



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 2 (2026) pp: 14748-14755

P-ISSN: 2963-9298, e-ISSN: 2963-914X

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, dosen03145@unpam.ac.id, naufal150305@gmail.com,

rafiibasilvahusodo184@gmail.com, raflizaliqbalalamsyah@gmail.com

Abstrak

Aplikasi berbasis web Gedung Diklat Tangsel memungkinkan calon penyewa untuk memesan (booking) gedung diklat secara online. Dengan fitur Decision Support System (DSS) dan metode Simple Additive Weighting (SAW), pengguna dapat memilih jadwal terbaik berdasarkan berbagai faktor seperti ketersediaan jadwal, kapasitas ruangan, harga sewa, dan kelengkapan fasilitas. Pengujian yang dilakukan dengan teknik Partitioning Equivalence dilakukan untuk mengevaluasi apakah fungsi web memenuhi harapan dan kebutuhan pengguna untuk memastikan bahwa situs web sudah sesuai dengan kebutuhan pengguna. Salah satu metode pengujian perangkat lunak yang dikenal sebagai pengujian black box berpusat pada fungsionalitas, khususnya pada masukan dan keluaran aplikasi, tanpa melihat struktur kode di dalamnya. Teknik pengujian black box ini membagi domain data masukan ke dalam kelas-kelas data yang dianggap valid atau tidak valid, dan dengan menggunakan kelas-kelas ini, kasus uji, atau test case, dapat diturunkan dengan lebih efisien. Dua form utama, Form Masuk (Login) dan Form Booking Gedung, diuji. Hasil pengujian menunjukkan bahwa validasi pada kedua bentuk—validasi format email dan validasi kolom wajib—menggunakan validasi bawaan browser (validasi HTML5 native). Oleh karena itu, tidak jelas apakah server memiliki lapisan validasi tersendiri.

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

1. Latar Belakang

Pengujian adalah bagian penting dari proses pengembangan aplikasi dan dilakukan untuk menemukan kesalahan dalam isi, fungsionalitas, fitur, kinerja, dan keamanan perangkat lunak (Fahrezi et al., 2022). 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 yang sesuai dengan kebutuhan pengguna, sebagaimana ditunjukkan pada pengujian aplikasi e-learning berbasis web (Amalia et al., 2021).

Perangkat lunak yang diuji dalam artikel ini adalah Gedung Diklat Tangsel, sebuah sistem informasi sewa gedung yang dapat diakses di situs webnya di <http://sewagedungdiklat.rf.gd/>. Orang 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 menggunakan menu Riwayat untuk melihat status pengajuan. Selain itu, sistem ini memiliki fitur Rekomendasi DSS (Decision Support System) melalui metode Simple Additive Weighting (SAW). Metode ini membantu pengguna memilih gedung dan jadwal terbaik berdasarkan empat kriteria: ketersediaan jadwal (30%), kapasitas ruangan (25%), harga sewa (25%, bersifat biaya), dan kelengkapan fasilitas (20%). Dari sudut pandang administrator, sistem menyediakan panel yang memungkinkan pengaturan 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 (Marlina & Wibowo, 2023; Wulandari et al., 2022). Temuan serupa juga dilaporkan pada pengujian front-end sistem akademik lain, di mana validasi wajib isi dan format data sepenuhnya ditangani oleh atribut HTML5 tanpa validasi tambahan di sisi server (Widhyaestoeti et al., 2021). 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 Tangel. Ini membantu mengetahui apakah fungsi, masukan, dan keluaran program perangkat lunak sesuai dengan yang ditentukan atau diinginkan (Susanto et al., 2021; Huda et al., 2022). Metode pengujian melibatkan menjalankan atau mengeksekusi unit aplikasi secara online yang dapat diakses publik. Setelah itu, hasil yang sebenarnya dan yang diharapkan dibandingkan.

Berdasarkan permasalahan pada sistem Gedung Diklat Tangel, 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, mengikuti tahapan yang serupa dengan penelitian pengujian black box pada website MyUnpam (Yulianti et al., 2023) dan pengujian sistem informasi akademik berbasis website (Yulianti et al., 2022).

2. Metode Penelitian

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 (Fahrezi et al., 2022; Efdi & Nuryasin, 2024).

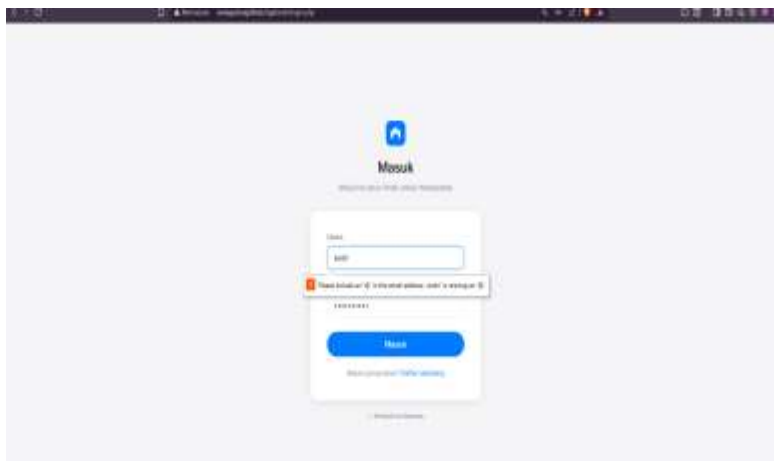
Teknik Partitioning Equivalence, yang menggunakan pendekatan Black Box, membagi data masukan ke dalam kelas-kelas yang menunjukkan status program yang diinginkan (Novalia & Voutama, 2022). Kasus uji berbasis nilai kelas digunakan untuk menguji status ini. Nilai boolean, numerik, rentang nilai, atau format tertentu dapat dimasukkan. Teknik ini telah banyak diterapkan pada berbagai jenis dan platform aplikasi, mulai dari aplikasi berbasis web hingga aplikasi Android (Frayudha et al., 2024).

Rancangan atau rangkaian yang diambil melalui pengujian untuk memverifikasi kemampuan atau fitur tertentu suatu perangkat lunak dikenal sebagai test case (Putri, 2022). Identitas kasus uji, deskripsinya, hasil yang diharapkan, hasilnya, dan kesimpulan adalah bagian dari kasus uji, sebagaimana diterapkan pada pengujian aplikasi berbasis web lainnya (Pratama et al., 2023).

Dua form utama, Form Masuk (Login) dan Form Booking Gedung, digunakan untuk menguji sistem Gedung Diklat Tangel, sejalan dengan pendekatan pengujian dua form utama pada penelitian pengujian aplikasi pemilihan karyawan terbaik (Desyani et al., 2022). Langkah pertama adalah membuat kasus uji untuk menentukan kemungkinan skenario. Setelah itu, pengujian masukan dan fitur dilakukan sesuai dengan rancangan sebelumnya.

2.1 Pengujian Form Login (Masuk)

Form Masuk pada sistem Gedung Diklat Tangel 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 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)

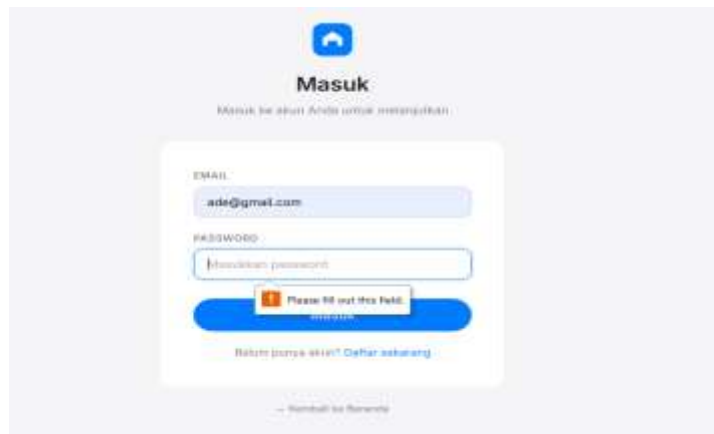
Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
L04	Mengisi email, password dikosongkan, lalu klik "Masuk"	Sistem menampilkan pemberitahuan bahwa field password wajib diisi sebelum proses login dilanjutkan.	Muncul tooltip peringatan bawaan browser "Please fill out this field." 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



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 Diskusi

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, sebagaimana dilakukan pada pengujian aplikasi sistem informasi akademik berbasis web (Wulandari et al., 2022) dan aplikasi sistem kasir (Huda et al., 2022). 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.

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 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 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",	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	Sesuai

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
	serta mengisi Tujuan Penyewaan, lalu klik "Ajukan Booking"		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".	
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 Tangel 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. Hasil pengujian ini juga memperkuat temuan pada penelitian sejenis, bahwa validasi berbasis browser (HTML5 native) sering ditemukan pada formulir login dan pendaftaran aplikasi web tanpa disertai validasi eksplisit di sisi server (Widhyaestoeti et al., 2021).

4. Kesimpulan

Perangkat lunak di website Gedung Diklat Tangel diuji dengan metode Black Box. Metode ini memungkinkan pengujian dilakukan tanpa melihat kode sistem. Hasil pengujian yang dilakukan pada formulir Masuk (Login) dan Booking Gedung menunjukkan bahwa proses utama sistem—proses autentikasi pengguna dan pengajuan booking gedung—berjalan dengan baik. Semua informasi yang dimasukkan benar dan lengkap, dan pengajuan reservasi berhasil, menghasilkan kode unik (BK-20260625-24DBA), dan mencatat status pending pada riwayat penyewaan. Namun, pengujian juga menemukan fakta penting tentang bagian validasi data: seluruh mekanisme validasi yang ditemukan pada kedua formulir—baik validasi format email maupun validasi kolom wajib diisi—merupakan validasi bawaan browser (validasi HTML5 native) dan bukan validasi yang dilakukan secara eksplisit oleh sistem atau back-end. Hasilnya menunjukkan bahwa validasi data sangat penting jika bergantung pada lebih dari satu lapisan. Dalam pengujian aplikasi e-learning berbasis web, ini adalah masalah umum dalam penelitian pengujian black box sejenis ini (Amalia et al., 2021), aplikasi penjualan elektronik berbasis desktop (Susanto et al., 2021), sistem informasi akademik berbasis web (Wulandari et al., 2022), sistem informasi perpustakaan (Marlina & Wibowo, 2023), serta pengujian front-end sistem akademik lainnya (Widhyaestoeti et al., 2021).

Referensi

1. Amalia, A., Putri Hamidah, S. W., & Kristanto, T. (2021). Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web. *Building of Informatics, Technology and Science (BITS)*, 3(3), 269–274. <https://doi.org/10.47065/bits.v3i3.1062>
2. Desyani, T., Mulyati, S., Kurnianto, E., Kamaludin, K., Afifah, N., & Fauziah, S. N. (2022). Pengujian Black Box menggunakan teknik equivalence partitions pada aplikasi sistem Pemilihan Karyawan terbaik. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 5(2), 110–114. <https://doi.org/10.32493/jtsi.v5i2.17578>
3. 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>

4. Fahrezi, A., Salam, F. N., Ibrahim, G. M., Rahman, R., & Saifudin, A. (2022). Pengujian Black Box Testing pada Aplikasi Inventori Barang Berbasis Web di PT. AINO Indonesia. *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, 1(1), 1–5. <https://www.journal.mediapublikasi.id/index.php/logic/article/view/1262>
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. Huda, M. N., Burhan, M., Satibi, A., Pradita, H. A., Saifudin, A., & Kusyad, 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>
7. Marlina, L. A., Harliana, & Wibowo, S. S. (2023). Pengujian Sistem Informasi Perpustakaan Dengan Teknik Equivalence Partitioning di SMA Nurul Muttaqin Albarokah. *JACIS: Journal Automation Computer Information System*, 3(2), 137–145. <https://jacis.pubmedia.id/index.php/jacis/article/view/55>
8. Novalia, E., & Voutama, A. (2022). Black Box Testing dengan Teknik Equivalence Partitions Pada Aplikasi Android M-Magazine Mading Sekolah. *Syntax: Jurnal Informatika*, 11(1), 23–35. <https://doi.org/10.35706/syji.v11i01.6413>
9. Pratama, S. D., Lasimin, & Dadaprawira, M. N. (2023). Pengujian Black Box Testing pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence dan Boundary Value. *J-SISKO TECH (Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD)*, 6(2), 560–569. <https://doi.org/10.53513/jsk.v6i2.8166>
10. Putri, D. I. (2022). Teknik Equivalence Partitions untuk Pengujian Aplikasi Manajemen Kas dan Inventaris Berbasis Web. *Information Management for Educators and Professionals: Journal of Information Management*, 6(2), 193. <https://doi.org/10.51211/imbi.v6i2.1922>
11. Susanto, J., Biqirrosyad, B., Junaidi, M. M., Sudrajat, Y., & Desyani, T. (2021). Pengujian Black Box pada Aplikasi Desktop Penjualan Elektronik Menggunakan Metode Equivalence Partitioning. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 4(1), 52–57. <https://doi.org/10.32493/jtsi.v4i1.8519>
12. 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>
13. 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>
14. Yulianti, Y., Desyani, T., Chaniago, R. R., Iswanto, H., Suroso, E., & Hermanto, T. S. (2022). Pengujian Aplikasi Sistem Informasi Akademik Berbasis Website Menggunakan Teknik Equivalence Partitioning dan Metode Black Box. *Jurnal Informatika Universitas Pamulang*, 7(1), 145–150. <https://doi.org/10.32493/informatika.v7i1.17528>
15. Widhyaestoeti, D., Iqam, S., Mutiyah, S. N., & Khairunnisa, Y. (2021). Black Box Testing Equivalence Partitions Untuk Pengujian Front-End Pada Sistem Akademik Sitoda. *JITTER: Jurnal Ilmiah Teknologi Infomasi Terapan*, 7(3), 211–216. <https://doi.org/10.33197/jitter.vol7.iss3.2021.626>