

Pengujian Sistem Informasi Parkir PT KISP Berbasis Desktop dengan Metode Black-Box

Aldi Arista Ilham¹, Ammar Azmi², Ardian Rizky Ramadhani³, Dimas Fakhti Abeda Falah⁴, Aries Saifudin⁵

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia
e-mail: ¹aldiariesta30@gmail.com, ²ammarsentrajaya@gmail.com, ³ardianrizky76@gmail.com,
⁴dimasfatkhiabdfalah@gmail.com, aries.saifudin@unpam.ac.id

Submitted Date: December 24th, 2020

Reviewed Date: December 31st, 2020

Revised Date: January 10th, 2021

Accepted Date: June 15th, 2021

Abstract

PT KISP is a market in the Ciledug area, which requires a parking application, it is hoped that this parking implementation can solve the problem of user needs, namely helping the parking system to be more effective and efficient in terms of processing data and information. With this application, it is hoped that it will make it easier for users to make transactions that are easy and in a short time. To design and implement a reliable parking application that is protected from damage or errors, comprehensive testing is needed, such as the functional test, performance test and user interface test aspects, so testing is needed. Testing parking applications based on black box, software testing is useful to ensure whether the actual results match the expected results and to ensure the application is not damaged. Based on the black box testing that has been carried out, it includes input, process and output in the parking application.

Keywords: Testing; Parking Application; Black Box

Abstract

PT KISP adalah suatu pasar di daerah ciledug, yang membutuhkan aplikasi parkir, diharapkan dari implementasi parkir ini dapat menyelesaikan masalah kebutuhan user, yaitu membantu sistem parkir menjadi lebih efektif dan efisien memberikan kemudahan dalam hal proses data dan informasi. Dengan adanya aplikasi ini diharapkan memudahkan para user untuk transaksi yang mudah dan waktu singkat. Untuk merancang dan implementasi aplikasi parkir yang handal yang terhindar dari kerusakan ataupun *error*, diperlukan testing yang komprehensif seperti aspek *functional test*, *performance test* dan *user interface test*, untuk itu di butuhkan pengujian. Pengujian aplikasi parkir berdasarkan black box, pengujian perangkat lunak berguna untuk memastikan apakah hasil aktual sesuai dengan hasil yang diharapkan dan untuk memastikan aplikasi tidak ada kerusakan. Berdasarkan pengujian black box yang telah dilakukan meliputi *input*, *proses* dan *output* pada aplikasi parkir.

Kata kunci: Testing; Aplikasi Parkir; Black-Box

1. Pendahuluan

Seiring dengan perkembangan zaman, banyak masyarakat yang menggunakan kendaraan seperti motor dan mobil sehingga menyebabkan lahan parkir yang tidak seimbang antara perkembangan kendaraan bermotor dengan lahan parkir. Oleh karena itu, informasi mengenai ketersediaan ruang parkir menjadi sangat penting bagi setiap pengendara yang akan memarkirkan kendaraannya.

Teknologi informasi telah berkembang dengan pesat. Teknologi informasi ditujukan untuk membantu pekerjaan dengan menyediakan informasi dan melakukan berbagai tugas yang berhubungan dengan pengolahan informasi (Hanum & Saifudin, 2019).

Sistem informasi digunakan untuk mengolah data yang diterima menjadi informasi yang dapat digunakan untuk mengambil keputusan (Andry & Loisa, The E-Commerce Potential for Home-Based Businesses: A Case Study, 2016). Misalnya data

tempat parkir yang kosong diinformasikan ke pengemudi atau pengatur tempat parkir untuk menunjukkan tempat parkir yang kosong. Penerapan sistem informasi tidak terpaku terhadap pekerjaan yang besar atau rumit, tetapi pekerjaan atau sesuatu yang bersifat sehari-hari yang awalnya bersifat manual dapat menerapkan sistem informasi. Penerapan sistem informasi terhadap parkir merupakan salah contohnya, Sistem Parkir yang telah menerapkan sistem informasi memberikan perubahan yang sangat besar. Perubahan yang terjadi di saat jalannya parkir yang masih bersifat manual dilakukan perubahan menjadi otomisasi, di mana aplikasi yang menjadi hasil penerapan sistem informasi terlibat dalam semua proses kegiatan parkir. Semua proses dari pencatatan informasi kendaraan, perhitungan parkir, dan pembayaran parkir terjadi dalam aplikasi yang telah dirancang tersebut.

Sistem informasi yang baik dalam hal penggunaannya haruslah menampilkan informasi yang akurat, oleh karena itu dalam proses pembuatannya perlu dilakukan serangkaian pengujian (testing). Pengujian merupakan tahapan penting yang harus dilakukan untuk memberikan jaminan terhadap kualitas perangkat lunak yang dikembangkan (Muslimin, et al., 2020). Pengujian perangkat lunak memiliki peranan penting dalam suatu sistem informasi, dengan pengujian ini dapat diketahui Galat atau Error yang akan muncul pada perangkat lunak (Ahrizal, Miftah, Kurniawan, Zaelani, & Yulianti, 2020). Sehingga sebelum diluncurkan, pengujian kualitas suatu aplikasi sangat dibutuhkan. Pengujian sangat diperlukan dalam sistem informasi untuk memastikan aplikasi yang dikembangkan telah memenuhi persyaratan teknis dan bisnis yang diharapkan sebelum diserahkan kepada pengguna (customer) (Febrian, Ramadhan, Faisal, & Saifudin, 2020). Tujuannya agar pelanggan yakin akan kelayakan aplikasi tersebut dan juga agar aplikasi tersebut bisa berjalan dengan lancar dan memenuhi kinerja yang dibutuhkan pelanggan.

2. Metodologi

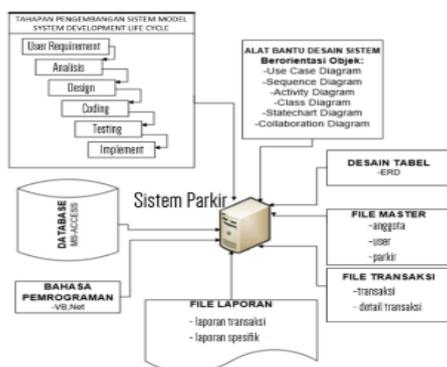
Metode yang digunakan yaitu SDLC. Metode System Development Life Cycle atau sering disingkat dengan SDLC merupakan pengembangan yang berfungsi sebagai sebuah mekanisme untuk mengidentifikasi perangkat lunak/tahapan dalam SDLC yang dikemukakan oleh para ahli memiliki perbedaan yang beragam sesuai dengan pemahamannya masing-masing, sehingga tahapan yang dilakukan oleh peneliti

memiliki perbedaan serupa dengan pemahaman yang kami miliki. Oleh karena itu, penelitian memberikan penjelasan tahapan yang telah dibuat, yaitu:

- a. *User Requirement*, mengumpulkan beberapa informasi mengenai kebutuhan user yang berguna untuk mengetahui dan membantu perancangan terhadap aplikasi yang akan dibuat. Informasi tersebut dapat berupa kebutuhan fungsi dalam aplikasi, masalah dan solusi yang dapat dicapai dari perancangan aplikasi, dan lainnya. Hasil keluaran dari tahap ini dapat berupa *Business Requirement Statement*.
- b. *Analysis*, memungkinkan untuk membuat gambaran terhadap logika perancangan aplikasi yang diinginkan. Gambaran logika dapat berupa struktur dan berorientasi objek, di mana struktur berarti menggunakan *Data Flow Diagram*, dan berorientasi objek menggunakan *Unified Modeling Language* (UML). Peneliti memilih perancangan berorientasi objek sehingga hasil keluaran dari tahap ini adalah UML.
- c. *Design*, menjelaskan bagaimana kami membuat gambaran terhadap rancangan aplikasi yang dibuat. Gambaran tersebut dapat berupa *User Interface* dan desain terhadap database yang akan digunakan dalam aplikasi. Pada tahap ini peneliti melakukan hasil keluaran adalah desain user interface dan desain rancangan database.
- d. *Coding*, tahapan di mana memulai untuk membuat kode pemrograman terhadap pembangunan aplikasi yang telah ditentukan. Hasil keluaran dari tahap ini adalah aplikasi selesai dibuat dan siap untuk dilakukan tahap pengujian.
- e. *Testing*, yaitu tahapan pengujian suatu aplikasi yang telah selesai dibuat. Pada penelitian ini digunakan metode *black box* testing untuk pengujiannya. Pengujian Black Box yang memiliki arti bahwa pengujian ini hanya memeriksa suatu perangkat lunak dari hasil eksekusinya, tanpa harus tahu mengetahui kode program dan hanya memperhatikan fungsionalnya saja (Nugraha, Aditama, Arrofi, Ahmad, & Yulianti, 2020). Pengujian harus dirancang secara sistematis agar dapat menemukan kesalahan memperbaikinya dalam waktu yang singkat (Maulana, Kurniawan, Keumala, Sukma, & Saifudin, 2020). Pengujian yang dilakukan tidak lengkap dan efektif dapat menimbulkan masalah dan menyebabkan kerugian ketika software digunakan (Rosalina, Rassi, Hadi, Ubaidillah, & Desyani, 2020).

Dengan *User Acceptance Testing* (UAT) dan *Coding Phase Defect* pengujian akan dilakukan, UAT dan *Coding Phase Defect* adalah hasil keluaran pada tahap pengujian tersebut. Pada tahap pengujian kita dapat mengetahui dan mengidentifikasi kesalahan yang mungkin terjadi dengan mengidentifikasi perbedaan antara hasil yang diharapkan dengan hasil sebenarnya (Pratama, Ristiano, Prayogo, Nasrullah, & Saifudin, 2020).

- f. Implementasi, yaitu tahapan di mana sistem/aplikasi siap untuk diinstalasi dan digunakan oleh pengguna. Fase implementasi adalah fase yang menyediakan dokumentasi dan persyaratan pelatihan bagi pengguna untuk menggunakan sistem secara efektif (Andry, Agung, & Erlyana, 2016).



Gambar 1. Kerangka Berpikir

Pada Gambar 1. Kerangka Berpikir, dijelaskan bahwa semua bagian dari struktur pemikiran untuk perancangan aplikasi sistem parkir dibuat menjadi satu gambaranya itu kerangka berpikir seperti gambaran di atas ini. Kerangka berpikir ini diberikan untuk memberikan gambaran jelas terhadap semua metode, database, file, dan lainnya yang digunakan untuk membangun aplikasi parkir pasar PT KISP.

3. Hasil dan Pembahasan

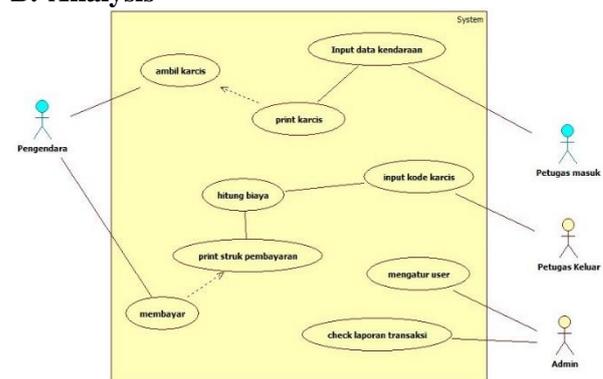
A. Requirement

Key Requirements dari aplikasi parkir adalah yang telah dikumpulkan, perancangan akan membuat pembagian sebanyak 3 modul dalam aplikasi sistem parkir tersebut, yaitu:

- a. Modul Admin, pada modul admin, diberikan kekuasaan untuk melakukan pengaturan user, dan melihat secara langsung laporan transaksi (pendapatan transaksi parkir) yang sesuai dengan hari, minggu, bulan, dan tahun yang diinginkan, berserta laporan spesifik terhadap jenis kendaraan yang diinginkan.

- b. Modul Proses Masuk Parkir, pada sistem masuk kendaraan, berisi bagian untuk memasukkan data kendaraan yang akan dilakukan petugas.
- c. Modul Proses Keluar Parkir, pada sistem keluar kendaraan, berisi data kendaraan yang akan ditampilkan dengan melakukan input terhadap plat nomor kendaraan yang melakukan parkir. Sistem memiliki fungsi penghitungan biaya tarif parkir dan memasukkan pembayaran, serta print struk pembayaran.

B. Analisis



Gambar 2: Use Case Diagram Sistem Parkir

Pada Gambar 2 diperlihatkan bahwa *use case* yang dibuat memiliki 4 aktor yaitu pengendara, petugas masuk, petugas keluar dan admin. Aktor tersebut memiliki masing-masing case yang dapat dilakukan, misalnya seperti pengendara yang dapat melakukan case ambil karcis ketika case print karcis telah dilakukan oleh petugas masuk. Pengendara dapat melakukan case pembayaran atas parkir jika case print struk pembayaran telah dilakukan oleh petugas keluar. *User Acceptance Testing* yang dibuat untuk aplikasi parkir Mall ABC memiliki 3 modul yaitu: Modul Admin, Modul Proses Masuk Parkir dan Modul Proses Keluar Parkir.

Pengujian *black-box*, juga dapat dikatakan sebagai pengujian perilaku, yang mana pengujian *black-box* berfokus kepada persyaratan-persyaratan fungsi perangkat lunak ataupun aplikasi. Pengujian black box memiliki arti bahwa pengujian yang dilakukan hanya mengambil hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak (Krismadi, et al., 2019). Software harus diprediksi dan konsisten, Membuat satu set data atau skenario test untuk data, program harus berjalan dengan benar dan dianggap program yang sukses.

C. Desain

Desain *interface* tidak data ditampilkan

keseluruhan dari gambar, karena keterbatasan halaman dari paper yang akan disubmit. User Interface halaman login, User dapat melakukan login pada halaman untuk menuju pada halaman berikutnya yang akan ditampilkan. User dibagi menjadi 3 jenis yaitu admin, petugas masuk dan petugas keluar. Setiap user memiliki halaman yang berbeda setelah melakukan login, sehingga informasi penting yang bersifat rahasia dapat tetap terjaga oleh admin saja. Pengisian password diberikan karakter (*) untuk menjaga kerahasiaan username dan password yang dimiliki oleh user, dan validasi perbedaan password dan jenis user terhadap login telah dibuat dalam halaman ini.

Halaman Home Admin hanya dapat ditampilkan setelah user berjenis admin telah melakukan login. Pada halaman ini admin dapat memilih untuk masuk pada halaman selanjutnya yaitu pengaturan user, laporan transaksi, dan pendaftaran member. Logout berfungsi untuk mengembalikan user kepada halaman login atau keluar dari halaman home admin. Penjelasan terhadap halaman yang dapat dituju oleh admin pada halaman ini dapat dilihat pada bagian selanjutnya.

Halaman Pengaturan User memberikan user berjenis admin hak untuk membuat, mengubah, dan menghapus user yang akan menggunakan aplikasi parkir ini. Pembuatan user harus dilakukan dengan mengisi username yang memiliki validasi tanpa spasi dan batasan hanya 10 karakter, nama yang memiliki validasi untuk tidak memasukkan angka, password dan tipe user yang dapat dipilih yaitu admin, petugas masuk, dan petugas keluar. Setelah pengisian maka user admin dapat menekan tombol "Insert" dan data informasi user yang telah dibuatkan ditampilkan pada data grid view dibagian bawah. Pengeditan "Edit" dan penghapusan "Delete" dapat dilakukan ketika user admin mengklik salah satu record dalam data grid view misalnya username atau password, kemudian data record akan muncul semuanya pada bagian diatas dan user dapat menekan tombol "Edit" atau "Delete" sesuai yang diinginkan. Ketika data record di data grid view dipilih atau melakukan klik, maka tombol "Insert" akan dimatikan.

Halaman Pendaftaran Member memberikan user admin akses untuk memasukkan sejumlah member atau anggota yang telah mendaftar dalam parkir pasar KISP. Bukti transaksi pembayaran keanggotaan seperti nomor plat, jenis kendaraan, tanggal register dan tanggal expired akan dimasukkan pada halaman ini. Memasukkan dilakukan dengan mengisi nomor plat dengan

validasi huruf karakter kapital dan batasan hanya 10 karakter, jenis kendaraan mobil atau motor, tanggal register, dan tanggal *expire*. Setelah pengisian dilakukan user admin dapat menekan tombol "Generate" untuk memasukkan data tersebut, lalu data akan ditampilkan pada data grid view bagian kanan.

Pengisian nomor plat yang telah terdaftar akan memberikan validasi untuk menjalankan fungsi pengubahan data anggota tersebut dan karena fungsi pengubahan ini maka tombol "Edit" tidak perlu dimasukkan kembali. Setiap user admin memasuki halaman ini, fungsi validasi keanggotaan akan dijalankan untuk memeriksa apakah tanggal *expire* telah dilewati atau tidak, jika tanggal *expire* telah dilewati maka secara langsung fungsi penghapusan data keanggotaan akan dilakukan dan nomor plat yang dihapus dari keanggotaan akan kembali menjadi status pengunjung. "Tombol Kembali" akan membawa user admin kembali menuju halaman home admin.

Pada halaman Laporan Admin atau Laporan Transaksi sesuai yang ditampilkan pada halaman home admin, user admin dapat menampilkan berbagai laporan seperti laporan transaksi dan laporan spesifik. Pengisian diperlukan untuk menampilkan laporan, yaitu pengisian terhadap tanggal awal, tanggal akhir, dan jenis kendaraan. Ketika jenis kendaraan yang dipilih ada motor atau mobil, maka laporan spesifik akan ditampilkan. Laporan transaksi akan ditampilkan jika jenis kendaraan yang dipilih adalah. Semua "Setelah pengisian maka user admin dapat menekan tombol "Generate" sehingga laporan yang diinginkan dapat ditampilkan. Tombol "Kembali" akan membawa user admin kembali menuju halaman home admin. Penjelasan terhadap gambaran tampilan laporan akan dijelaskan selanjutnya.

Pada halaman Laporan Transaksi atau Laporan Pendapatan, user admin dapat melihat tampilan mengenai keseluruhan pendapatan sesuai dengan jarak tanggal yang diinginkan yaitu hari, bulan, atau tahun. Halaman ini dibuat dengan Crystal Report yang ditampilkan setelah menekan tombol "Generate" pada halaman laporan admin sebelumnya. Penggunaan parameter seperti tanggal awal dan tanggal akhir yang didapatkan dari halaman yang berbeda pada crystal report sangat dibutuhkan, karena dengan parameter tersebut maka data yang terkait dengan memasukkan tanggal dapat ditampilkan sesuai yang diinginkan.

Halaman Laporan Spesifik memberikan user admin tampilan mengenai keseluruhan pendapatan spesifik jenis kendaraan yang diinginkan sesuai

dengan jarak tanggal yang diinginkan yaitu hari, bulan, atau tahun. Pada halaman ini laporan spesifik yang ditampilkan adalah jenis kendaraan motor yang dibuat dengan Crystal Report yang ditampilkan setelah menekan tombol "Generate" pada halaman laporan admin sebelumnya. Jumlah motor ditampilkan sesuai dengan jumlah nomor kendaraan yang berjenis motor melakukan transaksi parkir pada pasar KISP. Penggunaan parameter dilakukan sama seperti laporan transaksi sebelumnya, tetapi dengan tambahan parameter jenis kendaraan sehingga data transaksi yang ditampilkan dapat menjadi spesifik kendaraan yang diinginkan.

D. Coding

Pada aplikasi parkir ini, proses *development* atau *coding* menggunakan software Visual Basic dotnet yang terdapat pada Microsoft Visual Studio

2012.

E. Testing Black Box

Aliran proses modul Admin yaitu mengatur user (username, password, nama, dan tipe user) dan melihat laporan transaksi dan laporan spesifik. Aliran proses masuk parkir modul ini adalah memasukkan nomor plat dan jenis kendaraan, generate kode karcis, validasi keanggotaan, validasi kode karcis, validasi nomor plat kendaraan, simpan data parkir kendaraan dan print karcis parkir. Aliran proses keluar parkir modul ini adalah memasukkan kode karcis atau nomor plat, validasi data parkir, menampilkan data parkir, menghitung biaya tarif parkir, memasukkan pembayaran, validasi pembayaran, simpan data, transaksi dan print struk pembayaran. Keseluruhan modul tersebut dilakukan pengujian black box dengan skenario seperti ditampilkan pada Tabel 1.

Tabel 1: User Acceptance Test

| No | Use case / Test case | Pre-condition | Test steps | Expected Result | Actual Result |
|-----------------------------------|---|--|---|---|-----------------------------------|
| Modul ADMIN | | | | | |
| A-01 | Menguji button login dengan akun admin pada halaman Login | Akun Petugas Masuk telah dibuat dan dimiliki oleh user textbox Username, textbox Password, button Login | 1. Ketik 'aldo1' pada textbox username 2. Ketik '123' pada extbox password 3. Tidak menekan ENTER pada keyboard setelah mengisi textbox password 4. Klik button Login | Berhasil menampilkan form Home Admin menggunakan akun tipe admin | Admin dapat melakukan login |
| Modul Proses Masuk Parkir | | | | | |
| PM-01 | Menguji login dengan akun petugas masuk parkir | Akun Petugas Masuk telah dibuat dan dimiliki oleh user textbox Username, textbox Password, button Login | 1. Ketik 'chan2' pada textbox username 2. Ketik '123' pada textbox password 3. Tidak menekan ENTER pada keyboard setelah mengisi textbox password 4. Klik button Login | Berhasil menampilkan form Sistem Masuk menggunakan akun tipe user Petugas Masuk | Petugas petugas masuk dapat login |
| Modul Proses Keluar Parkir | | | | | |
| PK-01 | Menguji login dengan akun petugas keluar parkir | Akun Petugas Keluar telah dibuat dan dimiliki oleh user textbox Username, textbox Password, button Login | 1. Ketik 'stev3' pada textbox username 2. Ketik '123' pada textbox password 3. Tidak menekan ENTER pada keyboard setelah mengisi textbox password 4. Klik button Login | Berhasil menampilkan form Sistem Keluar menggunakan akun tipe user Petugas Keluar | Petugas Keluar dapat login |

Tabel 2: Coding Phase Defect Aplikasi Parkir Modul Admin

| No Modul | Skenario Testing |
|----------|--|
| A-02 | Menguji fungsi button Pengaturan User pada halaman home admin |
| A-03 | Pengujian button insert pada halaman pengaturan user |
| A-04 | Pengujian validasi button Insert pada halaman user |
| A-05 | Menguji validasi fungsi button Insert pada halaman Pengaturan User dengan username yang sama |
| A-06 | Pengujian button Edit pada halaman User |

| | |
|------|--|
| A-07 | Menguji fungsi button Delete pada halaman Pengaturan User |
| A-08 | Menguji fungsi button Reset pada halaman Pengaturan User |
| A-09 | Menguji fungsi button Kembali pada halaman Pengaturan User |
| A-10 | Menguji fungsi button Laporan Transaksi pada halaman Home Admin |
| A-11 | Menguji fungsi button Generate pada halaman Laporan Transaksi dengan memilih jenis kendaraan Mobil |
| A-12 | Menguji fungsi button Generate pada halaman Laporan Transaksi dengan memilih jenis kendaraan Motor |

| | |
|------|--|
| A-13 | Menguji fungsi button Generate pada halaman Laporan Transaksi dengan memilih jenis kendaraan Semua |
| A-14 | Menguji validasi fungsi button Generate pada halaman Laporan Transaksi |
| A-15 | Menguji fungsi button Kembali pada halaman Laporan Transaksi |
| A-16 | Menguji fungsi button Logout pada halaman Home Admin |
| A-17 | Menguji fungsi button Exit pada halaman Login setelah melakukan logout menggunakan user admin |

4. Kesimpulan

Berdasarkan implementasi pengujian dan analisisnya dibuat kesimpulan sebagai berikut:

- Berdasarkan aspek fungsi testing yang meliputi tiga proses yaitu input, proses dan output pada aplikasi parkir meliputi tiga pembahasan yaitu pembahasan admin telah berhasil mengatur user dan mengecek laporan transaksi. Skenario secara keseluruhan telah sesuai dengan *User Acceptance Test* (UAT) yang telah dilakukan
- Pengembangan parkir pasar PT KISP memberikan kemudahan kepada user petugas, serta memberikan efektifitas dan efisiensi dalam setiap prosesnya.
- Sistem parkir telah diuji dan diimplementasi dapat mengelola semua informasinya.

5. Saran

Berikut ini beberapa saran penelitian lanjutan:

- Untuk peneliti dan pengembang berikutnya agar dapat memodifikasi databasenya agar dapat memuat banyak transaksi dan memperluas jaringannya.
- Dengan adanya penelitian ini diharapkan menjadi acuan untuk peneliti dan pengembang selanjutnya.

Referensi

- Ahrizal, D., Miftah, M. K., Kurniawan, R., Zaelani, T., & Yulianti, Y. (2020). Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing. *Jurnal Informatika Universitas Pamulang*, 5(1), 73-77. doi:10.32493/informatika.v5i1.4338
- Andry, J. F., & Loisa, J. (2016). The E-Commerce Potential for Home-Based Businesses: A Case Study. *Jurnal Ilmiah FIFO*, 8(2), 139-146.
- Andry, J. F., Agung, H., & Erlyana, Y. (2016). Management Information System for Order Fulfillment: A Case Study. *Proceeding of 9th International Seminar on Industrial Engineering and Management*, (pp. 1-8). Padang.
- Febrian, V., Ramadhan, M. R., Faisal, M., & Saifudin, A. (2020). Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox. *Jurnal Informatika Universitas Pamulang*, 5(1), 61-66. doi:10.32493/informatika.v5i1.4340
- 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
- Krismadi, A., Lestari, A. F., Pitriyah, A., Mardangga, I. W., Astuti, M., & Saifudin, A. (2019). Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 2(4), 155-161.
- Maulana, A., Kurniawan, A., Keumala, W., Sukma, V. R., & Saifudin, A. (2020). Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Metode Equivalents Partitions (Studi Kasus: PT Arap Store). *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3(1), 50-56. doi:10.32493/jtsi.v3i1.4307
- Muslimin, D. B., Kusmanto, D., Amilia, K. F., Ariffin, M. S., Mardiana, S., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Sistem Informasi Akademik Menggunakan Teknik Equivalence Partitioning. *Jurnal Informatika Universitas Pamulang*, 5(1), 19-25. doi:10.32493/informatika.v5i1.3778
- Nugraha, B. F., Aditama, F., Arrofi, M., Ahmad, S. U., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Penghitungan Parkir Swalayan ADA Menggunakan Teknik Equivalence Partitions. *Jurnal Informatika Universitas Pamulang*, 5(2), 146-151. doi:10.32493/informatika.v5i2.5350
- Pratama, B. P., Ristiano, I. B., Prayogo, I. A., Nasrullah, & Saifudin, A. (2020). Pengujian Perangkat Lunak Sistem Informasi Penilaian Mahasiswa dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing. *Journal Of Artificial Intelligence And Innovative Applications*, 32-36.
- Rosalina, A., Rassi, A. A., Hadi, G. Y., Ubaidillah, R., & Desyani, T. (2020). Pengujian Black Box pada Sistem Informasi Penjualan HI Shoe Store Menggunakan Teknik Equivalence Partitions. *Jurnal Informatika Universitas Pamulang*, 5, 26-29. doi:10.32493/informatika.v5i1.3964