

PENGUJIAN APLIKASI E-FARMER DALAM PERHITUNGAN KEUNTUNGAN DENGAN METODE *BLACKBOX TESTING* *BOUNDARY VALUE ANALYSIS*

¹⁾Desi Novianti, ²⁾Dewi Anjani

^{1) 2)}Program Studi Teknik Informatika, Fakultas Teknik dan Ilmu Komputer

^{1) 2)}Universitas Indraprasta PGRI

Email: ¹⁾desi.novi4nti@gmail.com, ²⁾dewiunindra@gmail.com

ABSTRACT

A software Testing is one of the stages in building an application where the test device that is implemented imperfectly certainly has a bad effect on the resulting software. E-farmer is an android based application that can provide a farmer experience to calculate the yield of profits for each harvest. The purpose of testing E-farmer is to see the error rate that occurs in the software. So that by testing this application, it is hoped that the application made is in accordance with the function and purpose of the application. Where this test uses the Black-box Testing method with the Boundary Value Analysis approach. The results obtained from this study are testing shows that the application has a success rate of 88, 89%. Three fields need to be improved, namely the id_petani, year, and semester fields.

Keywords: *Software Testing, E-farmer, Boundary Value Analysis, Black-box Testing*

I. PENDAHULUAN

E-farmer adalah aplikasi berbasis android yang digunakan sebagai media penghubung antara petani yang menghasilkan beras dan masyarakat umum yang membutuhkan. Aplikasi ini juga dapat memberikan pengalaman petani untuk melakukan perhitungan hasil keuntungan pada setiap panen (Mufti, Novianti, & Anjani, 2017).

Dalam kaitannya dengan pembuatan aplikasi E-farmer, setelah aplikasi diimplementasi maka aplikasi perlu dilakukan Pengujian perangkat lunak atau *Software Testing* yaitu dimana melakukan aktivitas-aktivitas yang bertujuan untuk mengevaluasi atribut-atribut atau kemampuan sebuah program atau sistem dan menetapkan aplikasi sesuai dengan hasil yang diharapkan. Menurut Krichen & Tripakis, (2004) Pengujian bertujuan untuk melihat tingkat kesalahan yang terjadi pada perangkat lunak. Sehingga dengan melakukan pengujian aplikasi ini diharapkan aplikasi yang dibuat sudah sesuai dengan fungsi dan tujuan aplikasi.

Pengujian aplikasi pada penelitian ini menggunakan metode *Blackbox Testing* dengan pendekatan *Boundary Value Analysis* dimana pengujian merupakan pengujian fungsi dari setiap modul pada perhitungan keuntungan pada aplikasi. Dengan dilakukan pengujian ini maka aplikasi E-farmer terutama untuk perhitungan keuntungan sesuai dengan apa yang diharapkan.

II. METODE PENELITIAN

Pengujian merupakan salah satu tahapan dalam membangun sebuah aplikasi dimana perangkat pengujian yang diimplementasikan secara tidak sempurna tentunya memberikan pengaruh yang buruk terhadap perangkat lunak yang dihasilkan. Pengujian perangkat lunak yang tidak efektif dan tidak lengkap dapat menyebabkan berbagai masalah saat perangkat lunak digunakan oleh pengguna akhir (Hanifah, Alit, & Sugiarto, 2016). Umumnya metode pengujian perangkat lunak hanya mencakup suatu area dalam perangkat lunak (*Data Driven Testing*) hanya menguji kemampuan perangkat lunak untuk menerima dan memproses data masukan (Hanifah et al., 2016).

Pengujian perangkat lunak dapat dibedakan menjadi dua yaitu *Black Box Testing* dan *White Box Testing*. Metode pengujian *white box* menekankan pengujian berdasarkan kode sumber program yang

dibuat. Sementara *Black Box Testing* adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data pengujian dan pengecekan fungsionalitas perangkat lunak. Pengujian ini berfokus pada fungsi *system* (Syaban & Bunyamin, 2015). Dalam penelitian ini dibahas tentang *Black Box*, dengan melakukan pengujian fungsi aplikasi maka langkah awal dalam mendeteksi kesalahan sebuah aplikasi.

Keuntungan menggunakan metode *Blackbox Tetsting* adalah Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu, pengujian dilakukan dari sudut pandang pengguna, ini membantu mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan, dan Pemrogram dan penguji saling bergantung satu sama lain (Ammann, Offutt, & Versio, 2016).

Kekurangan dari metode Pengujian *Blackbox* adalah Kasus pengujian sulit untuk dirancang tanpa spesifikasi yang jelas, kemungkinan pengulangan tes yang telah dilakukan pemrogram, beberapa bagian belakang tidak diuji sama sekali (Ammann et al., 2016).

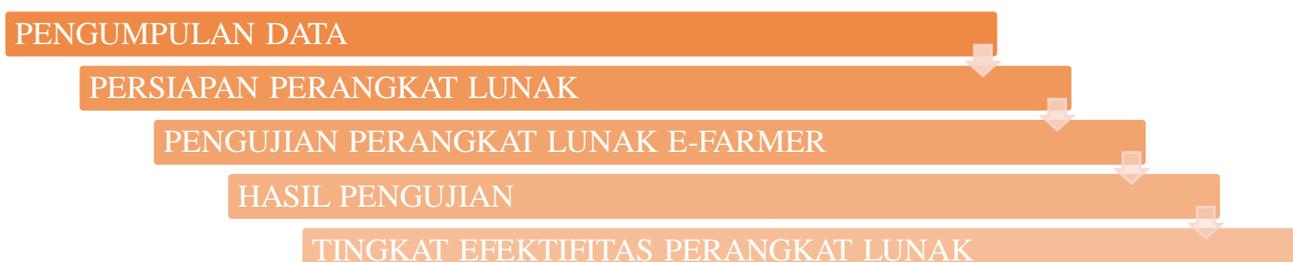
Pendekatan yang digunakan dalam pengujian ini adalah *Boundary Value Analysis (BVA)* yaitu metode yang memungkinkan untuk memilih kasus uji yang menguji batas nilai *input* (Julian 2015). Analisis nilai batas merupakan suatu teknik dalam metode pengujian *black box* yang menitikberatkan pada proses *input* dengan menguji nilai batas atas dan batas bawah (Dijkstra, Kipping, & Mézière., 2015). Prinsip kerja BVA adalah *error* yang sering terjadi adalah pada saat proses *input* dan BVA bekerja pada proses *input*.

Algoritma pengujian black box dengan teknik nilai batas Analisisnya adalah sebagai berikut (Xu, Chen, Wang, & Rud., 2016):

- 1) Jika kondisi masukan berada dalam kisaran nilai x dan y , maka kasus uji harus dibuat dengan data sampel $x-1$, x , y , $y + 1$.
- 2) Jika kondisi *input* menggunakan sejumlah nilai, maka *test case* dibuat untuk data sampel minimum -1 , minimum, maksimum, maksimum $+1$.
- 3) Lakukan langkah 1 dan 2 untuk proses keluaran.
- 4) Jika data sudah memiliki batasan *input* (misalnya *input* karakter disetel ke *maks.* 10), maka *test case* dibuat batas ini.

Pada penelitian ini dilakukan beberapa kegiatan untuk mendapatkan hasil yang diinginkan, tahapan tersebut dapat dilihat pada gambar 1, yaitu:

- 1) Pengumpulan Data. Tahap ini digunakan untuk mengumpulkan terori dan pengetahuan tentang pengujian *backbox* dengan metode BVA.
- 2) Persiapan Perangkat Lunak. Tahap ini, mempersiapkan perangkat lunak yang akan di uji coba sebagai objek penelitian
- 3) Pengujian Perangkat Lunak *E-Farmer*. Tahap pengujian ini, pertama melakukan pemilihan data untuk uji coba, kemudian mendapatkan data uji yang akan diterapkan pada pengujian kemudian dilakukan pengujian berdasarkan algoritma BVA.
- 4) Hasil Pengujian. Hasil pengujian di dapat dari perhitungan uji coba ini, adalah perhitungan terhadap tingkat *error* pada setiap modul yang di uji.
- 5) Tingkat Efektivitas Perangkat Lunak. Menghitung tingkat efektivitas perangkat lunak berdasarkan nilai *error* yang didapat.



Gambar 1. Metode Penelitian

III. HASIL DAN PEMBAHASAN

Pada aplikasi *e-farmer* terdapat *form* analisa dilihat pada gambar 2 yang mana difungsikan sebagai alat bantu produsen atau petani untuk melakukan penaksiran harga dari modal yang telah dikeluarkan dan pendapatan hasil dari pertanian. Di halaman ini, petani diberikan harga modal dari hasil pertaniannya dan estimasi akan harga jual yang dapat dijadikan acuan.

Pada *form* Analisa terdiri dari 17 *field* yang dapat dilihat pada gambar 3. Dimana setiap *field* memiliki karakter masing-masing. Seperti *id_petani* yang bertipe numerik dimana data yang dimasukkan adalah berupa angka yang berjumlah 10 karakter.

KETERANGAN	JUMLAH
MODAL:	
Benih	0 Kg
Pupuk Urea	0 Kg
Pupuk SP36	0 Kg
Pupuk NPK Ponska	0 Kg
Pestisida	0 Liter
Pengolahan Lahan	0 HOKp
Pencabutan Bibit dan Penanaman	0 HOKw
Penyiangan dan Pemupukan ke-1	0 HOKp
Penyiangan dan Pemupukan ke-2	0 HOKp
Penyemprotan	0 HOKp
Panen dan Pasca Panen	0 HOKp
Biaya Pengeringan	0 HOKp
PENDAPATAN:	
Hasil Gabah Basah	0 Kg

Gambar 2. Tampilan *Form* Analisa

Field Name	Data Type	Description (Optional)
ID petani	Number	terdiri dari 10 karakter angka
Tahun	Date/Time	vvvv
Semester	Short Text	hanva 10 karakter string
Luas sawah	Number	inputan bertipe angka
benih	Number	inputan bertipe angka
pupuk urea	Number	inputan bertipe angka
pupuk SP26	Number	inputan bertipe angka
Pupuk NPK Ponska	Number	inputan bertipe angka
Pestisida	Number	inputan bertipe angka
Pengolahan Lahan	Number	inputan bertipe angka
Pencabutan Bibit dan Penanaman	Number	inputan bertipe angka
Penyiangan dan Pemupukan ke-1	Number	inputan bertipe angka
Penyiangan dan pemupukan ke-2	Number	inputan bertipe angka
Penyemprotan	Number	inputan bertipe angka
Panen dan Pasca Panen	Number	inputan bertipe angka
Biaya Pengeringan	Number	inputan bertipe angka
Hasil Gabah Basah	Number	inputan bertipe angka

Gambar 3. Struktur Tabel *Form* Analisa

Berdasarkan gambar 3 maka dilakukan Uji terhadap 17 *field* yang terdapat pada *form*. Aturan data sampel yang digunakan adalah data normal, data minimal, data maksimal dan data maksimal +1. Hasil Pengujian *field id_petani* dapat dilihat pada Tabel 1.

Tabel 1. Hasil Uji *Field id_petani*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
123456	Salah	Benar	Sukses
A123456790	Salah	Benar	Sukses
0000000000	Salah	Salah	Gagal
1234567890	Benar	Benar	Sukses

Tabel 1 menghasilkan data sampel sukses yang bisa ditangani oleh *field id_petani* dimana tingkat kesuksesan adalah 75 %. Begitu juga untuk tabel 2 dimana Tingkat kesuksesan pada *field tahun* adalah 75 %.

Tabel 2. Hasil Uji *Field Tahun*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
23/02/2019	Benar	Benar	Sukses
02/23/2019	Salah	Benar	Sukses
2019/23/02	Salah	Benar	Sukses
2019	Salah	Salah	Gagal

Tabel 3. Hasil Uji *Field Semester*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
Satu	Benar	Benar	Sukses
1 (Satu)	Salah	Salah	Gagal
Satu-satu	Salah	Salah	Gagal

Sementara tabel 3, didapat tingkat kesuksesan pada *field semester* adalah 50% dimana *input* yang diharapkan adalah karakter yang hanya menunjukkan informasi semester dengan huruf namun jika dimasukkan keterangan huruf yang mengandung karakter selain huruf masih bisa diterima aplikasi.

Tabel 4. Hasil Uji *Field Luas Sawah*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
1.24	Benar	Benar	Sukses
1,25	Salah	Benar	Sukses
123	Benar	Benar	Sukses

Tabel 5. Hasil Uji *Field Benih*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
1.24	Benar	Benar	Sukses
1,25	Salah	Benar	Sukses
123	Benar	Benar	Sukses

Tabel 6. Hasil Uji *Field Pupuk Area, Pupuk SP26, Pupuk Ponska*

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
Satu Kilo Gram	Salah	Benar	Sukses
1 Kg	Salah	Benar	Sukses
1.5	Benar	Benar	Sukses

Tingkat keberhasilan pengujian untuk *field Luas sawah pada tabel 4 dan field benih* adalah 100%, dimana semua data sampel yang dimasukkan sukses dieksekusi atau sesuai dengan apa yang diharapkan. Data sampel untuk *Field Pupuk Area, Pupuk SP26, Pupuk Ponska* adalah sama, dan memiliki tingkat kesuksesan yang sama yaitu 100% dilihat pada tabel 6. Dimana data yang diinginkan adalah data berupa angka.

Tabel 7. Hasil Uji *Field* Pestisida

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
Satu Liter	Salah	Benar	Sukses
1 L	Salah	Benar	Sukses
1.5	Benar	Benar	Sukses

Tabel 8. Hasil Uji *Field* Pengolahan Lahan, Penyemprotan, Panen dan Pasca Panen, Biaya Pengeringan

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
Satu HOkp	Salah	Benar	Sukses
1,5	Salah	Benar	Sukses
1.5	Benar	Benar	Sukses

Tabel 9. Hasil Uji *Field* Hasil Gabah Bersih

Data Sampel	Perkiraan Hasil	Hasil	Kesimpulan
1	Benar	Benar	Sukses
Satu KG	Salah	Benar	Sukses
1 Kg	Salah	Benar	Sukses
1.5	Benar	Benar	Sukses

Tabel 10. Rekapitulasi Pengujian BVA

No	Nama <i>Field</i>	Tingkat Kesuksesan
1.	Id_petani	75%
2.	Tahun	75%
3.	Semester	50%
4.	Luas_sawah	100%
5.	Benih	100%
6.	Pupuk area	100%
7.	pupuk SP26	100%
8.	Pupuk NPK Ponska	100%
9.	Pestisida	100%
10.	Pengolahan Lahan	100%
11.	Pencabutan Bibit dan Penanaman	100%
12.	Penyiangan dan Pemupukan ke-1	100%
13.	Penyiangan dan pemupukan ke-2	100%
14.	Penyemprotan	100%
15.	Panen dan Pasca Panen	100%
16.	Biaya Pengeringan	100%
17.	Hasil Gabah Basah	100%
	Rata- rata	88.89%

Persentase tingkat keberhasilan untuk *field* Pestisida pada tabel 7, Pengolahan Lahan, Penyemprotan, Panen dan Pasca Panen, Biaya Pengeringan yang ditampilkan pada tabel 8, dan Hasil Gabah Bersih pada tabel 9 adalah 100%. Sehingga dapat dikatakan *field- field* tersebut sudah sesuai dengan yang dirancang dan dibutuhkan.

IV. PENUTUP

Dari penelitian ini dapat dilihat tingkat kesalahan yang terjadi pada perangkat lunak *E-Farmer*. Dimana hasil pengujian menunjukkan bahwa aplikasi mampu menjawab, baik data normal maupun data tidak normal dengan proporsi 88, 89% yang dapat dilihat pada tabel 10. Dimana tiga bidang perlu ditingkatkan yaitu *field id_petani*, tahun, dan semester, sehingga dapat meningkatkan kemampuan aplikasi dalam mengolah data dalam kondisi normal maupun tidak normal.

DAFTAR PUSTAKA

- Ammann, P., Offutt, J., & Versio, I. (2016). *Introduction to Software Testing Edition 2*.
- Dijkstra, K. D. B., Kipping, J., & Mézière., N. (2015). Sixty new dragonfly and damselfly species from Africa (Odonata). *Odonatologica*, 4, 447–678.
- Hanifah, U., Alit, R., & Sugiarto, S. (2016). Penggunaan Metode Black Box Pada Pengujian Sistem Informasi Surat Keluar Masuk. *SCAN - Jurnal Teknologi Informasi Dan Komunikasi*, 11(2), 33–40. Retrieved from <http://ejournal.upnjatim.ac.id/index.php/scan/article/view/643>
- Krichen, M., & Tripakis, S. (2004). Black-Box Conformance Testing for Real-Time Systems. *Conference: Model Checking Software, 11th International SPIN Workshop, Barcelona, Spain, LNCS(2989)*, 109–126. <https://doi.org/10.1007/s10703-009-0065-1>
- Mufti, A., Novianti, D., & Anjani, D. (2017). Designing mobile farmer application using object oriented analysis and design. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, 3(2), 105. <https://doi.org/10.26594/register.v3i2.1100>
- Syaban, R. M., & Bunyamin. (2015). Pengembangan Sistem Informasi Pengelolaan Surat Masuk Dan Surat Keluar Berbasis Web Di Dinas Sosial Tenaga Kerja Dan Transmigrasi Kabupaten Garut Menggunakan Framework Php. *Algoritma*, 12(2), 301–311.
- Xu, S., Chen, L., Wang, C., & Rud., O. (2016). A comparative study on black-box testing with open source applications. *IEEE/ACIS 17th Int. Conf. Softw. Eng. Artif. Intell. Netw. Parallel/Distributed Comput. SNPD*, 527–532.