

PENERAPAN METODE *UNIT TESTING* PADA PENGUJIAN APLIKASI *PETSHOP* BERBASIS WEB

Pundi Razzaq Widodo¹, Fikri Fadlillah²

Informatika, Fakultas Teknologi Industri Universitas Gunadarma, Gunadarma

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

pundirq@gmail.com, fikri_Fadlillah@staff.gunadarma.ac.id

Abstract (English)

The development of information technology has changed the way we interact, work and live everyday. Web-based Petshop application is one of the e-commerce innovations that allow buying and selling pet products online. In web application development, testing is an important step to ensure good functionality and in accordance with user expectations. Unit testing, a software testing method, is used to test small units of code. This writing is a continuation of the design of a web-based Petshop application and focuses on system implementation with unit testing. This research uses the PHP framework and only focuses on testing the model class of the MVC architecture. The purpose of this research is to test web-based applications with the unit testing method. The research was conducted with literature study, requirements analysis, software development, unit testing using PHP framework, and overall evaluation. The writing systematic consists of Introduction, Theoretical Basis, Implementation and Testing, and Closing.

Abstrak (Indonesia)

Perkembangan teknologi informasi telah mengubah cara kita berinteraksi, bekerja, dan hidup sehari-hari. Aplikasi Petshop berbasis web adalah salah satu inovasi e-commerce yang memungkinkan pembelian dan penjualan produk hewan peliharaan secara online. Dalam pengembangan aplikasi web, pengujian menjadi langkah penting untuk memastikan fungsi yang baik dan sesuai dengan harapan pengguna. Unit testing, metode pengujian perangkat lunak, digunakan untuk menguji unit-unit kecil dalam kode. Penulisan ini adalah kelanjutan dari perancangan aplikasi Petshop berbasis web dan fokus pada implementasi sistem dengan pengujian unit testing. Penelitian ini menggunakan framework PHP dan hanya memfokuskan pengujian pada kelas model dari arsitektur MVC. Tujuan penelitian ini adalah untuk menguji aplikasi berbasis web dengan metode unit testing. Penelitian dilakukan dengan studi literatur, analisis kebutuhan, pengembangan perangkat lunak, pengujian unit testing menggunakan framework PHP, dan evaluasi secara keseluruhan. Sistematika penulisan terdiri dari Pendahuluan, Landasan Teori, Implementasi dan Pengujian, serta Penutup.

Article History

Submitted: 20 January 2023

Accepted: 29 January 2023

Published: 30 January 2023

Key Words

Web, PHP, Unit Test

Sejarah Artikel

Submitted: 20 January 2023

Accepted: 29 January 2023

Published: 30 January 2023

Kata Kunci

Web, PHP, Unit Test

PENDAHULUAN

A. Latar Belakang

Dalam aktivitas pengujian sistem, pengujian menjadi hal yang penting untuk memastikan kinerja dan kesesuaian sistem dengan harapan yang diinginkan. Dengan adanya pengujian, proses pengembangan sistem dapat dipercepat karena memungkinkan identifikasi lebih mudah terhadap *bug* atau *error* yang mungkin terjadi. Terdapat dua pendekatan dalam mendefinisikan tingkat kecanggihan pengembangan dalam pengujian perangkat lunak. Pada tingkat terendah, pengujian dianggap serupa dengan *debugging*, namun pada tingkat yang lebih tinggi, pengujian diarahkan pada pola pikir yang bertujuan untuk memaksimalkan sistem.

Dalam rangka menyediakan pengalaman yang adaptif kepada pengguna, *Website* adalah sekumpulan halaman HTML yang dapat diakses melalui peramban web, dan dapat

disesuaikan dengan berbagai perangkat seperti *smartphone*, tablet, dan *gadget* tanpa perlu mengunduh aplikasi terlebih dahulu. Pendekatan "*develop once run everywhere*" telah menjadi prinsip dalam pengembangan *website*, memungkinkan operasionalitasnya di berbagai platform dengan sekali pengembangan. Untuk mengembangkan *website*, biasanya digunakan teknologi *web* terbuka seperti HTML, CSS, dan JavaScript. Kemudian, *website* ini dapat di-*host* pada server web yang ada dan diakses melalui *URL* standar melalui peramban *web* (Rizal, 2023). *Unit testing* merupakan salah satu metode pengujian perangkat lunak yang fokus pada pengujian bagian terkecil atau unit dari kode. Hal ini menjadi keterampilan yang penting bagi para pengembang perangkat lunak, karena dapat membantu dalam mengurangi kemungkinan adanya *bug* pada kode (Kua, 2019). Proses *unit testing* dilakukan seiring dengan pembuatan kode sistem untuk memastikan bahwa kode tersebut sesuai dengan desain yang telah direncanakan sebelumnya. Langkahlangkah dalam *unit testing* meliputi penulisan *test case* untuk setiap fungsi dan metode yang ada, sehingga memungkinkan identifikasi dan perbaikan kesalahan secara cepat. Jika setiap unit telah diuji dan berfungsi dengan baik, maka pengujian perangkat lunak dianggap telah memenuhi persyaratan. Dalam metode ini, teknik *white box testing* sering digunakan untuk memeriksa jalur-jalur tertentu guna memastikan cakupan jalur yang lengkap dan deteksi *error* yang maksimum (Jatnika dan Irwan, 2019). Proses *unit testing* adalah tahap implementasi perancangan sistem menjadi kumpulan program atau unit program, dan setiap unit program akan diuji untuk memastikan kesesuaian dengan spesifikasinya.

Penulisan ini merupakan bagian dari penelitian Perancangan Aplikasi *Petshop* Berbasis Web yang bertujuan untuk menerapkan metode *unit testing* pada aplikasi tersebut. Pengujian metode *unit testing* bertujuan untuk menemukan *error*, sehingga pengembang dapat menindak lanjuti *error* tersebut selama pengembangan aplikasi *petshop*.

METODE PENELITIAN

Metodologi yang diterapkan dalam penyusunan penulisan agar dalam tahap pengujian dapat terarah sesuai rencana dan mendapatkan hasil yang diharapkan antara lain:

1. Studi Literatur Metode studi literatur diterapkan dengan mengacu pada berbagai sumber seperti buku, jurnal, internet, dan sumber lain yang relevan dengan topik penulisan ini. Pendekatan ini juga digunakan sebagai dasar teoritis dalam penulisan ini.
2. Analisis Analisis kebutuhan melibatkan proses definisi dan penentuan kebutuhan spesifik dari aplikasi yang akan diuji.
3. Pengembangan Perangkat Lunak Pengembangan dilakukan dengan mengikuti langkah-langkah yang diperlukan untuk membangun dan mengembangkan sebuah sistem, serta memilih bagian-bagian tertentu yang akan dijadikan fokus dalam pengembangan perangkat lunak.
4. Penerapan *Unit Testing* Hasil produk kemudian diuji melalui metode unit testing menggunakan framework PHP Unit. Dari tahap pengujian ini, dapat diketahui apakah aplikasi sudah sesuai dengan desain yang telah direncanakan sebelumnya atau belum.
5. Evaluasi Evaluasi menyeluruh terhadap seluruh kegiatan dalam penelitian dilakukan untuk dapat mencapai kesimpulan yang sesuai berdasarkan hasil unit testing.

HASIL DAN PEMBAHASAN

Dalam penelitian ini, penerapan *unit testing* menggunakan PHPUnit.

PHPUnit dipilih karena aplikasi ini dikembangkan menggunakan bahasa pemrograman PHP. Kemudian pengujian dilakukan menggunakan metode *white box testing* karena pengujian dilakukan bertahap dan melihat alur logika dengan menghitung jumlah *statement* dengan cara melihat *source code* pada sistem yang dibuat. Teknik *statement coverage* dipilih sebagai pengukur tingkat keberhasilan pengujian. Teknik ini dilakukan dengan menjalankan data uji yang mencakup semua *statement (source code)* yang dijalankan agar mencapai nilai 100% pada *statement*.

Tahapan pengujian dilakukan dengan memilih fungsi yang memiliki percabangan, kemudian menghitung jumlah *statement* dan kondisi yang sudah ditentukan pada setiap fungsi. Karena keterbatasan dalam penulisan di dalam laporan Tugas Akhir ini maka hanya beberapa fungsi pada bagian Model yang akan disampaikan.

Tabel 3.1 Daftar Fungsi Model

Nama File Model	Nama Fungsi	Jumlah <i>State ment</i>	Jumlah Kondisi
Login_model	+login()	3	0
	+is_user_exists()	7	1
	+ get_role()	2	1
	+ get_password()	2	1
	+logged_user_id()	2	1
Register_model	+register_user()	3	2
	+register_customer()	3	0
Contact_model	+register_contact()	4	1
Customer_model	+data()	5	1
	+is_coupon_exists()	2	1
	+is_coupon_active()	2	2
	+is_coupon_expired()	4	3
	+get_coupon_credit()	4	2
	+get_coupon_id()	3	1
Product_model	+get_all_products()	2	0
	+best_deal_product()	5	1
	+is_product_exists()	2	1

	+product_data ()	3	0
	+related_products()	2	1
	+create_order ()	3	0
	+create_order_items()	2	0
Review_model	+count_all_reviews()	2	0
	+get_all_reviews()	2	3
	+is_review_exists()	2	1
	+review_data()	2	1

HASIL PENGUJIAN

Pengujian sistem menggunakan *Unit testing* yaitu PHPUnit dilakukan untuk mengetahui seberapa baik program dapat terhindar dari kesalahan seperti kesalahan logika dan asumsi pada eksekusi jalur yang tidak seharusnya.

Berdasarkan pengujian yang telah dilakukan, diperoleh bahwa fungsi bagian Model pada aplikasi *petshop* telah berhasil dilakukan dan aplikasi tersebut berjalan dengan baik sesuai dengan proses bisnis yang telah dirancang, sehingga sistem siap digunakan oleh pengguna. Terdapat 5 fungsi yang diuji yaitu fungsi `is_user_exists()`, `register_user()`, `register_contact()`, `data()`, dan `is_product_exists()`. Pengujian dilakukan dengan mengubah *source code* menjadi *flowgraph* untuk menentukan jalur yang dapat dilewati sebagai alur pengujian. Alur tersebut digunakan untuk mendapatkan *output* sesuai dengan yang diharapkan dan akan digunakan sebagai dasar pengujian. Setelah itu, dihitung tingkat keberhasilannya dengan menggunakan teknik *statement coverage* untuk melihat persentasi keberhasilan sehingga diperoleh hasil pengujian adalah 100%.

Hasil tingkat keberhasilan pengujian dapat dilihat pada Tabel 3.11 dimana telah dilakukan pengujian untuk lima fungsi dan setiap pengujian pada akhirnya memperoleh nilai 100%.

Tabel 3.2 Tingkat Akurasi Pengujian

Fungsi yang diuji	Jalur	TC	Jumlah PM Berhasil
login()	3	TC01 -1	4/10 x 100% = 40%
	2	TC01 -2	7/10 x 100% = 70%
	1	TC01 -3	10/10 x 100% = 100%
register_user()	3	TC02 -1	2/3 x 100% = 66.7%

	2	TC02 -2	3/3 x 100% = 100%
	1	TC02 -3	3/3 x100% = 100%
register_contact()	2	TC03 -1	4/4 x 100% = 100%
	1	TC03 -2	4/4 x 100% = 100%
data()	2	TC04 -1	4/5 x 100% = 80%
	1	TC04 -2	5/5 x 100% = 100%
is_product_exist()	2	TC05 -1	2/2 x 100% = 100%
	1	TC05 -2	2/2 x 100% = 100%

Berdasarkan Tabel 3.12 telah dilakukan pengujian *unit testing* dengan tujuan menguji unit terkecil dari aplikasi. Pada lima fungsi yang dipilih, semua fungsi yang diuji berhasil memperoleh nilai *coverage* sebesar 100% menggunakan teknik *statement coverage*. Pengujian *white box testing* dilakukan untuk memeriksa jalur yang ditentukan dan memastikan tidak terjadi error. Pada tahap pertama pengujian, terdapat tiga pengujian yang tidak mendapatkan nilai 100%, dikarenakan belum menjalankan keseluruhan *statement*. Kemudian dilakukan pengujian selanjutnya, yaitu dengan jumlah *statement* yang telah ditambahkan. Tahapan tersebut dilakukan hingga semua *statement* berhasil dijalankan. Semua fungsi yang memperoleh nilai 100% dikatakan berhasil karena telah melewati semua *statement* dan kondisi logis pada *source code* yang telah dieksekusi dengan minimal *test case* dijalankan dari jalur yang diperoleh melalui perhitungan dengan menggunakan metrik *cyclomatic complexity*.

KESIMPULAN

Penerapan *unit testing* pada aplikasi web *petshop* telah dilakukan menggunakan *framework* PHP. Metode pengujian yang digunakan adalah *white box testing* dengan teknik *statement coverage* untuk mengukur tingkat keberhasilan. *Unit testing* dengan PHP dilakukan pada setiap unit terkecil dalam aplikasi. Persentase keberhasilan pengujian dihitung berdasarkan teknik *statement coverage*, dimana setiap *test case* dieksekusi minimal satu kali dan harus melalui semua pernyataan kode hingga mencapai tingkat keberhasilan 100%. Pengujian pada web lain seperti pada *web mobile* bisnis jasa laundry dan web aplikasi pemesanan kantin *online* menggunakan metode yang sama juga berhasil mendapatkan hasil serupa.

SARAN

Dalam pengembangan aplikasi *petshop* dengan pengujian *unit testing*, terdapat beberapa saran yang dapat diberikan. Pertama, penting untuk melakukan peningkatan fitur dan perbaikan tahap demi tahap guna mengembangkan aplikasi web *petshop* ini yang saat ini masih bersifat sederhana. Kedua, pada pengembangan selanjutnya, diharapkan mampu

melakukan pengujian pada bagian kelas *view* dan *controller* dalam aplikasi. Terakhir, penelitian selanjutnya dapat mencoba metode lain untuk menguji tingkat keberhasilan sehingga hasilnya bisa dibandingkan dengan metode sebelumnya secara lebih akurat.

DAFTAR PUSTAKA

- Asthana, A. dan Olivieri, J. (2009). Quantifying Software Reliability and Readiness. IEEE International Workshop Technical Communications Quality and Reliability. Hlm 16.
- Anna. (2016). Aplikasi Kamus Bahasa Indonesia – Bugis Berbasis Web Dengan Metode Sequential Search. *Repository Universitas Bina Sarana Informatika (RUBSI)*, 3(September), pp 246–258.
- Bani, A. A. (2015). Implementasi Quality Assurance dalam Pengembangan Mutu Sumber Daya Manusia di Fakultas Agama Islam Universitas Muhammadiyah Malang. Yogyakarta: Program Studi Magister Manajemen Pendidikan Islam
- Darmeswara, B. (2013). Rekayasa Perangkat Lunak. Rancang Bangun E-commerce Pada Indo Mandiri Computer Semarang, 44-46.
- Gunawan, K. D., & Saputra, D. (2016, Mei). *Testing dan Implementasi. White box Testing*, 1- 5.
- Ismanto, Firman H., Kristinanti. (2020). Pemodelan Proses Bisnis Menggunakan Business Process Modelling Notation (BPMN) (Studi Kasus Unit Penelitian dan Pengabdian Kepada Masyarakat (P2KM) Akademi Komunitas Negeri Putra Sang Fajar Blitar). *Jurnal Riset dan Konseptual Vol 5, No 1*
- Jatnika, H., & Irwan, S. (2010). *TESTING DAN IMPLEMENTASI SISTEM*. Jakarta: hendrajatnika.web.id.
- Kartanti, L. L. (2015). Pengembangan dan Analisis Kualitas Sistem Administrasi Laboratorium Kompetensi Keahlian TKJ di SMK Negeri 1 Klaten Berbasis Web. Universitas Negeri Yogyakarta
- Khan, M. E., & Khan, F. (2012). A Comparative Study of *White box*, *Black box* and *Grey Box Testing* Techniques. *Internasional Jurnal of Advanced Computer Science and Applications*.
- Kua, P. (2019). *Unit Testing*. Oracle Australian Development Centre Oracle Corporation. Oracle Corporation
- Kurniansyah, R., & Hidayat, Y. A. (2016). Sistem Informasi. Pengujian Aplikasi Inventory Tufan Fish Farm menggunakan metode *white box testing* dan *black box testing*, 5-6.
- Kusumaningtiyas, C., Fitria, & Diputra, Y. (2016). Thesis Rendi. Perancangan Model Bisnis "MI.MU" sebagai ECommerce pada Produk Musik dan Derivatifnya (Music Destination Platform), pp. 1617.
- Mohd. Ehmer K. & Farneena K. (2012). A Comparative Study of White Box, Black Box and Grey Box Testing Techniques. *International Journal of Advanced Computer Science and Applications*, Vol. 3, No.6
- Muhamamad, A. R, Yunanto, Yoga, Caca A. H, Ahmad A. Y. R (2021). Implementasi Unit Testing Menggunakan Metode Test-First Development, Vol. 7 No. 1

- Padang, I. A., Hanson (2020). *Automated Testing Pada Website Marketplace Dengan PHPUnit*. Yogyakarta: Department of Informatics
- Pressman, R. S. (2010). *Software Engineering A Practitioner's Approach, Seventh Edition*. New York: McGraw-Hil.
- Rosa, A. S., & Shalahuddin. (2013). *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika.
- Shi, M. (2010). *Software Functional Testing from the Perspective of Business Practice*. Berlin: Canadian Center of Science and Education.
- Soeharto, I. (1995). *Manajemen Proyek Dari Konseptual Sampai Operasional, Jilid 2*. Jakarta: Erlangga.
- Sukamto, R.A., & Shalahuddin, M. (2015). *Kolaborasi Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Supono, dan Virdiandry Putratama. (2016). *Pemograman Web Dengan Menggunakan PHP dan Framework Codeigniter*. Yogyakarta: Deepublish (Grup Penerbitan CV Budi Utama)
- Yudho Y. (2013). *Pengantar BPMN Business Process Modeling Notation*. Bandung: Komunitas eLearning IlmuKomputer.Com
- Yuhefizar. (2013). *Cara Mudah & Murah Membangun & Mengelola Website*. Yogyakarta: Graha Ilmu.
- Yunisa, R. (2018). *Perbandingan 2 Teknik White box Testing (Studi Kasus: Sistem Informasi Reporting Community TB-HIV 'Aisyiyah Tanggamus)*. Yogyakarta: Informatics Engineering