

Teknik Pengujian *Equivalence Partions* untuk Pengujian Aplikasi Sistem Penunjang Keputusan Pegawai Terbaik Menggunakan *Black Box*

Ahmad Zuhair¹, Farhan Khadafi², Achmad Muzakih Andriansyah³, Bayu Saputra⁴, Aries Saifudin⁵

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

e-mail: ¹ahmdzuhairrr@gmail.com, ²farhankhadafi2@gmail.com, ³laieon000@gmail.com, ⁴bayusaputraubay@gmail.com, ⁵aries.saifudin@unpam.ac.id

Submitted Date: June 03rd, 2020
Revised Date: August 02nd, 2020

Reviewed Date: July 19th, 2020
Accepted Date: August 02nd, 2020

Abstract

The development of decision support system applications for the selection of the best employees is very important to support management in making decisions. This application must be free of errors because if there are errors can be detrimental to several parties, both the company and the employee. To guarantee that the floating application is error free, a test is needed. Testing on an application aims to check whether a program is running properly or there are still errors that must be corrected so that the program created will be a program that has good quality. The testing technique consists of White Box Testing and Black Box Testing which are widely used by testers to check the program is appropriate or not. The Best Employee Decision Support Application will be tested using Black Box Testing where the test only aims to see whether the program is in accordance with the function desired by the program without knowing the program code used. Types of testing of the various Black Box methods, one of which is the Equivalence Partitions used by the authors in this study. Equivalence Partitions Technique is a test based on data entry on each form in the Best Employee Decision Support System, each input menu will be tested and grouped according to function whether it is valid or invalid. With this test the quality of the software created can be expressed objectively very well and has fulfilled all the requirements set.

Keywords: Application; Testing; Black Box; Equality Partition; Decision Support System

Abstrak

Pengembangan aplikasi sistem penunjang keputusan pemilihan pegawai terbaik sangat penting untuk mendukung manajemen dalam mengambil keputusan. Aplikasi ini harus bebas dari kesalahan karena jika ada kesalahan dapat merugikan beberapa pihak, baik pihak perusahaan maupun pihak pegawai. Untuk menjamin bahwa aplikasi yang dikembangkan telah bebas dari kesalahan, maka perlu dilakukan pengujian. 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 *White Box Testing* dan *Black Box Testing* yang banyak digunakan para penguji untuk memeriksa program telah sesuai atau tidak. Aplikasi Penunjang Keputusan Pegawai Terbaik tersebut akan diuji menggunakan *Black Box 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 *Black Box* 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 Penunjang Keputusan Pegawai Terbaik, setiap menu masukan akan dilakukan pengujian dan dikelompokkan berdasarkan fungsinya baik itu bernilai valid ataupun tidak valid. Dengan adanya pengujian ini kualitas *software* yang dibuat dapat dinyatakan sangat baik secara obyektif dan telah memenuhi semua persyaratan yang ditetapkan.

Kata Kunci: Aplikasi; Pengujian; *Black Box*; *Equivalence Partitions*; Sistem Penunjang Keputusan

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. Pengujian perangkat lunak memiliki fungsi yang penting pada pengembangannya untuk menemukan galat (cacat) yang disebabkan oleh perbedaan antara hasil yang diharapkan dengan hasil sebenarnya (Pratama, Ristiano, Prayogo, Nasrullah, & Saifudin, 2020). 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.

Sistem penunjang keputusan untuk memilih pegawai terbaik sangat penting untuk mendukung manajemen dalam mengambil keputusan. Pemilihan pegawai terbaik menggunakan banyak kriteria, di antaranya adalah penjualan, komunikasi, inovasi, disiplin serta loyalitas. Sistem komputerisasi telah mendominasi di berbagai bidang kerja atau aktifitas untuk meningkatkan kemudahan, efektifitas dan efisiensi sehingga memiliki peranan yang sangat penting (Taufiq, Magfiroh, Yusup, & Yulianti, 2020), tetapi jika ada kesalahan dalam pemrograman dapat menimbulkan kerugian. Sehingga perlu dilakukan pengujian dengan tujuan menemukan kesalahan dan memperbaikinya sehingga sistem dikatakan layak untuk digunakan (Nurudin, Jayanti, Saputro, Saputra, & Yulianti, 2019).

Pada pengujian Sistem Penunjang Keputusan Pegawai Terbaik diusulkan menggunakan *Black Box Testing*, di mana pengujian bertujuan melihat program tersebut sama dengan tujuan program tersebut tanpa harus mengetahui kode programnya. Tahapan pertama pada pengujian dengan menggunakan *Black Box Testing* adalah mengidentifikasi masukan lalu diuji agar kita mengetahui letak kesalahannya. Pengujian menggunakan *Black Box Testing* merupakan sebuah pengujian yang digunakan untuk

melengkapi pengujian sebelumnya yaitu *White Box 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 penunjang keputusan pegawai 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 perangkat lunak memainkan peran penting dalam proses pengembangan karena mengadopsi keandalan dan kepuasan pelanggan serta memastikan kualitas produk/proyek yang dikembangkan (Akiladevi, Vidhupriya, & Sudha, 2018). Pengujian *software* sangat penting dilakukan karena setiap orang membuat kesalahan pada saat pembuatan *software*. 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). Kesalahan pada masing-masing *software* akan berbeda pada masing-masing *software*. 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 *Black Box* agar kualitas *software* lebih baik. Metode *Black Box 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 *Black Box Testing* adalah dengan menggunakan teknik *Equivalence Partitioning* (EP) yang penulis gunakan untuk menguji masukan serta membagi masukan ke dalam kelompok-kelompok berdasarkan fungsinya. *Equivalence Partitioning* merupakan sebuah pengujian berdasarkan masukkan data pada setiap form yang memecah domain masukan ke dalam kelompok valid dan tidak valid (Aziz, Setiawan, Khanh, Nurdiyansyah, & Yulianti, 2020). Sehingga didapatkan sebuah *Test Case* yang akurat (Aristoteles, Wardiyanto, & Pratama, 2015).

Dalam melakukan pengujian harus memilih

teknik yang tepat, yaitu teknik yang dapat menemukan kesalahan yang belum terdeteksi sehingga dapat meningkatkan kualitas software (Hendri, Manurung, Ferian, Hanaatmoko, & Yulianti, 2020). Di dalam penelitian ini pengujian akan dilakukan dalam 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 kasus uji (*Test Case*) yang telah dibuat pada Tabel 1 dan Tabel 2, maka

dapat dilakukan pengujian. Hasil pengujian ditunjukkan pada Tabel 3.

Tabel 3 Hasil Pengujian

ID	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
L01	Memasukkan Username dengan "dandi123", Username diisi dengan "dand123" 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* atau tidak berjalan semestinya sebanyak 1 fungsi dan 3 fungsi tidak ditemukan *error*.

4 Kesimpulan

Pengujian aplikasi dengan *Black Box Testing* bertujuan melihat program tersebut sama dengan tugas program tersebut tanpa mengetahui kode program yang dipakai. Berdasarkan pengujian kualitas aplikasi sistem Penunjang Keputusan Pegawai 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 Penunjang Keputusan Pegawai 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 Penunjang Keputusan Pegawai 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 *Black Box 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). A Study and Analysis on 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.
- 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
- Hendri, H., Manurung, J. W., Ferian, R. A., Hanaatmoko, W. F., & Yulianti, Y. (2020). Pengujian Black Box pada Aplikasi Sistem Informasi Pengelolaan Masjid Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3(2), 107-113. doi:10.32493/jtsi.v3i2.4694
- 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. *Jutis (Jurnal Teknik Informatika)*, 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.
- 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.
- 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
- 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.
- 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.
- 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
- 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.