

## Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox

Vadlan Febrian<sup>1</sup>, Muhamad Rizki Ramadhan<sup>2</sup>, Muhammad Faisal<sup>3</sup>, Aries Saifudin<sup>4</sup>

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia  
e-mail: <sup>1</sup>muhammadrizkiramadhan02@gmail.com, <sup>2</sup>phecan93@gmail.com, <sup>3</sup>vadlanf@gmail.com,  
<sup>4</sup>aries.saifudin@unpam.ac.id

Submitted Date: January 31<sup>st</sup>, 2020  
Revised Date: February 07<sup>th</sup>, 2020

Reviewed Date: February 03<sup>rd</sup>, 2020  
Accepted Date: March 30<sup>th</sup>, 2020

### Abstract

*In this employee payroll application, if there is an error program there will be a loss for employees and the company. Losses for employees, if this application program error occurs then the salary reduction will experience delays due to the difficulty in the process of calculating employee salaries and employees will be late in receiving salaries. Losses for the company, if there is an error program in this application, the company will suffer losses if the employee wants a salary reduction quickly but the company cannot calculate quickly and accurately. In solving this problem, the authors use the black box testing method. Black box testing method is a test that sees the results of execution through test data and ensures the function of the software. Black box testing method has several testing techniques, namely Sample Testing, Boundary Value Analysis, Equivalence Partitions and others. From the testing techniques that have been mentioned, we use the Equivalence Partitions testing technique. Equivalence Partitions are tests that refer to data entry on the employee payroll application form, input will be tested and then put together based on the test function, both valid and invalid values. The expected results of this test are a payroll system for employees who are computerized, have standard rules in the process of developing the program so that it is easy to develop and maintain, and can minimize errors in processing salary calculations for employees.*

*Keywords: Black Box Testing, Equivalence Partitions, Employee Payroll Application*

### Abstrak

Pada aplikasi penggajian pegawai ini, jika terdapat program *error* maka akan terjadi kerugian bagi pegawai dan perusahaan. Kerugian bagi pegawai, jika aplikasi ini terjadi program *error* maka penurunan gaji akan mengalami keterlambatan dikarenakan sulit dalam proses perhitungan gaji pegawai dan pegawai pun akan telat dalam menerima gaji. Kerugian bagi perusahaan, jika terjadi program *error* pada aplikasi ini, perusahaan akan mengalami kerugian jika pegawai menginginkan penurunan gaji dengan cepat tapi perusahaan tidak bisa menghitung dengan cepat dan tepat. Dalam menyelesaikan masalah ini, maka penulis menggunakan metode *black box testing*. Metode *black box testing* adalah pengujian yang melihat hasil eksekusi melalui data uji dan memastikan fungsi dari *software*. Metode *black box testing* mempunyai beberapa teknik pengujian, yaitu *Sample Testing*, *Boundary Value Analysis*, *Equivalence Partitions* dan lain-lain. Dari teknik pengujian yang sudah disebutkan, kami menggunakan teknik pengujian *Equivalence Partitions*. *Equivalence Partitions* merupakan pengujian yang mengacu kepada masukkan data pada form aplikasi penggajian pegawai, masukan akan dilakukan pengujian lalu disatukan berdasarkan fungsi pengujian, baik bernilai valid atau tidak valid. Hasil yang diharapkan dari pengujian ini yaitu sistem penggajian bagi para pegawai yang sudah terkomputerisasi, mempunyai standar aturan dalam proses pengembangan program sehingga mudah untuk dikembangkan dan dipelihara, serta dapat meminimalisir kesalahan dalam memproses perhitungan gaji bagi para pegawai.

Kata Kunci : *Black Box Testing, Equivalence Partitions, Aplikasi Penggajian Pegawai*

## 1. Pendahuluan

Pengujian perangkat lunak merupakan sebuah proses pengujian program yang dimaksudkan untuk mencari kesalahan pada *software*. Pengujian juga bertujuan untuk memastikan bahwa *software* memiliki kualitas yang baik. Kualitas *software* yang baik adalah *software* memenuhi kriteria yang diinginkan dan memberikan produktivitas yang tinggi (MZ, 2016). Pengujian perangkat lunak penting adanya untuk melakukan uji kualitas perangkat lunak agar menekan peluang terjadinya kesalahan pada manusia dan menutupi kekurangan manusia yang kurang mampu melakukan komunikasi dengan sempurna, hal ini mengakibatkan pengembangan perangkat lunak terhambat dan menjadi bergantung pada jaminan kualitas *software* (Liana, 2015).

Pengujian perangkat lunak memiliki peranan penting dalam suatu sistem informasi, dengan pengujian ini dapat diketahui galat atau *error* yang akan muncul pada perangkat lunak. Dengan pengujian perangkat lunak diharapkan dapat meminimalisir kesalahan dan cacat pada sebuah *software* dan sebagai pengukuran kualitas dari *software* tersebut (Kurniawan, 2017).

*Software* yang akan diuji adalah sebuah aplikasi penggajian pegawai. Aplikasi yang dapat digunakan untuk membantu perusahaan dalam menghitung gaji setiap pegawai dan menyimpan catatannya. Jika terjadi masalah pada *software*/program atau program *error* maka akan terjadi kerugian pada pegawai dan perusahaan.

Bagi pegawai, jika aplikasi ini terjadi program *error* maka penurunan gaji akan mengalami keterlambatan dikarenakan sulit dalam proses perhitungan gaji pegawai dan pegawai pun akan telat dalam menerima gaji. Kerugian bagi perusahaan, jika terjadi program *error* pada aplikasi ini, perusahaan akan mengalami kerugian jika pegawai menginginkan penurunan gaji dengan cepat tapi perusahaan tidak bisa menghitung dengan cepat dan tepat.

Metode yang kami gunakan untuk melakukan pengujian pada aplikasi penggajian pegawai adalah metode pengujian *black box*. Pengujian *black box* pengujian perangkat lunak yang tanpa perlu memperlihatkan hasil detail perangkat lunak. Pengujian *black box* cukup melihat nilai keluaran berdasarkan nilai masukan itu sendiri (Hanifah, Alit, & Sugiarto, 2016).

Rencana penelitian/pengujian pada aplikasi penggajian pegawai akan menggunakan metode pengujian *black box* dengan teknik *Equivalence Partitions* (Muttaqin & Hidayat, 2018) *Equivalence Partitions* merupakan pengujian yang mengacu kepada masukkan data pada form aplikasi penggajian pegawai, masukkan akan dilakukan pengujian lalu disatukan berdasarkan fungsi pengujian, baik bernilai valid atau tidak valid (Hidayat & Muttaqin, 2018).

## 2. Metodologi

Tahapan pengujian sangat diperlukan dalam sistem informasi untuk memastikan aplikasi telah memenuhi persyaratan teknis dan bisnis yang diharapkan sebelum diserahkan kepada pengguna (*customer*). Pengujian perangkat lunak adalah 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 *software* mempunyai dua metode yaitu *black-box* 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).

Pada pengujian ini kami menggunakan masukan data acak yang akan memastikan agar sistem dapat menolak menyimpan database, sehingga sistem dapat dikatakan layak digunakan (Nurudin, Jayanti, Saputro, Saputra, & Yulianti, 2019). Pengujian ini melakukan beberapa tahapan, tahapan yang pertama dimulai dengan *test case* dari perangkat lunak yang akan kami uji, menggunakan metode *Equivalence Partitions* lalu melakukan inialisasi standar berupa *grade partition* dari masukan dan keluaran. Demi mendapatkan dokumentasi dari efektifitas dan pengujian tersebut (Kesuma Jaya, Gumilang, Tresnawati, Andersen, & Desyani, 2020). Dari hasil pengujian dapat dilihat table *test case* digunakan sebagai acuan berhasil tidaknya pengujian tersebut (Arochman, Darmwan, & Santoso, 2016). Berikut adalah penjelasan dari tabel *Test Case*.

**Penggajian Sederhana**

**Aplikasi Penggajian Pegawai**

NIK

NAMA

STATUS

JABATAN

**Gaji Kotor**

GAJI POKOK

UANG TRANSPORT

UANG MAKAN

LEMBUR

SUB TOTAL

**Potongan**

BPJS

**Gaji Bersih**

GRAND TOTAL GAJI BERSIH

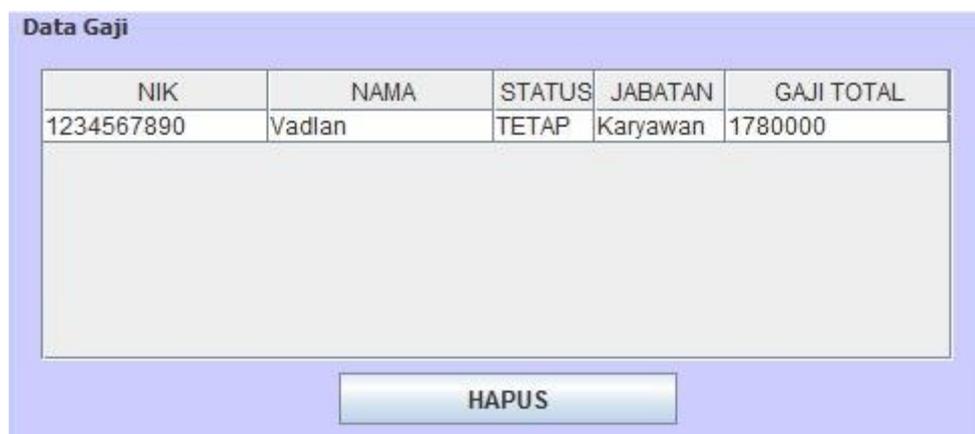
Gambar 1 Form Penggajian

Pada form dari Gambar 1, terdapat rencana pengujian NIK data akan valid jika NIK diisi dengan memasukkan angka (contohnya "123") yang terdiri dari 10 digit dan tidak boleh kosong. Sebaliknya, data tidak valid jika NIK diisi dengan memasukan huruf (contohnya "abc") atau lebih dari 10 digit, kolom NAMA diisi sesuai dengan nama pegawai perusahaan dan harus memasukan huruf serta tidak boleh kosong. Sebaliknya, data tidak valid jika NAMA diisi dengan memasukkan angka (contohnya "123") atau kosong, kolom STATUS diisi dengan memilih sesuai dengan status pegawai dan tidak boleh kosong, kolom JABATAN diisi sesuai dengan memilih jabatan pegawai dan tidak boleh kosong, jika kosong maka kolom GAJI POKOK tidak dapat menampilkan gaji pokok pegawai dan data tidak valid, kolom GAJI POKOK akan otomatis tampil sesuai

dengan jabatan pegawai perusahaan, kolom UANG TRANSPORT diisi dengan memasukkan angka (contohnya "123") atau boleh kosong, data tidak valid jika UANG TRANSPORT diisi dengan huruf (contohnya "abc"), kolom UANG MAKAN diisi dengan memasukkan angka (contohnya "123") dan boleh kosong, data tidak valid jika UANG MAKAN diisi dengan huruf (contohnya "abc"), kolom LEMBUR diisi sesuai data lembur pegawai dan otomatis nominal lembur pegawai akan tampil dan boleh kosong, kolom SUB TOTAL merupakan kolom yang menghitung gaji kotor pegawai, kolom BPJS diisi dengan memilih sesuai golongan yang diinginkan pegawai dan boleh kosong, kolom GRAND TOTAL GAJI BERSIH merupakan kolom yang menampilkan gaji bersih dari pegawai tersebut.

Tabel 1. Rancangan *Test Case* Form Penggajian

ID	Deskripsi	Hasil yang diharapkan
A01	Mengisi NIK dengan “1234567890”, mengisi NAMA “Vadlan”, memilih STATUS “TETAP”, memilih JABATAN “Karyawan”, mengisi UANG TRANSPORT “30000”, mengisi UANG MAKAN “30000”, memilih LEMBUR “2 jam”, memilih Golongan BPJS “Gol 1”, kemudian klik tombol “HITUNG” dan “TAMBAH”	Sistem bisa menampilkan gaji pokok pegawai dan menghitung gaji kotor lalu menampilkannya pada GRAND TOTAL GAJI BERSIH serta menyimpan data ke dalam database
A02	Mengisi NIK “AAA” atau tidak diisi, mengisi NAMA “123” atau tidak diisi, tidak memilih JABATAN, kemudian klik tombol “HITUNG” dan “TAMBAH”	Sistem menolak untuk menampilkan gaji pokok pegawai, dan menolak untuk menghitung gaji kotor, dan GRAND TOTAL GAJI BERSIH kosong serta menampilkan pesan peringatan, mohon data diisi dengan benar



Gambar 2 Data Gaji

Dari form pada Gambar 2, akan dilakukan rencana pengujian. Kolom NIK, NAMA, STATUS, JABATAN dan GAJI TOTAL adalah tempat penyimpanan data lengkap gaji pegawai yang sudah dimasukkan melalui form penggajian dan data tidak valid jika sudah

berhasil disimpan tetapi data kosong, tombol HAPUS merupakan tombol yang akan menghapus data salah satu pegawai yang sudah disimpan dan tidak valid jika mendapat pesan “data tidak dapat dihapus”.

Tabel 2. Rancangan *Test Case* Form Data Gaji

ID	Deskripsi	Hasil yang diharapkan
B01	Melihat Data Gaji pegawai yang sudah disimpan melalui Form Penggajian, memilih data gaji pegawai untuk dihapus dari Data Gaji	Sistem bisa menampilkan data gaji pegawai yang sudah disimpan pada Data Gaji dan sistem bisa menghapus data gaji pegawai yang terdapat pada Data Gaji

B02	Tidak menampilkan “Data Gaji” pegawai yang sudah disimpan, tidak memilih “Data Gaji” pegawai yang ingin dihapus dari “Data Gaji”	Sistem tidak menampilkan data gaji pegawai yang sudah disimpan dan sistem tidak dapat menghapus data gaji pegawai yang disimpan serta menampilkan pesan peringatan, data tidak dapat dihapus
-----	--	--

### 3. Hasil dan Pembahasan

Rancangan pengujian sudah dibuat, lalu dilakukan pengujian. Hasil dari pengujian kemudian dicatat didalam kolom pengujian. Setelah itu dibuat sebuah kesimpulan yang berdasarkan penilaian keselarasan antara hasil pengujian dengan hasil yang diharapkan. Jika hasilnya berhasil maka pengujian sesuai dengan yang diharapkan. Jika tidak berhasil maka hasil pengujian tidak sesuai seperti yang diharapkan. Selanjutnya hasil pengujian akan ditunjukkan pada tabel 3.

Tabel 3. Hasil Pengujian

ID	Deskripsi	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
A01	Mengisi NIK dengan “1234567890”, mengisi NAMA “Vadlan”, memilih STATUS “TETAP”, memilih JABATAN “Karyawan”, mengisi UANG TRANSPORT “30000”, mengisi UANG MAKAN “30000”, memilih LEMBUR “2 jam”, memilih Golongan BPJS “Gol 1”, kemudian klik tombol “HITUNG” dan “TAMBAH”	Sistem bisa menampilkan gaji pokok pegawai dan menghitung gaji kotor lalu menampilkannya pada GRAND TOTAL GAJI BERSIH serta menyimpan data ke dalam database	Menampilkan gaji pokok, dan menampilkan hasil hitung dari gaji kotor, menampilkan GRAND TOTAL GAJI BERSIH dan tersimpan di database.	Berhasil
A02	Mengisi NIK “AAA” atau tidak diisi, mengisi NAMA “123” atau tidak diisi, tidak memilih JABATAN, kemudian klik tombol “HITUNG” dan “TAMBAH”	Sistem menolak untuk menampilkan gaji pokok pegawai, dan menolak untuk menghitung gaji kotor, dan GRAND TOTAL GAJI BERSIH kosong serta menampilkan pesan peringatan, mohon data diisi dengan benar	Tidak menampilkan gaji pokok, tidak menghitung gaji kotor, dan GRAND TOTAL GAJI BERSIH kosong dan menampilkan pesan peringatan, mohon data diisi	Berhasil
B01	Melihat Data Gaji pegawai atas nama “Vadlan” yang sudah disimpan melalui Form Penggajian, memilih data gaji pegawai atas nama “Vadlan” untuk dihapus dari Data Gaji	Sistem bisa menampilkan data gaji pegawai yang sudah disimpan pada Data Gaji dan sistem bisa menghapus data gaji pegawai yang terdapat pada Data Gaji	Menampilkan data gaji pegawai berupa data NIK, NAMA, STATUS, JABATAN dan GAJI TOTAL serta dapat menghapus data gaji pegawai perusahaan	Berhasil

B02	Tidak menampilkan “Data Gaji” pegawai yang sudah disimpan, tidak memilih “Data Gaji” pegawai yang ingin dihapus dari “Data Gaji”	Sistem tidak menampilkan data gaji pegawai yang sudah disimpan dan sistem tidak dapat menghapus data gaji pegawai yang disimpan serta menampilkan pesan peringatan, data tidak dapat dihapus	Tidak menampilkan data gaji pegawai berupa data NIK, NAMA, STATUS, JABATAN dan GAJI TOTAL serta tidak dapat menghapus data gaji pegawai perusahaan	Berhasil
-----	--	--	--	----------

Pengujian pada aplikasi penggajian pegawai, dilakukan pengujian pada 2 form, yaitu form penggajian diuji sebanyak 2 kali, form data gaji diuji sebanyak 2 kali. Total keseluruhan pengujian ada 4 kali pengujian.

#### 4. Kesimpulan

Dari hasil pengujian aplikasi penggajian pegawai, dapat ditarik kesimpulan bahwa pengujian dengan metode *black box testing* sangat dibutuhkan, agar setiap perangkat lunak diuji terlebih dahulu untuk menguji tingkat kesalahan sistem. Dan sistem yang berjalan dengan optimal akan memudahkan kinerja *user/pihak* pengguna, tanpa adanya masalah yang dialami oleh sistem, maka dari itu pengujian sangat dibutuhkan untuk menentukan sistem yang nanti akan dipakai layak atau tidak.

#### 5. Saran

Jika ingin membuktikan kelayakan sistem sebaiknya dilakukan minimal dua kali pengujian, agar hasil dari dua pengujian tersebut dapat disamakan nilainya walaupun menggunakan dua metode yang berbeda. Hasil tersebut dapat memberikan gambaran yang pembuktiannya dapat diterima secara *logic* dan rasional.

#### Referensi

Akmaludin. (2013). Analisis Rancangan Animasi Interaktif Pembelajaran Anatomi Otak Manusia. *Techo Nusa Mandiri*, 1-11.  
Arochman, Darmwan, A. S., & Santoso, F. H. (2016). Otomatisasi Desain Test Case Pengujian Perangkat Lunak Metode Black-Box Testing Dengan Teknik Equivalence Partitioning Menggunakan Algoritma Genetika. *ictech arochmannk*, 47-52.

Hanifah, U., Alit, R., & Sugiarto. (2016). Penggunaan Metode Black Box Pada Pengujian Sistem Informasi Surat Keluar Masuk. *Jurnal Teknik Informatika*, 34-38.  
Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online Menggunakan Black Blox testing dengan Menggunakan Metode Equivalence Partitioning dan Boundary Value Analysis. *Jurnal Teknik Informatika UNIS*, 6(1), 25-29.  
Kesuma Jaya, M. A., Gumilang, P., Tresnawati, Andersen, Y. P., & Desyani, T. (2020). Pengujian Black Box pada Aplikasi Sistem Penunjang Keputusan Seleksi Calon Pegawai Negeri Sipil Menggunakan Teknik Equivalence Partitions. *Jurnal Informatika Universitas Pamulang*, 131-136.  
Kurniawan, T. A. (2017). Pengujian Perangkat Lunak. *Pengujian Struktur Program Dengan Pengujian Jalur Dasar (Basic Path Testing) : Teori dan Aplikasi*, 29-32.  
Liana, L. (2015). Pengujian Perangkat Lunak. *Pengujian Perangkat Lunak*, 1-15.  
Muttaqin, M., & Hidayat, T. (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*, 23-27.  
MZ, M. K. (2016). Pengujian Perangkat Lunak Metode Black-Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah. *Mikrotik*, 2-3.  
Nurudin, M., Jayanti, W., Saputro, R. D., Saputra, M. P., & Yulianti. (2019). Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika*, 27-33.