

## Pengujian *Black Box* pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik *Equivalence Partitions*

Fadhila Cahya Ningrum<sup>1</sup>, Dandi Suherman<sup>2</sup>, Sita Aryanti<sup>3</sup>, Handika Angga Prasetya<sup>4</sup>, dan Aries Saifudin<sup>5</sup>

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia

E-mail: <sup>1</sup>fadhilacahyaningrum@gmail.com, <sup>2</sup>dandisuherman55@gmail.com,

<sup>3</sup>sitaaryanti1997@gmail.com, <sup>4</sup>handikaangga97@gmail.com, <sup>5</sup>aries.saifudin@unpam.ac.id

Submitted Date: December 18<sup>th</sup>, 2019

Reviewed Date: December 18<sup>th</sup>, 2019

Revised Date: January 2<sup>nd</sup>, 2020

Accepted Date: January 9<sup>th</sup>, 2020

### Abstract

*Testing on an application aims to check whether a program is running properly so that the program created will be a program that has good quality. The testing technique consists of Whitebox Testing and Blackbox Testing which are widely used by examiners is appropriate. Study the software that will be examined using Blackbox Testing is a best sales selection system consisting of 1 form that will be filled with sales data, as well as sales performance while in the company. The Best Sales Selection Application will be tested using Blackbox Testing where this test only aims to look the program is the same with the function desired by program without knowing the program code used. Types of testing of the various Blackbox methods one of which is the Equivalence Partitions used by the authors in this study. Equivalence Partitions technique is a test based on entering data on each form in the best sales selection system, each menu input will be tested and grouped according to function whether it is valid or invalid. So it is expected that with this test is the quality of the software produced in accordance with the expected function, and can provide benefits for the company.*

*Keywords: Application, Testing, Blackbox, Equivalence Partitions, Best Selection System*

### Abstrak

Pengujian pada suatu aplikasi bertujuan untuk memeriksa apakah sebuah program telah berjalan dengan semestinya atau masih ada kesalahan yang harus diperbaiki agar program yang dibuat akan menjadi program yang memiliki kualitas yang baik. Teknik pengujian terdiri dari *Whitebox Testing* dan *Blackbox Testing* yang banyak digunakan para penguji untuk memeriksa program telah sesuai atau tidak. Pada pengkajian ini perangkat lunak yang akan dikaji menggunakan *Blackbox Testing* adalah sebuah sistem seleksi sales terbaik terdiri dari 1 *form* yang akan diisi dengan data sales, serta kinerja sales selama di perusahaan. Aplikasi Seleksi Sales Terbaik tersebut akan diuji menggunakan *Blackbox Testing* di mana pengujian ini hanya bertujuan untuk melihat program tersebut apakah sesuai dengan fungsi yang diinginkan program tersebut tanpa mengetahui kode program yang dipakai. Jenis pengujian dari metode *Blackbox* beragam salah satunya adalah *Equivalence Partitions* yang dipakai oleh penulis dalam penelitian ini. Teknik *Equivalence Partitions* merupakan sebuah pengujian berdasarkan masukkan data pada setiap *form* yang ada pada sistem seleksi sales terbaik, setiap menu masukan akan dilakukan pengujian dan dikelompokkan berdasarkan fungsinya baik itu bernilai valid ataupun tidak valid. Sehingga diharapkan dengan adanya pengujian ini adalah kualitas *software* yang dihasilkan sesuai dengan fungsi yang diharapkan, serta dapat memberikan keuntungan bagi perusahaan.

Kata Kunci: Aplikasi, Pengujian, *Black Box*, *Equivalence Partitions*, Sistem Seleksi Terbaik

### 1. Pendahuluan

Pengujian pada sebuah program penting untuk dilakukan guna memeriksa semua kesalahan yang ada pada program tersebut agar tidak terjadi

kerugian yang akan ditimbulkan dari kesalahan tersebut, sehingga sangat perlu untuk dilakukan pengujian untuk mengurangi terjadinya kesalahan yang merugikan tersebut (Sethi, 2017). Hal yang

perlu diperhatikan dalam melakukan pengujian adalah pengujian harusnya dapat mendapatkan kesalahan yang mungkin sebelumnya tidak terdeteksi dan dikatakan berhasil pengujian tersebut apabila dapat memperbaiki kesalahan tersebut sehingga kualitas software akan lebih baik. Dalam pengujian hal yang perlu diperhatikan adalah perancangan yang baik agar dapat dengan mudah menemukan kesalahan pada saat pengujian agar dapat diperbaiki dengan cepat sehingga dapat menghemat waktu dalam pengujian.

Pemilihan sales terbaik dipilih berdasarkan kriteria tertentu, di antaranya adalah penjualan, komunikasi, inovasi, disiplin serta loyalitas. Perlu adanya sebuah sistem yang dapat mengatur agar data yang telah dimasukan akan lebih terstruktur. Pada penelitian ini sistem yang akan dikerjakan adalah program pemilihan sales terbaik digunakan untuk memudahkan dalam memasukan data-data. Sistem Penunjang Keputusan tersebut menggunakan sebuah metode yaitu metode hasil bobot terbaik (Mukhayaroh, 2016). Metode hasil bobot terbaik adalah sebuah metode yang digunakan untuk menjumlahkan nilai bobot yang ada yang didapat dari atribut yang digunakan yang akan dilanjutkan dengan proses perangkingan yang akan menjadi penentu apakah nilai alternatif yang digunakan sudah yang terbaik (Wahyudi, Suheri, & Nurhadian, 2015).

Berdasarkan dari observasi yang dilakukan dalam menentukan sales terbaik masih menggunakan cara yang manual yang cenderung tidak akurat khususnya bagi perusahaan. Dari permasalahan yang ada dibutuhkan sebuah aplikasi penunjang keputusan pemilihan sales terbaik yang diharapkan dengan adanya aplikasi tersebut akan dapat memberikan keuntungan bagi perusahaan (Sasmito, 2017).

Program yang akan dibuat pada pengkajian ini adalah program sistem seleksi sales terbaik adalah *Blackbox Testing*, di mana pengujian bertujuan melihat program tersebut sama dengan tugas program tersebut tanpa mengetahui kode program yang dipakai. Tahapan pertama pada pengujian dengan menggunakan *Blackbox Testing* adalah mengidentifikasi masukan lalu diuji agar kita mengetahui letak kesalahannya. Pengujian menggunakan *Blackbox Testing* merupakan sebuah pengujian yang digunakan untuk melengkapi pengujian sebelumnya yaitu *Whitebox Testing* agar aplikasi yang kita buat memiliki kualitas yang baik serta waktu yang digunakan akan lebih efektif, sehingga dapat menguntungkan bagi perusahaan (Mustaqbal, Firdaus, & Rahmadi, 2015).

Ada beberapa cara dalam menguji *Black Box Testing* salah satunya adalah menggunakan teknik *Equivalence Partitions*. *Equivalence Partitions* merupakan sebuah pengujian berdasarkan masukkan data pada setiap *form* yang ada pada sistem seleksi sales terbaik, setiap menu masukan akan dilakukan pengujian dan dikelompokkan berdasarkan fungsinya baik itu bernilai valid ataupun tidak valid (Hidayat & Muttaqin, 2018).

Dalam penelitian ini terdapat beberapa tahapan, di mana diawali dengan menentukan *Test Case* yang akan diuji, kemudian membaginya dengan beberapa partisi masukan dan keluaran. Hal ini dilakukan untuk mendapatkan dokumentasi pengujian (MZ, 2016).

## 2. Metodologi

Pengujian *software* sangat penting dilakukan karena setiap orang membuat kesalahan pada saat pembuatan *software*. Kesalahan pada masing-masing *software* akan berbeda pada masing-masing *software* (Akiladevi, Vidhupriya, & Sudha, 2018). Maka dari itu perlu dilakukan pengujian *software* untuk melakukan verifikasi dan validasi bahwa program dibuat sama dengan kebutuhan dari perusahaan. Apabila tidak sama dengan kebutuhan dari perusahaan, maka perlu dilakukan evaluasi agar dapat dilakukan perbaikan pada *software* tersebut (Ardana, 2019). Pengujian yang akan digunakan adalah *Blackbox* agar kualitas *software* lebih baik.

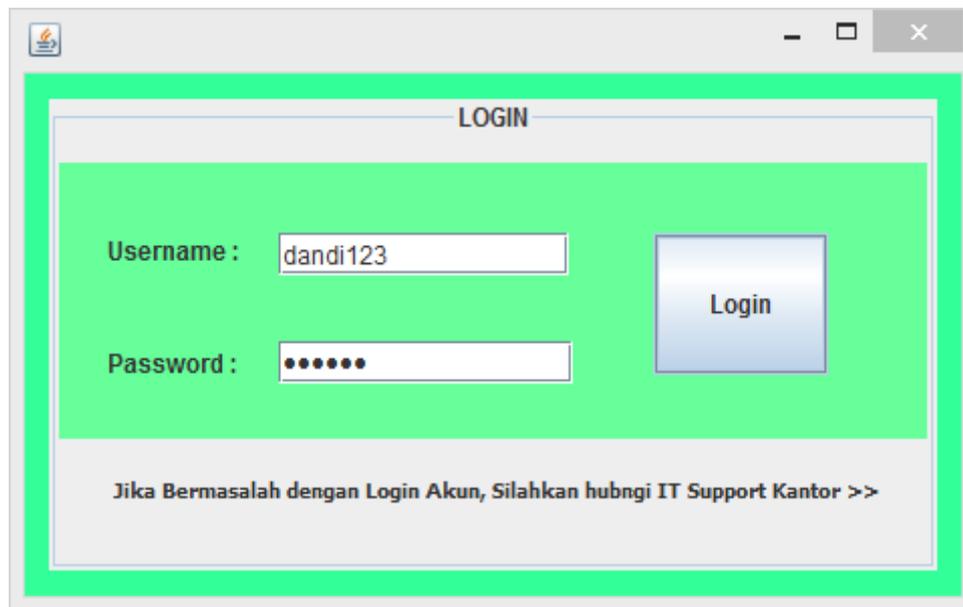
Metode *Blackbox Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai (Latif, 2015). Proses *Black Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan (Wahyudi, Utami, & Arief, 2016).

Salah satu jenis pengujian *Blackbox Testing* adalah dengan menggunakan teknik *Equivalence Partitioning* (EP) yang penulis gunakan untuk menguji masukan serta membagi masukan kedalam kelompok-kelompok berdasarkan fungsinya. Sehingga didapatkan sebuah *test case* yang akurat (Aristoteles, Wardiyanto, & Pratama, 2015). 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*.

Terdapat tabel rancangan *Test Case* yang berfungsi untuk melakukan pengecekan program

sudah sama dengan kebutuhan yang diinginkan atau masih perlu diperbaiki agar kualitas program yang dihasilkan akan lebih baik. Berikut ini penjelasan dari beberapa tabel Rancangan *Test Case* berdasarkan *Equivalence Partitions*.



Gambar 1 Tampilan *Form Login*

Pengujian ini dilakukan pada *Form Login* dengan memasukkan data yang tidak sesuai dengan *type* data. Rencana pengujian Username data akan valid jika Username diisi dengan memasukkan angka “123” atau huruf “abc” maupun gabungan dari angka atau huruf “abc123” dengan maksimal huruf 50. Begitupun sebaliknya data akan tidak valid jika Username diisi melebihi 50 huruf

maupun angka. Rencana pengujian Password data akan valid jika Password diisi dengan memasukkan angka “123” atau huruf “abc” maupun gabungan dari angka atau huruf “abc123” dengan maksimal huruf 20. Begitupun sebaliknya data akan tidak valid jika Password diisi dan data yang diisi melebihi 20 huruf maupun angka.

Tabel 1 Rancangan Test Case Form Login

Id	Deskripsi Pengujian	Hasil yang Diharapkan
L01	Mengisi Username dengan “dandi123”, Password diisi dengan “dan123” dengan ketentuan data yang diisi merupakan data yang ada di <i>database</i> kemudian klik tombol Login.	Akses Login berhasil, sistem akan menutup form Login dan menampilkan form Data.
L02	Mengisi Username dengan “abc123”, Password diisi dengan “abc123” dengan ketentuan data yang diisi merupakan data yang tidak ada atau belum tersimpan di <i>database</i> , kemudian klik tombol Login.	Akses Login Gagal, sistem tidak dapat membuka form Data dan tampilan tetap pada form Login.

Gambar 2 Tampilan *Form Data*

Pengujian berikutnya dilakukan pada *form Data* dengan memasukkan data yang tidak sesuai dengan *type data*. Rencana pengujian NIP data akan valid jika NIP diisi dengan memasukkan angka "123" dan tidak boleh kosong. Sebaliknya, data tidak valid jika NIP diisi dengan memasukkan abjad "abc". Pada rencana pengujian Nama data akan valid jika Nama diisi dengan memasukkan abjad "abc" dan tidak boleh kosong. Sebaliknya,

data tidak valid jika Nama diisi dengan memasukkan angka "123". Pada rencana pengujian Penjualan data akan valid jika diisi dengan memasukkan angka "123". Sebaliknya, data tidak valid jika Penjualan diisi dengan memasukkan abjad "abc". Pada rencana pengujian Komunikasi, Inovasi, Disiplin dan Loyalitas harus dipilih salah satu data pada *radiobutton* tersebut.

Tabel 2 Rancangan Test Case Form Data

Id	Deskripsi Pengujian	Hasil yang Diharapkan
A01	Mengisi NIP dengan "5", Nama diisi dengan "dandi", penjualan diisi dengan "80", memilih <i>radiobutton</i> Komunikasi, Inovasi, Disiplin dan Loyalitas pada <i>radiobutton</i> "Cukup(2)" selanjutnya yang harus dilakukan adalah klik Simpan.	Data akan tersimpan ke <i>database</i> dan data yang tersimpan akan ditampilkan pada tabel alternatif.
A02	Mengisi NIP dengan "abc", Nama diisi dengan "123", penjualan diisi dengan "abc", memilih <i>radiobutton</i> Komunikasi, Inovasi, Disiplin dan Loyalitas pada <i>radiobutton</i> "Cukup(2)" selanjutnya yang harus dilakukan adalah klik Simpan.	Menolak untuk tersimpan data tersebut ke dalam <i>database</i> .

### 3. Hasil dan Pembahasan

Berdasarkan rancangan uji kasus (*test case*) yang telah dibuat pada Tabel 1 dan 2, maka dapat

dilakukan pengujian. Hasil pengujian ditunjukkan pada Tabel 4.

Tabel 4 Hasil Pengujian

Id	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
L01	Memasukkan Username dengan "dandi123", Username diisi dengan "dandi123" kemudian klik tombol Login.	Akses Login berhasil, sistem akan menampilkan form Data.	Sistem akan muncul pesan "berhasil Login" dan setelah itu sistem muncul form Data.	Sesuai
L02	Mengisi Username dengan "abc123", Password diisi dengan "abc123" kemudian klik tombol Login.	Akses Login Gagal, sistem tetap menampilkan form Login.	Sistem akan muncul pesan "username atau password salah" dan tampilan tetap pada form Login.	Sesuai
A01	Mengisi NIP dengan "5", Nama diisi dengan "dandi", penjualan diisi dengan "80", memilih <i>radiobutton</i> Komunikasi, Inovasi, Disiplin dan Loyalitas pada <i>radiobutton</i> "Cukup(2)" selanjutnya yang harus dilakukan adalah klik Simpan.	Data akan tersimpan ke <i>database</i> dan data yang tersimpan akan ditampilkan pada tabel alternatif.	Data berhasil tersimpan, akan muncul pesan "Data Berhasil Tersimpan" dan data yang tersimpan ditampilkan di tabel alternatif	Sesuai
A02	Mengisi NIP dengan "abc", Nama diisi dengan "123", penjualan diisi dengan "abc", memilih <i>radiobutton</i> Komunikasi, Inovasi, Disiplin dan Loyalitas pada <i>radiobutton</i> "Cukup(2)" selanjutnya yang harus dilakukan adalah klik Simpan.	Menolak untuk tersimpan data tersebut ke dalam <i>database</i> .	Data gagal tersimpan dan menampilkan pesan kesalahan.	Sesuai

Total keseluruhan fungsi yang diuji ada 4 fungsi dengan fungsi login diuji sebanyak 2 kali, fungsi simpan diuji sebanyak 6 kali, fungsi edit diuji sebanyak 1 kali, dan fungsi hapus diuji sebanyak 1 kali. Jadi total pengujian sebanyak 10 kali. Jumlah fungsi yang ditemukan *error* sebanyak 1 fungsi dan 3 fungsi tidak ditemukan *error*.

#### 4. Kesimpulan

Pengujian aplikasi dengan *Blackbox Testing* bertujuan melihat program tersebut sama dengan tugas program tersebut tanpa mengetahui kode program yang dipakai. Berdasarkan pengujian kualitas aplikasi sistem seleksi sales terbaik, dapat ditarik kesimpulan bahwa pengujian dengan metode *Black Box* berbasis *Equivalence Partitions* dapat membantu proses pembuatan *case* pengujian, uji kualitas dan menemukan kesalahan yang tidak terdeteksi yang disebabkan oleh kesalahan pengetikan. Dalam pengujian aplikasi seleksi sales terbaik ditemukan kesalahan dalam mengedit data di mana tidak sesuai antara pesan yang muncul yaitu data berhasil disimpan sedangkan yang terjadi data tersebut tidak berubah. Pengujian ini menunjukkan bahwa terdapat celah *error* pada suatu form aplikasi sistem seleksi sales terbaik. Diharapkan kesalahan tersebut dapat diperbarui agar kualitas aplikasi yang dibuat akan lebih baik.

#### 5. Saran

Saran untuk pengembangan aplikasi selanjutnya adalah dapat menggunakan teknik pengujian *Blackbox Testing* lainnya seperti *Boundary Value* atau *Error Guessing* agar kita dapat menemukan kesalahan/error lainnya pada pengembangan aplikasi ini sehingga kualitas aplikasi yang dihasilkan jauh lebih baik sehingga dapat memberikan keuntungan bagi pengguna aplikasinya.

#### Referensi

- Akiladevi, R., Vidhupriya, P., & Sudha, V. (2018). Software Testing Tools. *International Journal of Pure and Applied Mathematics*, 118(18), 1783-1800.
- Ardana, I. S. (2019). Pengujian Software Menggunakan Metode Boundary Value Analysis dan Decision Table Testing. *Jurnal Teknologi Informasi*, XIV(03), 40-47.
- Aristoteles, Wardiyanto, & Pratama, A. A. (2015). Sistem Pakar Diagnosa Penyakit Pada Ikan Budidaya Air Tawar dengan Metode Forward Chaining. *Jurnal Komputasi*, 3(2), 99-168.
- 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.

- Latif, A. (2015). Implementasi Kriptografi Menggunakan Metode Advanced Encryption Standar (AES) Untuk Pengamanan Data Teks. *Jurnal Ilmiah Mustek Anim*, 4(2), 163-172.
- Mukhayaroh, A. (2016). Sistem Pendukung Keputusan Pemilihan Dstribution Sales Supervisor Dengan Metode FMADM-SAW Pada PT.Nirmala Pangan Sejahtera Bekasi. *Informatics For Educators And Professionals*, 1(1), 51-64.
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Apilkasi Prediksi Kelulusan SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, 1(3), 31-36.
- MZ, M. K. (2016). Pengujian Perangkat Lunak Metode Black Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah. *Jurnal Mikrotik*, 6(3), 30-48.
- Sasmito, G. W. (2017). Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal. *Jurnal Pengembangan IT (JPIT)*, 2(1), 7-12.
- Sethi, M. A. (2017). A Review Paper On Levels, Types & Techniques In Software Testing. *International Journal of Advanced Research In Computer Science*, 8(7), 269-271.
- Wahyudi, R., Utami, E., & Arief, M. R. (2016). Sistem Pakar E-Tourism Pada Dinas Pariwisata D.I.Y Menggunakan Metode Forward Chaining. *Jurnal Ilmiah DASI*, 17(2), 67-75.
- Wahyudi, S., Suheri, H., & Nurhadian, T. (2015). Impelementasi Sistem Keputusan Pengangkatan Karyawan Tetap PT. Imanuel Surya Utama Menggunakan Metode SAW. *Jurnal PROSISKO*, 2(1), 34-41.