

Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan

Adi Krismadi¹, Ana Fisca Lestari², Anisah Pitriyah³, I Wayan Putra Ardhie Mardangga⁴,
Muji Astuti⁵, Aries Saifudin⁶

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia
E-mail: ¹adi.krismadi@gmail.com, ²anafiscalestari.afl@gmail.com, ³affitriyah@gmail.com,
⁴ardhiee27@gmail.com, ⁵mujiastuti1202@gmail.com, ⁶aries.saifudin@unpam.ac.id

Abstract

To guarantee the application (software) Selection Promotion Preferment is free from errors, testing is needed to test the quality of the application. This application has a problem when delete data, then the application will be tested with the Equivalence Partition-based Black Box Method thoroughly regarding the use, benefits, and results found from the use of the software. In testing the software Selection Promotion Preferment using Weight Product Method consists of 6 forms. Equivalence Partition-based Black Box method tests the quality of the application that will be carried out by testing software documentation with the discovery of errors on each form that is divided into three error models, namely errors in Function, Data Structure and Interface. The results of this test, to ensure the quality of the Selection Promotion Preferment application is free from errors by finding unintentional errors in the Selection Promotion Preferment application.

Keywords: Testing, Software, Black Box, Equivalence Partition

Abstrak

Untuk menjamin aplikasi (software) Seleksi Promosi Kenaikan Jabatan bebas dari kesalahan maka perlu dilakukan pengujian untuk menguji kualitas aplikasi. Aplikasi ini memiliki masalah ketika akan menghapus data, maka aplikasi akan diuji dengan Metode *Black Box* berbasis *Equivalence Partition* secara menyeluruh berkenaan dengan penggunaan, manfaat, dan hasil yang didapati dari pemanfaatan perangkat lunak tersebut. Dalam pengujian perangkat lunak Seleksi Promosi Kenaikan Jabatan menggunakan Metode *Weight Product* terdiri dari 6 form. Metode *Black Box* berbasis *Equivalence Partition* menguji kualitas aplikasi yang akan dilakukan dokumentasi pengujian perangkat lunak dengan ditemukannya kesalahan pada setiap form yang dibagi menjadi tiga model kesalahan, yaitu kesalahan pada fungsi, struktur data dan *interface*. Hasil dari pengujian ini, untuk menjamin kualitas aplikasi Seleksi Promosi Kenaikan Jabatan bebas dari kesalahan dengan menemukan adanya kesalahan yang tidak disengaja pada aplikasi Seleksi Promosi Kenaikan Jabatan.

Kata Kunci: Pengujian, Software, Black Box, Equivalence Partition

1. Pendahuluan

Pengujian adalah serangkaian aktifitas yang direncanakan dan teratur untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu rangkaian di mana dapat menempatkan desain khusus uji coba yang spesifik. Kepuasan pelanggan tergantung pada kualitas perangkat lunak dan kualitas sejumlah perangkat lunak perlu dijaga dengan beberapa alasan (Cholifah, Yulianingsih, & Sagita, 2018).

Aplikasi Seleksi Promosi Kenaikan Jabatan merupakan cara penghitungan untuk menentukan nama pegawai yang akan naik jabatan. Jadi, Sistem Penunjang Keputusan (SPK) yang dikembangkan bertujuan untuk menghasilkan keluaran yang dapat memberikan pertimbangan kepada Kepala Perusahaan untuk menentukan siapa pegawai yang akan naik jabatan dari beberapa kandidat pegawai dengan menggunakan Metode *Weight Product*. Dalam aplikasi Seleksi Promosi Kenaikan Jabatan ini terdapat 5 kriteria yang digunakan dalam penilaian, yaitu absensi,

loyalitas, masa kerja, nilai tes dan keterampilan (Egi, Handoko, & Rina, 2016). Tim seleksi akan memberikan bobot untuk masing-masing kriteria dan menentukan atribut untuk setiap kriteria untuk selanjutnya dilakukan penghitungan dengan menggunakan metode *Weight Product*.

Pada aplikasi Seleksi Promosi Kenaikan Jabatan terdapat kesalahan pada saat menghapus data sehingga data tidak dapat terhapus dan merugikan Kepala Perusahaan dalam menentukan pegawai yang tepat untuk naik jabatan. Maka, aplikasi ini perlu diuji untuk menemukan kesalahan yang tidak disengaja, pengujian dilakukan menggunakan metode *Black Box* yang memiliki arti bahwa pengujian yang dilakukan hanya mengambil hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dapat disamakan bahwa kita hanya memeriksa sesuatu dari luarnya saja. *Black Box* hanya mengevaluasi bagian dari tampilan antarmuka (*interface*) dan fungsional tanpa mengetahui apa yang terjadi dalam proses detailnya yang berarti bahwa hanya dapat mengetahui masukan dan keluaran saja. Maka dilakukan langkah pemeriksaan untuk memastikan bahwa data tersebut telah sesuai dengan yang ditetapkan dan bertujuan untuk memastikan bahwa data akan dimasukkan ke dalam basis data itu telah diketahui dan dapat dijelaskan kebenaran datanya (MZ, 2016).

Untuk menjamin aplikasi Seleksi Promosi Kenaikan Jabatan bebas dari kesalahan, maka perlu dilakukan pengujian. Pada penelitian ini diusulkan pengujian dengan menggunakan metode *Black Box*. *Black Box* adalah teknik pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji dapat mendefinisikan kumpulan kondisi masukan dan melakukan pengetesan pada spesifikasi fungsional program (Hidayat & Muttaqin, 2018). Tujuan *Black Box Testing* untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data keluaran telah berjalan sebagaimana yang telah diharapkan dan apakah informasi yang disimpan serta *eksternal* selalu dijaga kemutakhirannya (Maharani & Merlina, 2014). *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih sebagai pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black Box Testing* mengidentifikasi jenis kesalahan dalam beberapa kategori antara lain fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, kesalahan performansi

(*performance errors*) dan kesalahan inisialisasi maupun terminasi (Mustaqbal, Firdaus, & Rahmadi, 2015).

Metode *Equivalence Partitions* adalah metode pengujian *Black Box* yg memecah atau membagi domain masukan dari program ke dalam kelas-kelas data sehingga *Test Case* dapat diperoleh. Perancangan *Test Case Equivalence Partitions* berdasarkan evaluasi kelas *Equivalence* untuk kondisi masukan yang menggambarkan kumpulan keadaan yang *valid* atau tidak. Kondisi masukan dapat berupa nilai *numeric*, *range* nilai, kumpulan nilai yang berhubungan atau kondisi *Boolean*. Semua tahapan akan dibahas pada bab selanjutnya.

2. Metodologi

Pengujian merupakan salah satu tahapan yang harus dilalui untuk menghasilkan perangkat lunak atau sistem yang berkualitas. Melalui pengujian, penguji dapat mengetahui kekurangan dan kesalahan yang terdapat didalam sebuah sistem sehingga dapat segera memperbaikinya. Tujuannya adalah menyesuaikan fungsi-fungsi dengan spesifikasi yang diinginkan (Andriansyah, 2018). Pengujian *software* sangat diperlukan untuk memastikan aplikasi / *software* yang sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Penguji *software* harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. (Shi Mingtao, 2010).

Pengujian *Black Box* berusaha untuk menemukan klasifikasi kesalahan dalam beberapa kategori, yaitu fungsi-fungsi yang salah atau hilang, kesalahan *interface*, kesalahan dalam struktur data atau akses database eksternal, kesalahan performa, kesalahan dan terminasi (Agarwal, Tayal, & Gupta, 2010). Dalam pengujian *Black Box Testing* digunakan alat untuk pengumpulan data yang disebut dengan *user acceptance test*, dokumen ini terdiri deskripsi indikator dari prosedur – prosedur pengujian fungsionalitas dari perangkat lunak. Dengan metode *Black Box Testing* ini, pengembang perangkat lunak Seleksi Promosi Kenaikan Jabatan dapat menguji keseluruhan fitur atau menu yang terdapat pada sistem tersebut (Setiyani, 2019). Metode *Equivalence Partitions* merupakan pengujian berdasarkan masukan setiap menu yang terdapat pada sistem seleksi pemenang tender, setiap menu masukan dilakukan pengujian melalui klasifikasi dan pengelompokan berdasarkan fungsinya (Hidayat & Muttaqin, 2018). Dalam

penelitian ini akan dilakukan beberapa tahapan. Pada tahapan pertama diawali dengan menentukan *Test Case* perangkat lunak yang akan diuji dengan metode *Equivalence Partitions* kemudian menginisialisasi standar *grade partition* masukan dan keluaran. Hal ini dilakukan untuk mendapatkan dataset berupa dokumentasi pengujian dengan metode *Equivalence Partitions* dan nilai tingkat efektifitas metode *Equivalence Partitions*.

Pada hasil pengujian terdapat tabel rancangan *Test Case* yang berfungsi untuk menyimpulkan apakah sistem berhasil dalam pengujian tipe tersebut atau tidak. Berikut terdapat penjelasan dari beberapa tabel Rancangan *Test Case* berdasarkan *Equivalence Partitions*.

Pada form pada Gambar 1 terdapat beberapa rencana pengujian. Rencana pengujian *ID* data akan valid jika *ID* diisi dengan memasukkan angka “123” yang terdiri dari 5 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika memasukkan abjad “abc”. Pada rencana pengujian Kriteria data akan valid jika Kriteria diisi dengan memasukkan abjad “abc” yang terdiri dari 25 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika Kriteria diisi dengan memasukkan angka “123” atau lebih dari 25 digit. Pada rencana pengujian Rentang Nilai data akan valid jika diisi dengan memasukkan angka “123” yang terdiri dari 3 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika Rentang Nilai diisi dengan memasukkan abjad “abc” atau lebih dari 3 digit. Pada rencana pengujian Satuan data

akan valid jika diisi dengan memasukkan abjad “abc” yang terdiri dari 10 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika Satuan diisi dengan memasukkan angka “123” atau lebih dari 10 digit. Pada rencana pengujian Bobot data akan valid jika diisi dengan memasukkan angka “123” yang terdiri dari 2 digit dan tidak boleh kosong. Sebaliknya, data tidak valid Bobot diisi dengan memasukkan abjad “abc” atau lebih dari 2 digit.

Gambar 1 Form Tambah Kriteria

Tabel 1 Rancangan Test Case Form Tambah Kriteria

Id	Deskripsi Pengujian	Hasil Yang Diharapkan
K01	Mengisi <i>ID</i> dengan “6”, Kriteria isi dengan “Kerjasama”, Rentang Nilai dengan “100”, Satuan isi dengan “Point” dan Bobot isi dengan “10” kemudian klik tombol Daftar	Sistem berhasil menerima penambahan data
K02	Mengisi <i>ID</i> dengan “abc123” atau tidak diisi, Kriteria isi dengan “123” atau “100” atau tidak diisi, Rentang Nilai dengan “abc123” atau tidak diisi, Satuan isi dengan “123” atau tidak diisi dan Bobot isi dengan “abc123” atau tidak diisi kemudian klik tombol Daftar	Sistem menolak untuk menyimpan data

Gambar 2 Form Edit Kriteria

Pada form pada Gambar 2 terdapat beberapa rencana pengujian. Rencana pengujian Kriteria data akan valid jika Kriteria diisi dengan memasukkan abjad “abc” yang terdiri dari 25 digit

dan tidak boleh kosong. Sebaliknya, data tidak valid jika Kriteria diisi dengan memasukkan angka “123” atau lebih dari 25 digit. Pada rencana pengujian Rentang Nilai data akan valid jika diisi dengan memasukkan angka “123” yang terdiri dari 3 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika Rentang Nilai diisi dengan memasukkan abjad “abc” atau lebih dari 3 digit. Pada rencana pengujian Satuan data akan valid jika diisi dengan memasukkan abjad “abc” yang terdiri dari 10 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika Satuan diisi dengan memasukkan angka “123” atau lebih dari 10 digit. Pada rencana pengujian Bobot data akan valid jika diisi dengan memasukkan angka “123” yang terdiri dari 2 digit dan tidak boleh kosong. Sebaliknya, data tidak valid Bobot diisi dengan memasukkan abjad “abc” atau lebih dari 2 digit.

Tabel 2 Rancangan Test Case Form Edit Kriteria

Id	Deskripsi Pengujian	Hasil Yang Diharapkan
EK 01	Mengisi Kriteria dengan “Kepemimpinan”, Rentang Nilai dengan “100”, Satuan isi dengan “Point” dan Bobot isi dengan “10” kemudian klik tombol Simpan	Sistem mampu menyimpan data kedalam database dan mengalami perubahan tampilan pada sistem
EK 02	Mengisi Kriteria dengan “123” atau “100” atau tidak diisi, Rentang Nilai dengan “abc123” atau tidak diisi, Satuan isi dengan “123” atau tidak diisi dan Bobot isi dengan “abc123” atau tidak diisi kemudian klik tombol Daftar	Sistem menolak untuk menyimpan data

Gambar 3 Form Tambah Alternatif

Pada form pada Gambar 3 terdapat beberapa rencana pengujian. Rencana pengujian ID data akan valid jika ID diisi dengan memasukkan angka “123” yang terdiri dari 5 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika memasukkan abjad “abc”. Pada rencana pengujian Alternatif data akan valid jika Alternatif harus diisi dengan memasukkan abjad “abc” yang terdiri dari 30 digit dan tidak boleh kosong. Begitupun sebaliknya data akan tidak valid jika Alternatif diisi dengan memasukkan angka “123” atau lebih dari 30 digit. Pada rencana pengujian C1, C2, C3, C4 dan C5 data akan valid jika diisi dengan memasukkan angka contohnya “123” yang terdiri dari 3 digit dan tidak boleh kosong. Begitupun sebaliknya data akan tidak valid jika C1, C2, C3, C4 dan C5 diisi dengan memasukkan abjad “abc” atau lebih dari 3 digit.

Tabel 3 Rancangan Test Case Form Tambah Alternatif

Id	Deskripsi Pengujian	Hasil Yang Diharapkan
A01	Mengisi <i>ID</i> dengan “11”, isi Alternatif dengan “Lulu”, isi C1 dengan “85”, isi C2 dengan “50”, isi C3 dengan “75”, isi C4 dengan “90”, isi C5 dengan “40” kemudian klik tombol Daftar	Sistem berhasil menerima penambahan data
A02	Mengisi Alternatif dengan “abc123” atau “123” atau tidak diisi dan C1, C2, C3, C4, C5 dengan “1000” kemudian klik tombol Daftar	Sistem menolak untuk menyimpan data

Gambar 4 Form Edit Alternatif

Berdasarkan form pada Gambar 4 terdapat beberapa rencana pengujian. Pada rencana pengujian Alternatif data akan valid jika Alternatif harus diisi dengan memasukkan abjad “abc” yang terdiri dari 30 digit dan tidak boleh kosong. Begitupun sebaliknya data akan tidak valid jika Alternatif diisi dengan memasukkan angka “123” atau lebih dari 30 digit. Pada rencana pengujian C1, C2, C3, C4 dan C5 data akan valid jika diisi dengan memasukkan angka “123” yang terdiri dari 3 digit dan tidak boleh kosong. Begitupun sebaliknya data akan tidak valid jika C1, C2, C3, C4 dan C5 diisi dengan memasukkan abjad “abc” atau lebih dari 3 digit.

Tabel 4 Perancangan Test Case Form Edit Alternatif

Id	Deskripsi Pengujian	Hasil Yang diharapkan
EA 01	Mengisi Alternatif dengan “Tasya”, isi C1 dengan “85”, isi C2 dengan “50”, isi C3 dengan “75”, isi C4 dengan “90”, isi C5 dengan “40” kemudian klik tombol Simpan	Sistem mampu menyimpan data kedalam database dan mengalami perubahan tampilan pada sistem.
EA 02	Mengisi Alternatif dengan “abc123” atau “123” atau tidak diisi dan C1, C2, C3, C4 C5 dengan “1000” kemudian klik tombol Simpan	Sistem menolak untuk menyimpan data

3. Hasil dan Pembahasan

Berdasarkan rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut.

Tabel 1 Rencana Pengujian

Id	Deskripsi Pengujian	Hasi Yang diharapkan	Hasil Pengujian	Kesimpulan
K0 1	Mengisi <i>ID</i> dengan “6”, Kriteria isi dengan “Kerjasama”, Rentang Nilai dengan “100”, Satuan isi dengan “Point” dan Bobot isi dengan “10” kemudian klik tombol Daftar	Sistem mampu menyimpan data kedalam database dan mengalami perubahan tampilan pada sistem	Tampilan halaman utama aplikasi Sistem Seleksi Pemenang Tender	Sesuai
K0 2	Mengisi <i>ID</i> dengan “abc123” atau tidak diisi, Kriteria isi dengan “123” atau “100” atau tidak diisi, Rentang Nilai dengan “abc123”	Sistem menolak untuk menyimpan data	Tampilan pesan peringatan data tidak	Sesuai

	atau tidak diisi, Satuan isi dengan "123" atau tidak diisi dan Bobot isi dengan "abc123" atau tidak diisi kemudian klik tombol Daftar		lengkap	
EK 01	Mengisi Kriteria dengan "Kepemimpinan", Rentang Nilai dengan "100", Satuan isi dengan "Point" dan Bobot isi dengan "10" kemudian klik tombol Simpan	Sistem mampu menyimpan data kedalam database dan mengalami perubahan tampilan pada sistem	Tampilan halaman utama aplikasi Sistem Seleksi Pemenang Tender	Sesuai
EK 02	Mengisi Kriteria dengan "123" atau "100" atau tidak diisi, Rentang Nilai dengan "abc123" atau tidak diisi, Satuan isi dengan "123" atau tidak diisi dan Bobot isi dengan "abc123" atau tidak diisi kemudian klik tombol Daftar	Sistem menolak untuk menyimpan data	Tampilan pesan peringatan data tidak lengkap	Sesuai
A0 1	Mengisi ID dengan "11", isi Alternatif dengan "Lulu", isi C1 dengan "85", isi C2 dengan "50", isi C3 dengan "75", isi C4 dengan "90", isi C5 dengan "40" kemudian klik tombol Daftar	Sistem berhasil menerima penambahan data	Tampilan data kriteria dan alternatif	Sesuai
A0 2	Mengisi Alternatif dengan "abc123" atau "123" atau tidak diisi dan C1, C2, C3, C4, C5 dengan "1000" kemudian klik tombol Daftar	Sistem menolak untuk menyimpan data	Tampilan pesan peringatan data tidak lengkap	Sesuai
EA 01	Mengisi Alternatif dengan "Tasya", isi C1 dengan "85", isi C2 dengan "50", isi C3 dengan "75", isi C4 dengan "90", isi C5 dengan "40" kemudian klik tombol Simpan	Sistem mampu menyimpan data kedalam database dan mengalami perubahan tampilan pada sistem.	Tampilan data kriteria dan alternatif	Sesuai
EA 02	Mengisi Alternatif dengan "abc123" atau "123" atau tidak diisi dan C1, C2, C3, C4 C5 dengan "1000" kemudian klik tombol Simpan	Sistem menolak untuk menyimpan data	Tampilan pesan peringatan data tidak lengkap	Sesuai

Jumlah Form yang diuji 6 form dengan form kriteria diuji sebanyak 5 kali, form edit kriteria diuji sebanyak 1 kali, form alternatif diuji sebanyak 10 kali, form edit alternatif diuji sebanyak 1 kali, form hapus kriteria diuji sebanyak 1 kali dan form hapus alternatif diuji sebanyak 1 kali dan. Jadi, total pengujian ada 19 kali. Jumlah form yang ditemukan *error* sebanyak 1 form dan 5 form tidak ditemukan *error*.

4. Kesimpulan

Pengujian perangkat lunak dengan menggunakan metode *Black Box* memberikan dokumentasi hasil pengujian yang

menginformasikan kesesuaian perangkat lunak yang diuji dengan spesifikasi yang telah ditentukan. Berdasarkan pengujian kualitas aplikasi Seleksi Promosi Kenaikan Jabatan, dapat diambil kesimpulan bahwa pengujian dengan metode *Black Box* berbasis *Equivalence Partitions* dapat membantu proses pembuatan *case* pengujian, uji kualitas dan menemukan kesalahan yang tidak disengaja. Dalam pengujian aplikasi Seleksi Promosi Kenaikan Jabatan ditemukan kesalahan dalam menghapus sebuah data yang sudah tersimpan. Pengujian ini menunjukkan bahwa terdapat celah *error* pada suatu form aplikasi Seleksi Promosi Kenaikan Jabatan.

5. Saran

Berdasarkan proses pengujian kualitas aplikasi Seleksi Promosi Kenaikan Jabatan, penulis memberikan saran pada penelitian selanjutnya untuk mengkaji lebih banyak sumber maupun referensi yang terkait agar hasil penelitian dapat lebih baik dan lebih lengkap lagi. Diharapkan pada penelitian selanjutnya lebih dipersiapkan dengan matang dalam aplikasi Seleksi Promosi Kenaikan Jabatan dan juga menggunakan metode pengujian lain agar hasilnya sesuai dengan yang diharapkan.

Referensi

- Agarwal, B., Tayal, S. P., & Gupta, M. (2010). *Software Engineering & Testing*. Boston: AuthorJones & Bartlett Learning.
- Andriansyah, D. (2018). Pengujian Kotak Hitam Boundary Value Analysis Pada Sistem Informasi Manajemen Konseling Tugas Akhir. STMIK Nusa Mandiri Jakarta. *Indonesia Journal on Networking and Security*, 7(1), 13-18.
- Cholifah, W., Yulianingsih, & Sagita, S. (2018). Pengujian Black Box Testing Pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *Jurnal String*, 3(2), 206-210.
- Egi, Handoko, & Rina, M. (2016). Sistem Pendukung Keputusan Kenaikan Jabatan Karyawan Plaza Asia Dengan Menggunakan Metode Weighted Product. *JSTMIK Tasikmalaya: CSRID Journal*, 8(2), 121-130.
- Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis. *Jurnal Teknik Informatika UNIS*, 6(1), 25-29.
- Komarudin, & M, K. M. (2016). Pengujian Perangkat Lunak Metode Black Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah. *Jurnal Mikrotik*, 6(3), 1-8.
- Maharani, M., & Merlina, N. (2014). Penerapan Metode Straight Selection Pada Sistem Parkir Universitas Bina Nusantara. *Jurnal Pilar Nusa Mandiri*, 10(1), 95-100.
- Mustaqbal, M., Firdaus, R., & Rahmadi, H. (2015). engujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, 1(3), 31-36.
- Setiyani, L. (2019). Pengujian Sistem Informasi Inventory Pada Perusahaan Distributor Farmasi Menggunakan Metode Black Box Testing. *Jurnal Ilmu Komputer & Teknologi Informasi*, 4(1), 1-9.
- Shi Mingtao. (2010). Software Functional Testing from the Perspective of Business Practice. *Computer and Information Science*, 3(4), 49-52.