivalence Partitioning* telah mendapatkan hasil yang cukup memuaskan, hal itu dapat dibuktikan dari hasil yang telah didapatkan dan ditemukan 94 kasus uji yang lolos dan 13 kasus uji yang tidak lolos dari total 107 kasus uji yang sudah dilakukan terhadap aplikasi *panic button*. Adapun beberapa modul yang memiliki nilai presentase tertinggi diantaranya seperti pengujian metode *Equivalence Partitioning* pada modul login yaitu sebesar 98,07% dan adapun pengujian metode *Robustness Testing* pada modul daftar (password) yang memiliki nilai presentase terkecil yaitu sebesar 50%. Nilai keseluruhan kualitas aplikasi *panic button* yang

sudah diuji pada 7 sub modul dari 2 modul memiliki rata rata presentase sebesar 75,79% yang menandakan bahwa aplikasi kantin edu digital memiliki kualitas yang cukup efektif.

B. Saran

Bedasarkan hasil pengujian menggunakan metode *Robustness Testing* dan *Equivalence Partitioning* terhadap aplikasi *panic button*, penulis memberikan saran diantaranya sebagai berikut :

1. Diharapkan kepada pihak pengembang atau developer supaya bisa segera memperbaiki kesalahan atau error pada aplikasi *panic button* yang ditemukan terhadap modul login dan modul daftar.
2. Diharapkan kepada pihak pengembang atau developer supaya dapat menambah fitur scan sidik jari supaya setiap admin dapat dengan mudahnya untuk login dengan cepat.
3. Diharapkan kepada pihak pengembang aplikasi atau developer agar dapat meningkatkan sitem keamanan pada aplikasi *panic button* dikarenakan didalam aplikasi tersebut sangat berguna untuk kondisi darurat agar tidak disalah gunakan.

DAFTAR PUSTAKA

- [1] E. Novalia dan A. Voutama, “Black Box Testing dengan Teknik Equivalence Partitions Pada Aplikasi Android M-Magazine Mading Sekolah,” *J. Inform.*, vol. 11, no. 11, hal. 23–35, 2022.
- [2] F. Wardah Gracillaria Suharyono, Kartini, dan A. Junaidi, “Penerapan Metode Boundary Value Analysis Dan Equivalence Partitioning Dalam Pengujian Black Box Untuk Aplikasi Siadita,” *J. Mhs. Tek. Inform.*, vol. 8, no. 1, hal. 1013–1020, 2024.
- [3] F. Tangguh Admojo dan M. Leo Adi Saputra, “Analisis Sistem Keuangan Desa (SISKEUDes) di Kecamatan Muara Sugihan Menggunakan Metode Black Box Testing,” *J. Teknol. dan Inform.*, vol. 12, no. 2, hal. 141–152, 2022.
- [4] U. Hanifah, R. Alit, dan S. Sugiarto, “Penggunaan Metode Black Box Pada Pengujian Sistem Informasi Surat Keluar Masuk,” *J. Teknol. Inf. dan Komun.*, vol. 11, no. 2, hal. 33–40, 2016.
- [5] M. Anike, F. C, dan D. Melo, “Application Emergency Panic Button (Aepb) Berbasis Android (Studi Kasus Rs St. Carolus Boromeus-Bello),” *J. Sist. Inf.*, vol. 8, no. 3, hal. 367–376, 2019.
- [6] H. Sarifatin Munawaroh, S. Rosad, dan E. Sulistyanto, “Implementasi Tombol Panik SOS Untuk Lansia Berbasis IoT Pada Panti Pelayanan Sosial Lanjut Usia Dewanata Cilacap,” *J. Tecnoscienza*, vol. 9, no. 1, hal. 29–42, 2024.
- [7] N. Fahlevi Abdi dan S. R. Candra Nursari, “Pengujian Black Box pada Website dengan Metode Robustness Testing (Studi Kasus : Eiger Adventure),” *J. Informatics Adv. Comput.*, vol. 3, no. 2, hal. 93–96, 2022.
- [8] D. Jaya, “Uji Ketahanan (Robustness Test) Algoritma Steganografi Pada Aplikasi Media Sosial Berbasis Android,” *J. Konfrensi Nas. Sist. dan Inform.*, hal. 876–879, 2015.
- [9] R. Pramudita, “Pengujian Black Box pada Aplikasi Ecampus Menggunakan Metode Equivalence Partitioning,” *J. Informatics*, vol. 4, no. 2, hal. 193–202, 2020.
- [10] A. Ardilla, F. Prima Aditiawan, dan Y. Vita Via, “Penerapan Teknik Equivalence Partitioning Dan Boundary Value Analysis Dalam Pengujian Black Box Aplikasi Kediri Single Window For Investment (Studi Kasus : Dinas Penanaman Modal Dan Pelayanan Terpadu Satu Pintu Kota Kediri),” *J. Mhs. Tek. Inform.*, vol. 8, no. 3, hal. 3501–3510, 2024.
- [11] P. Ayu Desi Anggara Santi, R. Afwani, M. Ali Albar, S. Endang Anjarwani, dan A. ZafrullahMardiansyah, “Black Box Testing with Equivalence Partitioning and Boundary Value Analysis Methods (Study Case : Academic Information System of Mataram University),” *J. P.A.D.A. Santi al.*, hal. 207–219, 2022.
- [12] A. Valerian, D. Bernady, F. Wahyudi, J. Pratama Dinatha, dan O. Dhylan Barletyano, “Implementasi Unit Testing, Integration Testing, System Testing, Dan Validation Testing Pada Aplikasi Berbasis Website,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 4, hal. 6354–6362, 2025.
- [13] T. Maulana dan A. Voutama, “Black Box Testing Dengan Teknik Equivalence Partitions Pada Website Dinas Pengendalian Penduduk Dan

- Keluarga Berencana Karawang,” *J. Sist. Inf. dan Inform.*, vol. 6, no. 2, hal. 112–121, 2023.
- [14] S. Luginawati dan A. Purno Wahyu, “Pengujian Perangkat Lunak Menggunakan Metode Regression Testing Pada Aplikasi Pembelajaran Matematika Untuk Siswa Tingkat Smp Berbasis Android,” *J. Accid. Anal. Prev.*, vol. 5, no. 1, hal. 34–46, 2023.
 - [15] A. Pratama, A. Gema Pratama, dan H. Permatasari, “Pengujian Perangkat Lunak Sistem Informasi Perpustakaan Menggunakan Metode Blackbox Testing,” *J. Univ. DUTA BANGSA SURAKARTA*, hal. 413–420, 2023.
 - [16] I. D. M. Widia, S. Rosalin, S. R. Asriningtias, dan E. Sonalita, “Black Box Testing Menggunakan Boundary Value Analysis dan Equivalence Partitioning pada Aplikasi Pengadaan Bahan Baku Batik dengan Pendekatan Use Case,” *J. Inform. Merdeka Pasuruan*, vol. 3, no. 1, hal. 15–21, 2022.
 - [17] M. Syaeful Fajar, S. Veri Yulianto, N. Hafidhoh, dan T. Lestariningsih, “Implementasi Panic Button Berbasis Android Sebagai Bentuk Kewaspadaan, Pencegahan, dan Keamanan Pada Masa Pandemi Covid-19,” *JPM (Jurnal Pemberdaya. Masyarakat)*, vol. 7, no. 1, hal. 731–738, 2022.
 - [18] M. Sidi Mustaqbal, R. Fajri Firdaus, dan H. Rahmadi, “Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan Snmptn),” *J. Ilm. Teknol. Inf. Terap.*, vol. I, no. 3, hal. 31–36, 2015.
 - [19] S. Rika Yulistina, T. Nurmala, R. Muhamad Agung Trisakti Supriawan, S. Hidayat Ikhsan Juni, dan A. Saifudin, “Penerapan Teknik Boundary Value Analysis untuk Pengujian Aplikasi Penjualan Menggunakan Metode Black Box Testing,” *J. Inform. Univ. Pamulang*, vol. 5, no. 2, hal. 129–135, 2020.
 - [20] A. Ariftha Arwaz, T. Kusumawijaya, R. Putra, K. Putra, dan A. Saifudin, “Pengujian Black Box pada Aplikasi Sistem Seleksi Pemenang Tender Menggunakan Teknik Equivalence Partitions,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, hal. 130–134, 20