

## Implementasi Teknik *Equivalence Partitions* untuk Pengujian *Black Box* pada Sistem Informasi DAPODIKDASMEN

Alvin Pratama Putra<sup>1</sup>, Noviar Nurdin<sup>2</sup>, Rio Valentino Rondonuwu<sup>3</sup>, Irpan Kusyadi<sup>4</sup>

Teknik Informatika, Universitas Pamulang, Jl. Raya Puspitek No.46, Buaran, Serpong, Tangerang Selatan, Banten, Indonesia, 15310

e-mail: <sup>1</sup>alvin199123@yahoo.com, <sup>2</sup>viar.exc@gmail.com, <sup>3</sup>rio.valentino@ymail.com, <sup>4</sup>dosen00673@unpam.ac.id

Submitted Date: June 03<sup>rd</sup>, 2020

Reviewed Date: July 26<sup>th</sup>, 2020

Revised Date: August 02<sup>nd</sup>, 2020

Accepted Date: August 02<sup>nd</sup>, 2020

### Abstract

DAPODIKDASMEN (*Data Pokok Pendidikan Dasar dan Menengah*) is an application that is very important for schools to be used in managing all data related to institutional, curriculum, student data, teacher data, and employee data, as well as data on facilities and infrastructure of all schools in Indonesia and schools- Indonesian schools abroad. This application is quite complex, if something goes wrong can result in large losses. For this reason, testing of features in the DAPODIKDASMEN application is needed to ensure that there are no problems that cause losses, especially for stakeholders (stakeholders), this test is carried out to provide assurance that the DAPODIKDASMEN application is error free and suitable for use. Testing the DAPODIKDASMEN application is proposed by *Black Box* using the *Equivalence Partitions* technique so that errors / defects can be identified and can be corrected immediately. The results of testing with the *Equivalence Partitions* technique can find errors that exist due to errors in the pre-coding stage, as well as unintentional errors in coding, so that the resulting application can be guaranteed to be of high quality and in accordance with established requirements.

*Keywords: Black Box; DAPODIKDASMEN; Equivalence Partitions; Testing; Information Systems*

### Abstrak

DAPODIKDASMEN (*Data Pokok Pendidikan Dasar dan Menengah*) merupakan aplikasi ini sangat penting bagi sekolah untuk digunakan dalam mengelola semua data berhubungan dengan kelembagaan, kurikulum, data siswa, data guru, dan data pegawai, maupun data sarana dan prasarana semua sekolah di Indonesia dan sekolah-sekolah Indonesia yang ada di luar negeri. Aplikasi ini cukup kompleks, jika terjadi kesalahan dapat menimbulkan kerugian yang besar. Untuk itu diperlukan pengujian fitur pada Aplikasi DAPODIKDASMEN untuk memastikan tidak ada permasalahan yang menyebabkan kerugian terutama bagi *stakeholder* (pihak yang berkepentingan), pengujian ini dilakukan untuk memberikan jaminan bahwa aplikasi DAPODIKDASMEN telah bebas dari kesalahan dan layak untuk digunakan. Pengujian aplikasi DAPODIKDASMEN diusulkan secara *Black Box* menggunakan teknik *Equivalence Partitions* supaya dapat diketahui adanya kesalahan/cacat dan dapat segera diperbaiki. Hasil pengujian dengan teknik *Equivalence Partitions* dapat menemukan kesalahan yang ada akibat kesalahan pada tahap sebelum pengkodean, maupun kesalahan yang tidak disengaja dalam pengkodean, sehingga aplikasi yang dihasilkan dapat dijamin telah berkualitas tinggi dan sesuai dengan persyaratan yang telah ditetapkan.

*Kata Kunci: Black Box; DAPODIKDASMEN; Equivalence Partitions; Pengujian; Sistem Informasi*

### 1. Pendahuluan

Teknologi informasi ditujukan untuk menyediakan informasi dan digunakan sebagai alat bantu dalam mengerjakan pengolahan informasi (Hanum & Saifudin, 2019). Perkembangan

teknologi diikuti meningkatnya penggunaan dan kompleksitas software, sehingga jaminan kualitas software menjadi kritis dan tantangan yang sulit (Saifudin, Heryadi, & Lukas, 2019). Jaminan kualitas harus diberikan, karena jika produk

software yang dihasilkan mengandung cacat dapat menyebabkan kerugian yang lebih besar. Kesalahan/cacat yang terdapat dalam software yang dihasilkan dapat menyebabkan kegagalan dan dapat menimbulkan kerugian yang besar (Aziz, Setiawan, Khanh, Nurdiansyah, & Yulianti, 2020).

Sistem informasi dapat dinyatakan sebagai kumpulan sistem yang saling terintegrasi satu sama lain guna menghasilkan informasi yang bermanfaat bagi pemakainya (Taufiq, Magfiroh, Yusup, & Yulianti, 2020). DAPODIKASMEN (Data Pokok Pendidikan Dasar dan Menengah) merupakan aplikasi ini sangat penting bagi sekolah untuk digunakan dalam mengelola semua data berhubungan dengan kelembagaan, kurikulum, data siswa, data guru, dan data pegawai, maupun data sarana dan prasarana semua sekolah di Indonesia dan sekolah-sekolah Indonesia yang ada di luar negeri. Aplikasi ini cukup kompleks, jika terjadi kesalahan dapat menimbulkan kerugian yang besar. Untuk menjamin software yang dihasilkan telah bebas dari kesalahan umumnya dilakukan dengan pengujian yang merupakan tahapan paling mahal dalam pengembangan software (Saifudin & Yulianti, 2020).

Pengujian perangkat lunak merupakan tahap yang harus dilalui untuk menjamin kualitas sistem/aplikasi yang dikembangkan (Pratala, Asyer, Prayudi, & Saifudin, 2020). Pengujian fitur perangkat lunak merupakan metode yang digunakan untuk memperoleh suatu data tentang mutu dari perangkat lunak yang diuji (Sulistiyanto, 2014). Pengujian fitur perangkat lunak ditujukan untuk mengetahui terdapatnya kesalahan, yang menimbulkan kegagalan fitur perangkat lunak (Irawan, 2017). Proses pengujian untuk mengenali bagian dari fitur perangkat lunak yang rawan hadapi kegagalan (Hanifah, Alit, & Sugiarto, 2016). Suatu fitur perangkat lunak dinyatakan kandas, bila fitur perangkat lunak tersebut tidak memenuhi spesifikasi (MZ, 2016).

Tujuan menguji fitur program komputer dicoba untuk mendeteksi kesalahan pada fitur program komputer (Wibisono & Baskoro, 2016). Pengujian fitur program komputer bertujuan untuk mendapatkan software yang bermutu yang bebas dari kesalahan dan memenuhi kebutuhan. Pengujian aplikasi dibutuhkan untuk menjamin kalau yang dibesarkan bisa menunjang *business process* cocok dengan persyaratan (*requirement*). Kesalahan pada aplikasi bisa menyebabkan aplikasi tidak mendukung proses bisnis, serta butuh revisi ataupun pengembangan ulang bila jumlah

cacat/kesalahannya banyak (Saifudin & Wahono, 2015). Dalam proses pengujian fitur perangkat lunak, buat tiap permasalahan yang hendak diuji wajib mempunyai bukti diri serta memiliki keterkaitan antara masukan yang diberikan dengan hasil yang diharapkan (MZ, 2016).

Perangkat lunak yang belum diuji dan tidak ada jaminan bebas dari kesalahan dapat mengandung cacat/kesalahan yang bisa muncul pada saat digunakan dan menimbulkan kerugian. Kerugian yang timbul akibat cacat/kesalahan pada perangkat lunak dapat merugikan pemilik maupun *stakeholder* yang lain. Sehingga perlu dilakukan pengujian untuk memeriksa fungsional dari perangkat lunak dan mengamati hasil akhir melalui data uji yang valid (Syaban & Bunyamin, 2015).

Wajib merancang pengujian dengan baik dan sistematis supaya bisa menemukan cacat/kesalahan dan memperbaikinya dengan waktu serta usaha yang minimum. Pengujian secara *Black Box* dengan teknik *Equivalence Partitions* berupaya untuk menemukan kesalahan/cacat untuk fungsi yang hilang atau salah, kesalahan antarmuka, kesalahan pada struktur data dan informasi maupun akses mengarah database, serta kesalahan performa (Febiharsa, Sudana, & Hudallah, 2018). Dalam proses pengujian, tiap permasalahan yang hendak diuji wajib mempunyai bukti diri serta memiliki keterkaitan antara sekumpulan masukan dengan hasil yang diharapkan.

Aplikasi diuji untuk menentukan jika setiap fungsi telah beroperasi sesuai yang diharapkan sehingga aplikasi dapat dinyatakan telah sesuai persyaratan dan dapat dinyatakan layak diterapkan (Nurudin, Jayanti, Saputro, Saputra, & Yulianti, 2019). Pengujian juga bertujuan untuk menemukan kesalahan yang mungkin ada pada tiap fungsi dengan berfokus pada data masukan, antarmuka, penggunaan memori, dan waktu eksekusi, jika ada hasil atau keluaran yang tidak sesuai dengan harapan maka fungsi tersebut dinyatakan gagal (Mustaqbal, Firdaus, & Rahmadi, 2015). Fungsi yang dinyatakan gagal harus diperbaiki sebelum aplikasi digunakan. Sehingga dengan pengujian akan diketahui kesalahan/cacat yang mungkin ada pada aplikasi. Pada pengujian aplikasi DAPODIKASMEN akan dilakukan secara *Black Box* menggunakan teknik *Equivalence Partitioning* untuk mengetahui apakah fungsi-fungsi yang ditetapkan sudah valid. Pada pengujian ini bertujuan untuk mengidentifikasi kekurangan atau kelemahan dari sistem supaya informasi yang dihasilkan cocok dengan informasi yang

dimasukkan sesudah aplikasi dieksekusi dan menghindari adanya kesalahan/cacat pada aplikasi sebelum digunakan oleh user.

## 2. Metodologi

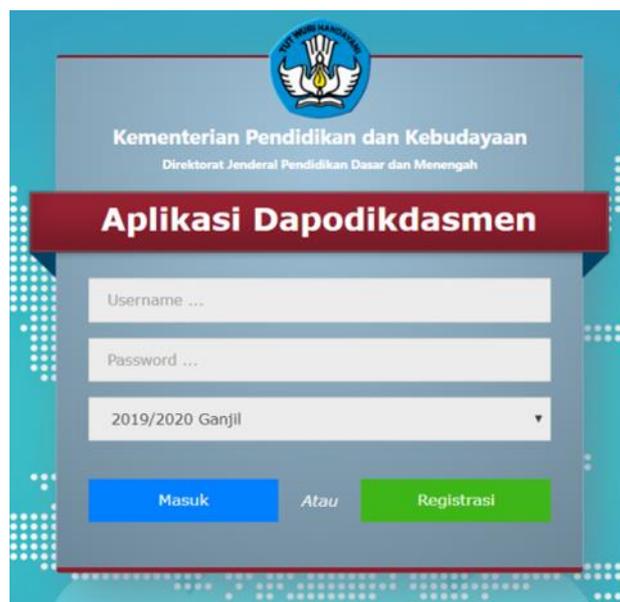
Tujuan dari pengujian adalah untuk memastikan bahwa sistem yang dihasilkan sesuai dengan kebutuhan dan layak untuk digunakan (Arwaz, Putra, Putra, Kusumawijaya, & Saifudin, 2019). Pengujian perlu dilakukan untuk untuk memverifikasi dan validasi apakah sistem yang dikembangkan telah sesuai dengan persyaratan yang ditetapkan. Pengujian terhadap perangkat lunak sangat penting dilakukan dengan tujuan untuk memberikan jaminan kualitas perangkat lunak yang dihasilkan agar bebas dari terjadinya kesalahan (Debiyanti, Sutrisna, Budrio, Kamal, & Yulianti, 2020). Sehingga pengujian merupakan tahapan yang sangat penting untuk memperbaiki kemungkinan adanya cacat/kesalahan yang dapat merugikan (Ningrum, Suherman, Aryanti, Prasetya, & Saifudin, 2019). Pengujian perangkat lunak merupakan sebuah proses pengeksekusi program yang bermaksud untuk mencari bug (error) yang menyebabkan kegagalan perangkat lunak. Pengujian dikatakan berhasil jika mampu memunculkan kesalahan dari program yang sedang diuji.

Pengujian adalah rangkaian kegiatan yang direncanakan secara sistematis untuk menguji atau mengevaluasi kebenaran fungsi aplikasi berdasarkan desain kasus uji (*test case*) yang spesifik (Yulistina, Nurmala, Supriawan, Juni, &

Saifudin, 2020). Pengujian software mempunyai dua metode yaitu blackbox dan white-box. Pengujian black-box menguji dari sisi fungsionalitas, aplikasi diuji agar dapat memenuhi fungsi sebagaimana mestinya. Sedangkan, pengujian white-box menguji berdasarkan cara kerja dari aplikasi yang telah dibuat, metode pengujian ini diarahkan untuk menunjukkan tingkat kebenaran metode yang digunakan (Akmaludin, 2013). Yang dilakukan pada pengujian menggunakan teknik *Equivalence Partitions* adalah:

1. Mengidentifikasi komponen masukan
2. Mengelompokkan nilai masukan ke dalam kelompok valid dan tidak valid
3. Membuat rancangan *test case* dengan memberi nilai masukan dari nilai tiap kelompok yang valid dan tidak valid
4. Menjalankan *test case* sesuai rancangan
5. Mendokumentasikan dan mengevaluasi hasil pengujian berdasarkan *test case* yang dibuat

Aplikasi DAPODIKDASMEN yang dikembangkan diuji menggunakan Teknik *Equivalence Partitions*. Teknik *Equivalence Partitions* mengelompokkan nilai masukan dengan status nilai valid dan tidak valid. Dari setiap kelompok yang terbentuk dipilih secara acak sebagai data masukan untuk pengujian aplikasi. Rancangan kasus uji (*test case*) untuk menguji Form Login pada Gambar 1 ditunjukkan pada Tabel 1.



Gambar 1 Form Login

Pada form dari Gambar 1, terdapat rencana pengujian *Username*, data akan valid jika *Username* diisi dengan memasukan *Username* yang sudah didaftarkan sebelumnya ke dalam database dan tidak boleh kosong. Sebaliknya data tidak valid apabila *Username* belum terdaftar

dalam database dan dikosongkan. Pada kolom *password* data akan valid apabila diisi sesuai dengan *password* yang telah terdaftar dalam database dan tidak kosong. Apabila *password* yang diisi tidak terdaftar dalam database dan kolom dikosongkan maka data tidak akan valid.

Tabel 1 Rancangan Kasus Uji untuk Form Login

No	Skenario Pengujian	Hasil yang Diharapkan
1.	Mengisi kolom <i>username</i> dengan data yang telah terdaftar yaitu "admin001", dan mengisi kolom <i>password</i> dengan data yang sesuai pada <i>username</i> di atas dalam <i>database</i> yaitu "admin1234".	Tampil <i>Pop-Up</i> bertuliskan "Berhasil Login" dan sistem menampilkan halaman beranda DAPODIKASMEN. Username yang Anda masukkan tidak terdaftar! Mohon gunakan username lain.
2.	Mengisi kolom <i>username</i> dengan data yang telah terdaftar yaitu "admin001", dan mengisi kolom <i>password</i> dengan data yang tidak sesuai pada <i>username</i> di atas dalam <i>database</i> yaitu "1234".	Tampil <i>Pop-Up</i> bertuliskan "Password yang Anda masukkan salah!" lalu sistem kembali ke halaman login.
3.	Mengisi kolom <i>username</i> dengan data yang tidak terdaftar dalam database contoh : "administrator", dan mengisi kolom <i>password</i> dengan data yang sesuai pada dalam <i>database</i> yaitu "admin1234".	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan username lain." lalu sistem kembali ke halaman login.
4.	Mengisi kolom <i>username</i> dengan data yang tidak terdaftar dalam database contoh : "administrator", dan mengisi kolom <i>password</i> dengan data yang tidak sesuai pada <i>database</i> dalam database yaitu "1234".	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan username lain." lalu sistem kembali ke halaman login.
5.	Tidak mengisi data di kolom <i>username</i> maupun di kolom <i>password</i>	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan username lain." lalu sistem kembali ke halaman login.

### 3. Hasil dan Pembahasan

Rancangan pengujian sudah dibuat, lalu dilakukan pengujian. Pengujian dilakukan pada sistem informasi DAPODIKASMEN (Data Pokok Pendidikan Dasar dan Menengah) dengan harapan dapat menemukan adanya kesalahan sebelum sistem digunakan agar dapat segera diperbaiki. Hasil pengujian yang tidak sesuai

harapan segera diperbaiki, sehingga setelah melakukan pengujian dan memperbaiki kesalahannya, maka dapat memberikan jaminan bahwa sistem yang dibuat telah bebas dari kesalahan dan layak untuk digunakan. Hasil pengujian yang dilakukan berdasarkan *test case* didokumentasikan seperti pada Tabel 2.

Tabel 2 Hasil Pengujian Form login menggunakan Teknik *Equivalence Partitioning*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
1	Mengisi kolom <i>username</i> dengan data yang telah terdaftar yaitu "admin001", dan mengisi	Tampil <i>Pop-Up</i> bertuliskan "Berhasil Login" dan sistem menampilkan	Tampil <i>Pop-Up</i> bertuliskan "Berhasil Login" dan sistem menampilkan	Berhasil

	kolom <i>password</i> dengan data yang sesuai pada <i>username</i> di atas dalam <i>database</i> yaitu "admin1234".	halaman beranda DAPODIKDAEMEN.	halaman beranda DAPODIKDAEMEN.	
2	Mengisi kolom <i>username</i> dengan data yang telah terdaftar yaitu "admin001", dan mengisi kolom <i>password</i> dengan data yang tidak sesuai pada <i>username</i> di atas dalam <i>database</i> yaitu "1234".	Tampil <i>Pop-Up</i> bertuliskan "Password yang Anda masukkan salah!" dan sistem tetap menampilkan halaman login.	Tampil <i>Pop-Up</i> bertuliskan "Password yang Anda masukkan salah!" dan sistem tetap menampilkan halaman login.	Berhasil
3	Mengisi kolom <i>username</i> dengan data yang tidak terdaftar dalam <i>database</i> yaitu "administrator", dan mengisi kolom <i>password</i> dengan data yang sesuai pada dalam <i>database</i> yaitu "admin1234".	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan <i>username</i> lain." lalu sistem kembali ke halaman login.	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan <i>username</i> lain." lalu sistem kembali ke halaman login.	Berhasil
4	Mengisi kolom <i>username</i> dengan data yang tidak terdaftar dalam <i>database</i> yaitu "administrator", dan mengisi kolom <i>password</i> dengan data yang tidak sesuai pada <i>database</i> dalam <i>database</i> yaitu "1234".	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan <i>username</i> lain." lalu sistem kembali ke halaman login.	Tampil <i>Pop-Up</i> bertuliskan "Username yang Anda masukkan tidak terdaftar! Mohon gunakan <i>username</i> lain." lalu sistem kembali ke halaman login.	Berhasil
5	Tidak mengisi data di kolom <i>username</i> maupun di kolom <i>password</i>	Tampil <i>Pop-Up</i> bertuliskan "Username dan <i>password</i> harus diisi !" lalu sistem kembali ke halaman login.	Tampil <i>Pop-Up</i> bertuliskan "Username dan <i>password</i> harus diisi !" lalu sistem kembali ke halaman login.	Berhasil

Berdasarkan pengujian yang dilakukan, beberapa kesalahan dapat ditemukan dan langsung diperbaiki. Pada karya ilmiah ini hanya ditampilkan 5 contoh pengujian yang dimulai dari merancang kasus uji (*test case*) sampai hasil pengujian. Dari pengujian yang dilakukan menunjukkan bahwa pengujian dapat menemukan kesalahan yang ada pada aplikasi DAPODIKDAEMEN dan langsung diperbaiki. Hal ini juga menunjukkan bahwa pengujian dapat meningkatkan kualitas aplikasi yang dihasilkan.

#### 4. Kesimpulan

Dari hasil pengujian form login DAPODIKDAEMEN dapat disimpulkan bahwa pengujian menggunakan metode *Black Box Testing* sangat dibutuhkan agar setiap perangkat lunak teruji dari kesalahan sebuah sistem. Dari hasil

pengujian secara *Black Box* menggunakan teknik *Equivalence Partitions* yang dilakukan dapat menemukan beberapa kesalahan dan langsung diperbaiki sehingga setelah pengujian aplikasi DAPODIKDAEMEN telah bebas dari kesalahan. Pengujian dapat digunakan untuk menjamin bahwa kualitas aplikasi yang dihasilkan memiliki kualitas yang baik dan bebas dari kesalahan, serta semua fungsi telah berfungsi sesuai persyaratan yang ditetapkan.

#### 5. Saran

Pengujian aplikasi sangat penting untuk menemukan kesalahan dan memperbaikinya untuk memberikan jaminan kualitas aplikasi. Tetapi jika fungsi aplikasi sangat banyak, maka membutuhkan banyak sumberdaya untuk pelaksanaannya. Jika fungsinya banyak disarankan untuk menerapkan

teknik otomatisasi pengujian dengan menggunakan software bantu.

## Referensi

- Akmaludin. (2013). Analisis Rancangan Animasi Interaktif Pembelajaran Anatomi Otak Manusia. *Techo Nusa Mandiri*, 1-11.
- Arwaz, A. A., Putra, K., Putra, R., Kusumawijaya, T., & Saifudin, A. (2019). Pengujian Black Box pada Aplikasi Sistem Seleksi Pemenang Tender Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 2(4), 130-134. doi:10.32493/jtsi.v2i4.3708
- Aziz, I. A., Setiawan, B., Khanh, R., Nurdiyansyah, G., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Sistem Kasir Berbasis Website Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3(2), 10.32493/jtsi.v3i2.4693. doi:10.32493/jtsi.v3i2.4693
- Debiyanti, D., Sutrisna, S., Budrio, B., Kamal, A. K., & Yulianti, Y. (2020). Pengujian Black Box pada Perangkat Lunak Sistem Penilaian Mahasiswa Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(2), 162-166. doi:10.32493/informatika.v5i2.5446
- Febiharsa, D., Sudana, I. M., & Hudallah, N. (2018). Uji Fungsionalitas (Blackbox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik dengan AppPerfect Web Test dan Uji Pengguna. *Joined Journal (Journal of Informatics Education)*, 1(2), 117-126.
- Hanifah, U., Alit, R., & Sugiarto, S. (2016). Penggunaan Metode Black Box Pada Pengujian Sistem Informasi Surat Keluar Masuk. *SCAN - Jurnal Teknologi Informasi dan Komunikasi*, 11(2), 33-40.
- Hanum, W. S., & Saifudin, A. (2019). Rancang Bangun Aplikasi Panduan Pariwisata di Kabupaten Banyuwangi Mobile Berbasis Android. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 2(2), 59-65. doi:10.32493/jtsi.v2i2.2798
- Irawan, Y. (2017). Pengujian Sistem Informasi Pengelolaan Pelatihan Kerja Upt. BLK Kabupaten Kudus dengan Metode Whitebox Testing. *Speed - Sentra Penelitian Engineering dan Edukasi*, 9(3), 59-63.
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, 1(3), 31-36. doi:10.33197/jitter.vol1.iss3.2015.62
- MZ, M. K. (2016). Pengujian Perangkat Lunak Metode Black-Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah. *Jurnal Mikrotik*, 2-8.
- Ningrum, F. C., Suherman, D., Aryanti, S., Prasetya, H. A., & Saifudin, A. (2019). Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions. *Jurnal Informatika Universitas Pamulang*, 4(4), 125-130. doi:10.32493/informatika.v4i4.3782
- Nurudin, M., Jayanti, W., Saputro, R. D., Saputra, M. P., & Yulianti, Y. (2019). Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 4(4), 143-148. doi:10.32493/informatika.v4i4.3841
- Pratala, C. T., Asyer, E. M., Prayudi, I., & Saifudin, A. (2020). Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path. *Jurnal Informatika Universitas Pamulang*, 5(2), 111-119. doi:10.32493/informatika.v5i2.4713
- Saifudin, A., & Wahono, R. S. (2015). Penerapan Teknik Ensemble untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software. *Journal of Software Engineering*, 1(1), 28-37.
- Saifudin, A., & Yulianti, Y. (2020). Dimensional Reduction on Cross Project Defect Prediction. *Journal of Physics: Conference Series*. 1477, p. 022030. Tangerang: IOP Publishing. doi:10.1088/1742-6596/1477/3/032011
- Saifudin, A., Heryadi, Y., & Lukas. (2019). Ensemble Undersampling to Handle Unbalanced Class on Cross-Project Defect Prediction. *IOP Conference Series: Materials Science and Engineering*. 662, pp. 1-8. Bandung: IOP Publishing. doi:10.1088/1757-899X/662/6/062012
- Sulistyanto, H. (2014). Urgensi Pengujian pada Kemajemukan Perangkat Lunak dalam Multi Perspektif. *Komunitas: Jurnal Komunikasi dan Teknologi Informasi*, 6(1), 65-74. doi:10.23917/komunitas.v6i1.2944
- Syaban, R. M., & Bunyamin, H. (2015). Pengembangan Sistem Informasi Pengelolaan Surat Masuk dan Keluar Berbasis Web di Dinas Sosial Tenaga Kerja dan Transmigrasi Kabupaten Garut Menggunakan Framework PHP. *Jurnal Algoritma*, 12(2), 301-311.
- Taufiq, R., Magfiroh, D. A., Yusup, D., & Yulianti, Y. (2020). Analisis dan Desain Sistem Informasi Pembayaran Sumbangan Pembinaan Pendidikan (SPP) di SMK Avicena Rajeg. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3(1), 15-21. doi:10.32493/jtsi.v3i1.4308
- Wibisono, W., & Baskoro, F. (2016). Pngujian Perangkat Lunak Dengan menggunakan model

- Behavior UML. *Jurnal Ilmiah Teknologi Informasi*, 1-8.
- Yulistina, S. R., Nurmala, T., Supriawan, R. M., Juni, S. H., & Saifudin, A. (2020). Penerapan Teknik Boundary Value Analysis untuk Pengujian Aplikasi Penjualan Menggunakan Metode Black Box Testing. *Jurnal Informatika Universitas Pamulang*, 5(2), 129-135. doi:10.32493/informatika.v5i2.5366
- Sulistyanto, H., & SN, A. (2014). Urgensi Pengujian pada Kemajemukan Perangkat Lunak dalam Multi Perspektif. *Komuniti: Jurnal Komunikasi dan Teknologi Informasi*, 6(1), 65-74. doi:10.23917/komuniti.v6i1.2944
- Syaban, R. M., & Bunyamin, H. (2015). Pengembangan sistem informasi pengelolaan surat masuk dan keluar berbasis web di dinas sosial tenaga kerja dan transmigrasi kabupaten garut menggunakan framework php. *Jurnal Algoritma*, 12(2), 301-311
- Wibisono, W., & Baskoro, F. (2016). Pengujian Perangkat Lunak dengan Menggunakan Model Behavior UML. *JUTI (Jurnal Ilmiah Teknologi Informasi)*, 1(1), 43-50. doi:10.12962/j24068535.v1i1.a95