Indonesian Journal of Multidisciplinary on Social and Technology
Homepage: https://journal.ilmudata.co.id/index.php/ijmst
Vol. 4 No. 3 (2026) pp: 6516-6522
P-ISSN: 2986-6790, e-ISSN: 2986-6782

Pengujian Black Box pada Sistem Sewa Gedung Diklat Tangsel menggunakan Teknik Equivalence Partitioning

Ade Ardianto, Endri Puta Bintang, Rajin Nahampun, Naufal Nasrullah Burhan, Rafi Basilva Husodo, Raflizal Iqbal Alamsyah
Teknik Informatika, Ilmu Komputer, Universitas Pamulang
adeardianto@gmail.com, putabintang8@gmail.com, dosen00404@gmail.com, naufal150305@gmail.com, rafibasilvahusodo184@gmail.com, raflizaliqbalalamsyah@gmail.com

Abstrak

Penelitian ini membahas pengujian fungsional pada sistem sewa Gedung Diklat Tangsel berbasis web dengan metode Black Box menggunakan teknik Equivalence Partitioning. Sistem memungkinkan pengguna melakukan pemesanan gedung secara daring serta memperoleh rekomendasi melalui fitur Decision Support System (DSS) dengan metode Simple Additive Weighting (SAW) berdasarkan ketersediaan jadwal, kapasitas ruangan, harga sewa, dan kelengkapan fasilitas. Pengujian difokuskan pada dua bagian utama, yaitu Form Masuk (Login) dan Form Booking Gedung, untuk mengevaluasi kesesuaian fungsi, masukan, keluaran, serta mekanisme validasi terhadap kebutuhan pengguna. Tahapan pengujian meliputi identifikasi fitur, penyusunan test case berdasarkan kelas masukan valid dan tidak valid, penentuan hasil yang diharapkan, pelaksanaan pengujian, dan perbandingan hasil aktual dengan hasil yang diharapkan. Sebanyak enam skenario pengujian dilakukan, terdiri atas empat skenario pada Form Login dan dua skenario pada Form Booking Gedung. Hasil pengujian menunjukkan bahwa proses login dengan data yang benar berjalan sesuai harapan dan pengajuan booking yang lengkap berhasil disimpan dengan menghasilkan kode booking unik serta status Pending pada riwayat penyewaan. Namun, pengujian juga menunjukkan bahwa validasi format email dan kolom wajib masih ditangani oleh validasi bawaan browser HTML5. Kondisi tersebut menyebabkan keberadaan validasi pada sisi server belum dapat dipastikan. Oleh karena itu, sistem perlu dilengkapi validasi back-end dan pengujian lanjutan agar keandalan, keamanan, serta konsistensi proses input data dapat ditingkatkan.

Kata Kunci: Pengujian Black Box; Equivalence Partitioning; Sistem Sewa Gedung; Gedung Diklat Tangsel

Abstract

This study examines functional testing of the web-based Tangsel Training Building rental system using the Black Box method with the Equivalence Partitioning technique. The system enables users to book training facilities online and provides recommendations through a Decision Support System (DSS) based on the Simple Additive Weighting (SAW) method, considering schedule availability, room capacity, rental price, and facility completeness. Testing focuses on two main components, namely the Login Form and the Building Booking Form, to evaluate whether system functions, inputs, outputs, and validation mechanisms operate according to user requirements. The testing procedure includes identifying relevant features, designing test cases from valid and invalid input classes, defining expected results, executing the test scenarios, and comparing actual outputs with expected outcomes. Six test scenarios were conducted, consisting of four scenarios for the Login Form and two scenarios for the Building Booking Form. The results show that login using valid credentials works as expected, while a complete booking request is successfully stored, generates a unique booking code, and appears in the rental history with Pending status. However, invalid email formats and required fields are currently handled by the browser's native HTML5 validation. Consequently, the presence of an independent server-side validation layer cannot be confirmed from these tests. The system should therefore implement back-end validation and conduct additional testing to improve reliability, security, and consistency in processing user input.

Keywords: Black Box Testing; Equivalence Partitioning; Building Rental System; Tangsel Training Building; Web Application

1. Pendahuluan

Pengujian adalah bagian penting dari proses pengembangan aplikasi dan dilakukan untuk menemukan kesalahan dalam isi, fungsionalitas, fitur, kinerja, dan keamanan perangkat lunak. Pengujian perangkat lunak adalah proses mengoperasikan aplikasi dengan tujuan menemukan kesalahan pada aplikasi tersebut dengan membandingkan hasil aktual dengan fungsi yang sebenarnya diperlukan, sehingga dapat dihasilkan aplikasi (Uminingsih et al., 2022; Widia et al., 2022; Wulandari et al., 2022; Zidan et al., 2022).

Dalam artikel ini, perangkat lunak yang diuji adalah Gedung Diklat Tangsel, sebuah sistem informasi sewa gedung yang dapat diakses melalui situs webnya yang beralamat http://sewagedungdiklat.rf.gd/. User yang ingin menyewa gedung dapat melihat daftar gedung yang tersedia, serta kapasitas dan harga sewanya, mengajukan booking gedung pada tanggal dan waktu yang diinginkan, dan melihat status pengajuan melalui menu Riwayat. Selain itu, sistem ini dilengkapi dengan fitur Rekomendasi DSS (Decision Support System) dengan metode Simple Additive Weighting (SAW). Metode ini membantu pengguna memilih gedung dan jadwal terbaik berdasarkan empat kriteria: ketersediaan jadwal (bobot 30%), kapasitas ruangan (bobot 25%), harga sewa (bobot 25%, bersifat biaya), dan kelengkapan fasilitas (bobot 20%). Dari sudut pandang administrator, sistem menyediakan panel untuk mengelola data gedung, jadwal, dan penyewaan.

Email dan password digunakan untuk login ke sistem Gedung Diklat Tangsel. Hasil observasi awal menunjukkan bahwa validasi pada kolom email menggunakan validasi bawaan browser—validasi HTML5 native—yang memeriksa keberadaan simbol "@" pada masukan. Ini berbeda dengan validasi yang dibuat secara khusus oleh sistem. Hal serupa juga terjadi pada validasi kolom wajib diisi (required field) pada formulir login dan booking gedung. Ini dilakukan oleh atribut HTML5 "wajib" browser daripada logika back-end. Kondisi ini sangat penting untuk pengujian karena dalam penelitian sejenis terhadap sistem informasi akademik berbasis web ditemukan bahwa ketergantungan pada validasi pada sisi klien (browser) tanpa validasi pada sisi server dapat menjadi sumber kerentanan. Hal ini dapat dilakukan dengan cara-cara tertentu, seperti menonaktifkan JavaScript atau mengirimkan permintaan langsung ke server, untuk menghindari validasi browser.

Pengujian Black Box digunakan untuk menguji website Gedung Diklat Tangsel. Ini membantu mengetahui apakah fungsi, masukan, dan keluaran program perangkat lunak sesuai dengan yang ditentukan atau diinginkan. Metode pengujian melibatkan menjalankan atau mengeksekusi unit aplikasi secara online yang dapat diakses publik. Setelah itu, hasil yang sebenarnya dan yang diharapkan dibandingkan (Huda et al., 2022; Novalia & Voutama, 2022; Raihan & Voutama, 2023; Triady et al., 2023).

Berdasarkan permasalahan pada sistem Gedung Diklat Tangsel, pengujian dirancang dengan Black Box teknik Equivalence Partitioning melalui beberapa tahapan, yaitu: identifikasi fitur, pembuatan deskripsi pengujian (test case), penentuan hasil yang diharapkan, pelaksanaan pengujian, dan penarikan kesimpulan.

2. Metodologi

Salah satu cara untuk menguji program perangkat lunak adalah dengan menggunakan pengujian kotak hitam, yang berkonsentrasi pada sisi kemampuan program, khususnya pada input dan output perangkat lunak, untuk memastikan apakah hasilnya sudah sesuai dengan perkiraan.

Teknik Equivalence Partitioning, yang menggunakan pendekatan Black Box, membagi data masukan ke dalam kelas-kelas yang menunjukkan status program yang diinginkan. Kasus uji berbasis nilai kelas digunakan untuk menguji status ini. Nilai boolean, numerik, rentang nilai, atau format tertentu dapat dimasukkan (Maulana & Voutama, 2023; Pratama et al., 2023; Yulianti et al., 2023).

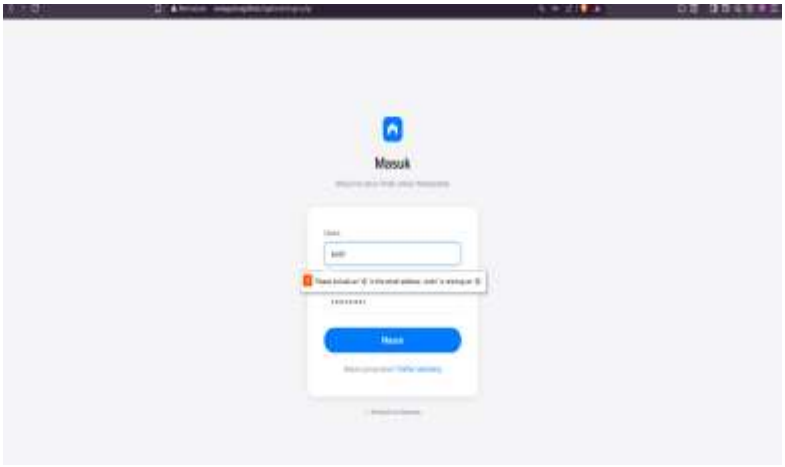
DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Rancangan atau rangkaian yang diambil melalui pengujian untuk memverifikasi kemampuan atau fitur tertentu suatu perangkat lunak dikenal sebagai test case. Identitas kasus uji, deskripsinya, hasil yang diharapkan, hasilnya, dan kesimpulan adalah bagian dari kasus uji.

Dua form utama, Form Masuk (Login) dan Form Booking Gedung, digunakan untuk menguji sistem Gedung Diklat Tangsel. Langkah pertama adalah membuat kasus uji untuk menentukan kemungkinan skenario. Setelah itu, pengujian masukan dan fitur dilakukan sesuai dengan rancangan sebelumnya (Azizah et al., 2024; Frayudha et al., 2024; Hardika et al., 2024; Ismiati et al., 2024).

2.1 Pengujian Form Login (Masuk)

Form Masuk pada sistem Gedung Diklat Tangsel meminta pengguna untuk memasukkan Email dan Password. Dalam pengujian ini, masukan email dan password tidak boleh kosong, dan format email harus mengikuti standar format email yang valid (mengandung simbol “@” dan domain yang sesuai).



Gambar 1. Form Masuk (Login) Gedung Diklat Tangsel — contoh kasus email tanpa simbol “@” ditolak oleh validasi browser

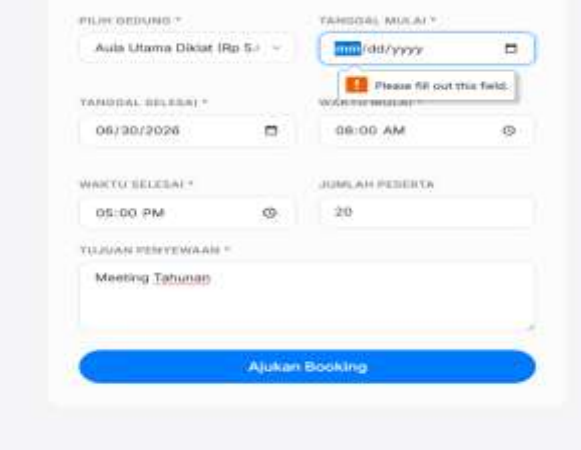
Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
L01	Mengisi email yang sudah terdaftar (contoh: admin@gedungdiklat.id), mengisi password yang benar, lalu klik "Masuk"	Sistem berhasil membaca email dan password yang sudah tersimpan di database, lalu mengarahkan user ke halaman Dashboard sesuai peran (Admin/User).	Data dengan email dan password yang benar berhasil dibaca oleh database, user diarahkan ke halaman Dashboard dengan pesan “Selamat datang, Administrator!”	Sesuai
L02	Mengisi email dengan format tidak valid, contoh "admid" (tanpa simbol "@" dan domain), lalu klik "Masuk"	Sistem (back-end) menolak format email yang tidak valid sebelum data dikirim ke server, dan menampilkan pemberitahuan kesalahan format email.	Input “admid” ditolak oleh validasi bawaan browser (HTML5 native validation) dengan pesan “Please include an '@' in the email address. 'admid' is missing an '@'.” Data tidak sempat dikirim ke server, sehingga tidak dapat dipastikan apakah back-end memiliki validasi format email tersendiri.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)
L03	Email dikosongkan, mengisi password, lalu klik "Masuk"	Sistem menampilkan pemberitahuan bahwa field	Muncul tooltip peringatan bawaan browser “Please fill out this field.” pada kolom Email.	Sesuai (tervalidasi, namun validasi

DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		email wajib diisi sebelum proses login dilanjutkan.	Proses login tidak dilanjutkan ke server.	murni mengandalkan browser, bukan sistem)
L04	Mengisi email, password dikosongkan, lalu klik "Masuk"	Sistem menampilkan pemberitahuan bahwa field password wajib diisi sebelum proses login dilanjutkan.	Muncul tooltip peringatan bawaan browser “Please fill out this field.” pada kolom Password. Proses login tidak dilanjutkan ke server.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)

2.2 Pengujian Form Booking Gedung

Form Booking Gedung terdiri atas field Pilih Gedung, Tanggal Mulai, Tanggal Selesai, Waktu Mulai, Waktu Selesai, Jumlah Peserta, dan Tujuan Penyewaan. Berdasarkan pengamatan terhadap form ini, dilakukan pengujian untuk memastikan bahwa seluruh field yang ditandai wajib (*) tidak dapat dikosongkan, dan bahwa pengajuan booking yang lengkap dan valid berhasil tersimpan ke dalam sistem dengan menghasilkan kode booking unik.



Gambar 2. Form Booking Gedung — contoh kasus field Tanggal Mulai dikosongkan ditolak oleh validasi browser



Gambar 3. Form Login — contoh kasus email dikosongkan mengisi password lalu login

DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)



Gambar 4. Form Login — contoh kasus password dikosongkan mengisi email lalu login

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
B01	Mengisi seluruh field wajib pada form Booking Gedung: memilih gedung "Ruang Kelas B (Rp 1.800)", Tanggal Mulai dan Tanggal Selesai "26 Juni 2026", Waktu Mulai "08:00 AM", Waktu Selesai "05:00 PM", serta mengisi Tujuan Penyewaan, lalu klik "Ajukan Booking"	Sistem berhasil membaca seluruh data booking, menyimpannya ke database, menghasilkan kode booking otomatis, dan menampilkan notifikasi booking berhasil diajukan.	Booking berhasil diajukan dengan kode otomatis BK-20260625-24DBA. Sistem menampilkan notifikasi hijau “Booking berhasil diajukan! Kode: BK-20260625-24DBA” dan data langsung muncul di halaman Riwayat Penyewaan dengan status “Pending”, gedung “Ruang Kelas B”, tanggal “26 Juni 2026”, waktu “08:00–17:00 WIB”, dan biaya “Rp 1.800.000”.	Sesuai
B02	Memilih gedung pada field "Pilih Gedung", namun mengosongkan field "Tanggal Mulai" (field wajib lain tetap terisi default), lalu klik "Ajukan Booking"	Sistem menampilkan pemberitahuan bahwa field Tanggal Mulai wajib diisi, dan proses pengajuan booking tidak dilanjutkan ke server.	Muncul tooltip peringatan bawaan browser “Please fill out this field.” tepat di atas field Tanggal Mulai yang masih menampilkan placeholder “mm/dd/yyyy”. Proses booking tidak dilanjutkan.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)

3. Hasil dan Pembahasan

Setelah membuat rancangan test case pengujian, dilakukan beberapa pengujian sesuai dengan rancangan yang sudah dibuat. Setelah itu disusun kesimpulan berdasarkan perbandingan antara output yang diharapkan dengan output yang dikeluarkan oleh sistem. Jika output sudah sesuai dengan yang diharapkan, maka pengujian dinyatakan berhasil; jika belum, maka perlu segera diperbaiki oleh pengembang aplikasi. Hasil dari pengujian secara lengkap dapat dilihat pada Tabel 3 (Artanto et al., 2024; Efda & Nuryasin, 2024; Akbar et al., 2025; Rizal & Nuryasin, 2025).

Tabel 3. Hasil Pengujian

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
L01	Mengisi email yang sudah terdaftar (contoh: admin@gedungdiklat.id), mengisi password yang benar, lalu klik "Masuk"	Sistem berhasil membaca email dan password yang sudah tersimpan di database, lalu mengarahkan user ke halaman	Data dengan email dan password yang benar berhasil dibaca oleh database, user diarahkan ke halaman Dashboard dengan pesan “Selamat datang, Administrator!”	Sesuai

DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
		Dashboard sesuai peran (Admin/User).		
L02	Mengisi email dengan format tidak valid, contoh "admid" (tanpa simbol "@" dan domain), lalu klik "Masuk"	Sistem (back-end) menolak format email yang tidak valid sebelum data dikirim ke server, dan menampilkan pemberitahuan kesalahan format email.	Input “admid” ditolak oleh validasi bawaan browser (HTML5 native validation) dengan pesan “Please include an '@' in the email address. 'admid' is missing an '@'.” Data tidak sempat dikirim ke server, sehingga tidak dapat dipastikan apakah back-end memiliki validasi format email tersendiri.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)
L03	Email dikosongkan, mengisi password, lalu klik "Masuk"	Sistem menampilkan pemberitahuan bahwa field email wajib diisi sebelum proses login dilanjutkan.	Muncul tooltip peringatan bawaan browser “Please fill out this field.” pada kolom Email. Proses login tidak dilanjutkan ke server.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)
L04	Mengisi email, password dikosongkan, lalu klik "Masuk"	Sistem menampilkan pemberitahuan bahwa field password wajib diisi sebelum proses login dilanjutkan.	Muncul tooltip peringatan bawaan browser “Please fill out this field.” pada kolom Password. Proses login tidak dilanjutkan ke server.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)
B01	Mengisi seluruh field wajib pada form Booking Gedung: memilih gedung "Ruang Kelas B (Rp 1.800)", Tanggal Mulai dan Tanggal Selesai "26 Juni 2026", Waktu Mulai "08:00 AM", Waktu Selesai "05:00 PM", serta mengisi Tujuan Penyewaan, lalu klik "Ajukan Booking"	Sistem berhasil membaca seluruh data booking, menyimpannya ke database, menghasilkan kode booking otomatis, dan menampilkan notifikasi booking berhasil diajukan.	Booking berhasil diajukan dengan kode otomatis BK-20260625-24DBA. Sistem menampilkan notifikasi hijau “Booking berhasil diajukan! Kode: BK-20260625-24DBA” dan data langsung muncul di halaman Riwayat Penyewaan dengan status “Pending”, gedung “Ruang Kelas B”, tanggal “26 Juni 2026”, waktu “08:00–17:00 WIB”, dan biaya “Rp 1.800.000”.	Sesuai
B02	Memilih gedung pada field "Pilih Gedung", namun mengosongkan field "Tanggal Mulai" (field wajib lain tetap terisi default), lalu klik "Ajukan Booking"	Sistem menampilkan pemberitahuan bahwa field Tanggal Mulai wajib diisi, dan proses pengajuan booking tidak dilanjutkan ke server.	Muncul tooltip peringatan bawaan browser “Please fill out this field.” tepat di atas field Tanggal Mulai yang masih menampilkan placeholder “mm/dd/yyyy”. Proses booking tidak dilanjutkan.	Sesuai (tervalidasi, namun validasi murni mengandalkan browser, bukan sistem)

Pengujian website Gedung Diklat Tangsel dilakukan pada dua form, yaitu Form Masuk (Login) diuji sebanyak empat kali dan Form Booking Gedung diuji sebanyak dua kali. Secara total, terdapat enam kali pengujian.

4. Kesimpulan

Metode Black Box digunakan untuk menguji perangkat lunak di website Gedung Diklat Tangsel. Metode ini memungkinkan pengujian tanpa melihat kode sistem yang digunakan. Hasil pengujian yang dilakukan pada form Masuk (Login) dan Form Booking Gedung menunjukkan bahwa proses utama sistem—proses autentikasi pengguna dan pengajuan booking gedung—berjalan dengan baik. Data yang dimasukkan valid dan lengkap, dan proses

DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

pengajuan booking berhasil, menghasilkan kode unik (BK-20260625-24DBA), dan tercatat pada Riwayat Penyewaan dengan status Pending. Namun, pengujian juga menemukan satu catatan penting tentang bagian validasi data: seluruh mekanisme validasi yang ditemukan pada kedua formulir—baik validasi format email ("Please include an "@" in the email address") maupun validasi kolom wajib diisi ("Please fill out this field")—merupakan validasi bawaan browser (HTML5 native validation) dan bukan validasi yang secara eksplisit dilakukan oleh sistem atau back-end. Hasilnya menunjukkan bahwa validasi data sangat penting jika bergantung pada lebih dari satu lapisan. Ini adalah masalah umum dalam penelitian pengujian black box sejenis.

Referensi

1. Akbar, D. F., Nugraha, L. R. S. R., Aldi, P. N., Akbar, A. P. F., Ulhaq, F. Z., & Atmaja, S. A. (2025). Penerapan Black Box Testing Menggunakan Teknik Equivalence Partitioning pada Sistem Informasi Akademik (SIAKAD) Kampus UKRI. Journal on Pustaka Cendekia Informatika, 3(1), 76–89. https://doi.org/10.70292/pctif.v3i1.49
2. Artanto, F. A., Khambali, A., Nadifa, S., & Azarine, V. A. (2024). Black Box Testing Dengan Teknik Equivalence Partitioning Pada Aplikasi MJ Autocare. Digital Transformation Technology, 4(1), 693–698. https://doi.org/10.47709/digitech.v4i1.4240
3. Azizah, D. N., Mahendar, I. A., Alfatih, M. F., Anwar, S. I., Al Hapid, N. M., Wicaksono, A., & Mindara, G. P. (2024). Analysis and Testing of the Combox Web Application System Using Black Box Testing with the Equivalence Partitioning Method. International Journal of Electrical Engineering, Mathematics and Computer Science, 1(4), 37–43. https://doi.org/10.62951/ijeemcs.v1i4.118
4. Efda, M. R., & Nuryasin, I. (2024). Pengujian Aplikasi Scantion Menggunakan Metode Blackbox dengan Teknik Equivalence Partitioning. Jurnal Teknologi Sistem Informasi dan Aplikasi, 7(2), 832–838. https://doi.org/10.32493/jtsi.v7i2.39048
5. Frayudha, A. D., Pande, I. R., & Juwita, M. B. (2024). Implementation of Black Box Testing with the Application of Equivalence Partitioning Techniques in the M-Magazine Android Application at Semen Gresik High School. Elinvo (Electronics, Informatics, and Vocational Education), 9(1), 134–143. https://doi.org/10.21831/elinvo.v9i1.70382
6. Hardika, B., Kurniawan, M. D., Adzka, M., Prastowiyono, D., Banyubasa, A., Wicaksono, A., & Nasir, M. (2024). Blackbox Testing of the Garuda Farm Website Using the Equivalence Partitioning Technique. Jurnal Kridatama Sains dan Teknologi, 6(02), 740–753. https://doi.org/10.53863/kst.v6i02.1420
7. Huda, M. N., Burhan, M., Satibi, A., Pradita, H. A., Saifudin, A., & Kusyadi, I. (2022). Implementasi Black Box Testing pada Aplikasi Sistem Kasir dengan Menggunakan Teknik Equivalence Partitions. Jurnal Teknologi Sistem Informasi dan Aplikasi, 5(2), 120–127. https://doi.org/10.32493/jtsi.v5i2.17645
8. Ismiati, S., Aditiawan, F. P., & Nurlaili, A. L. (2024). Black Box Testing with the Equivalence Partitioning and Cause Effect Graph Method in Archive Information System. Jurnal Teknik Informatika (JUTIF), 5(4), 981–990. https://doi.org/10.52436/1.jutif.2024.5.4.1944
9. Maulana, T., & Voutama, A. (2023). A Black Box Testing Dengan Teknik Equivalence Partitions Pada Website Dinas Pengendalian Penduduk dan Keluarga Berencana Karawang. Jurnal Sistem Informasi dan Informatika (Simika), 6(2), 112–121. https://doi.org/10.47080/simika.v6i2.2536
10. Novalia, E., & Voutama, A. (2022). Black Box Testing with the Equivalence Partitions Technique on the M-Magazine Mading School Android Application. Syntax: Jurnal Informatika, 11(01), 23–35. https://doi.org/10.35706/syji.v11i01.6413
11. Pratama, S. D., Lasimin, L., & Dadaprawira, M. N. (2023). Pengujian Black Box Testing pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence dan Boundary Value. Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD, 6(2), 560–569. https://doi.org/10.53513/jsk.v6i2.8166
12. Raihan, H., & Voutama, A. (2023). Pengujian Black Box Pada Aplikasi Database Perguruan Tinggi dengan Teknik Equivalence Partition. Antivirus: Jurnal Ilmiah Teknik Informatika, 17(1), 1–18. https://doi.org/10.35457/antivirus.v17i1.2501
13. Rizal, D. F., & Nuryasin, I. (2025). Implementasi Blackbox Testing Pada Sistem Informasi Sirkulasi Perpustakaan Berbasis Website dengan Teknik Equivalence Partitioning. Decode: Jurnal Pendidikan Teknologi Informasi, 5(1), 65–78. https://doi.org/10.51454/decode.v5i1.1052
14. Triady, D., Musdar, I. A., & Surasa, H. (2023). Blackbox Testing on Worker's Website Using Equivalence Partitioning Method. KHARISMA Tech, 18(1), 84–98. https://doi.org/10.55645/kharismatech.v18i1.292
15. Uminingsih, Ichsanudin, M. N., Yusuf, M., & Suraya, S. (2022). Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan dengan Metode Black Box Testing Bagi Pemula. STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer, 1(2), 1–8. https://doi.org/10.55123/storage.v1i2.270
16. Widia, I. D. M., Rosalin, S., Asriningtias, S. R., & Sonalita, E. (2022). Black Box Testing Menggunakan Boundary Value Analysis dan Equivalence Partitioning pada Aplikasi Pengadaan Bahan Baku Batik dengan Pendekatan Use Case. JIMP: Jurnal Informatika Merdeka Pasuruan, 6(1), 16–21. https://doi.org/10.37438/jimp.v6i1.300
17. Wulandari, A. S., Saepudin, A., Kinanti, M. P., Sudesi, Z., Saifudin, A., & Yulianti, Y. (2022). Pengujian Aplikasi Sistem Informasi Akademik Berbasis Web Menggunakan Metode Black Box Testing Equivalence Partitioning. Jurnal Teknologi Sistem Informasi dan Aplikasi, 5(2), 102–109. https://doi.org/10.32493/jtsi.v5i2.17561
18. Yulianti, Y., Al Khaidar, A., Fazriansyah, R., Ramadhan, S. G., Putra, W. A., & Sitio, S. L. M. (2023). Pengujian Black Box pada Website MyUnpam Menggunakan Teknik Equivalence Partitioning. Jurnal Teknologi Sistem Informasi dan Aplikasi, 6(2), 154–161. https://doi.org/10.32493/jtsi.v6i2.25501
19. Zidan, M., Nur'aini, S., Wibowo, N. C. H., & Ulinuha, M. A. (2022). Black Box Testing pada Aplikasi Single Sign On (SSO) di Diskominfostandi Menggunakan Teknik Equivalence Partitions. Walisongo Journal of Information Technology, 4(2), 127–137. https://doi.org/10.21580/wjit.2022.4.2.12135

DOI: https://doi.org/10.69693/ijmst.v4i3.11224
Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)