



Pengujian Fungsional Aplikasi Pembayaran Zakat untuk Memvalidasi Proses Pembayaran dengan Tepat

Andri Elistiawan¹, Ade Dwi Putra¹, Muhammad Kemal Qodrat¹, Nisan Yedidiya Sorayana Mendrofa¹, Aries Saifudin^{1*}

¹Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia

E-mail: ¹andryapp2006@gmail.com, ²adedwiputra16@gmail.com, ³muhammadkemalq@gmail.com, ⁴nisanyedidya@gmail.com, ^{5*}aries.saifudin@unpam.ac.id

Abstrak–Aplikasi Pembayaran Zakat merupakan solusi digital yang bertujuan untuk memudahkan masyarakat dalam membayar zakat sesuai prinsip keagamaan. Namun, kualitas dan keandalan proses pembayaran menjadi aspek kritis yang harus dipastikan. Penelitian ini membahas pengujian fungsional pada Aplikasi Pembayaran Zakat dengan fokus pada memvalidasi proses pembayaran. Metode pengujian yang diusulkan mencakup black-box testing untuk memastikan keakuratan perhitungan zakat dan keamanan transaksi. Hasil pengujian diharapkan dapat memastikan bahwa aplikasi berfungsi sebagaimana mestinya dan dapat memberikan pengalaman pengguna yang optimal dalam proses pembayaran zakat.

Kata Kunci: Aplikasi Pembayaran Zakat; Pengujian Fungsional; Black-box Testing; Keamanan Transaksi;

Abstract–The Zakat Payment Application is a digital solution that aims to make it easier for people to pay zakat according to religious principles. However, the quality and reliability of the payment process is a critical aspect that must be ensured. This research discusses functional testing on the Zakat Payment Application with a focus on validating the payment process. The proposed testing method includes black-box testing to ensure the accuracy of zakat calculations and transaction security. It is hoped that the test results will ensure that the application functions as it should and can provide an optimal user experience in the Zakat payment process.

Keywords: Zakat Payment Application; Functional Testing; Black-box Testing; Transaction Security;

1. PENDAHULUAN

Pengembangan perangkat lunak merupakan proses kompleks yang membutuhkan perhatian khusus terhadap kualitas produk akhir. Salah satu aspek kritis dalam memastikan kualitas perangkat lunak adalah melalui pengujian fungsional yang teliti. komputer sudah merupakan kebutuhan yang sangat penting untuk kelancaran aktifitas di suatu badan lembaga atau organisasi.(Ramadan et al., 2021). Pengujian memainkan peran penting dalam menjamin bahwa perangkat lunak memenuhi spesifikasi, berfungsi sebagaimana mestinya, dan dapat diandalkan oleh pengguna akhir. Dalam konteks aplikasi pembayaran zakat, pengujian fungsional membantu memastikan bahwa proses pembayaran dilakukan dengan benar, akurat, dan sesuai dengan prinsip zakat yang berlaku.

Zakat merupakan suatu ibadah yang memiliki nilai sosial yang tinggi. Selain itu, zakat juga memberi dampak positif terhadap kesejahteraan masyarakat.(Hidayat & Mukhlisin, 2020). Penelitian ini difokuskan pada pengujian fungsional aplikasi pembayaran zakat yang telah dikembangkan. Aplikasi ini dirancang untuk memudahkan masyarakat dalam melakukan pembayaran zakat secara digital. Aplikasi ini mencakup fitur-fitur seperti pemberdayaan zakat, pemilihan jenis zakat, dan proses pembayaran yang harus diuji secara menyeluruh untuk memastikan keakuratan dan keandalannya.

Pengelolaan zakat berbasis manajemen dapat dilakukan dengan asumsi dasar bahwa semua aktivitas yang terkait dengan zakat dilakukan secara profesional.(Hafiz, 2021). Proses pembayaran zakat yang mengandalkan aplikasi dapat menimbulkan berbagai masalah apabila terjadi kesalahan dalam perhitungan atau pelaksanaan transaksi. Kesalahan perhitungan zakat bisa mengakibatkan dana yang seharusnya diberikan tidak sesuai dengan ketentuan agama, mengganggu keseimbangan distribusi zakat yang seharusnya adil. Di samping itu, keamanan transaksi juga menjadi masalah serius jika ada kesalahan keamanan, informasi keuangan pribadi pengguna dapat terancam, membuka pintu bagi potensi pencurian identitas dan penyalahgunaan dana. Dalam situasi yang sangat tidak diinginkan, kesalahan dapat mengakibatkan kerugian finansial bagi pengguna,



mengganggu kepercayaan pada platform pembayaran zakat, dan bahkan dapat mencoreng reputasi organisasi yang mengelolanya.

Pengujian fungsional aplikasi pembayaran zakat akan menggunakan metode black-box testing untuk memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang telah ditentukan. Pengujian ini akan mencakup pengujian fungsionalitas, keamanan, dan keandalan aplikasi. Pengujian fungsionalitas akan memeriksa keamanan dan mengevaluasi apakah data keuangan pengguna diamankan dengan baik selama proses pembayaran. Pengujian keandalan akan memastikan bahwa aplikasi dapat melakukan proses pembayaran dengan lancar dan dapat diandalkan pada kondisi yang berbeda.

Rencana penelitian ini akan mencakup langkah-langkah rinci untuk melaksanakan pengujian fungsional aplikasi pembayaran zakat. Langkah-langkah tersebut meliputi identifikasi fungsionalitas yang harus diuji, perencanaan pengujian, penyusunan kasus uji, pelaksanaan pengujian, dokumentasi hasil pengujian, dan analisis data. Setiap langkah akan dilaksanakan secara terstruktur untuk memastikan pengujian yang komprehensif dan akurat, sehingga aplikasi pembayaran zakat dapat beroperasi dengan optimal dan memberikan manfaat yang diinginkan.

2. METODE

2.1 Metode Pengujian *Black-box Testing*

Metode yang kami terapkan dalam pengujian aplikasi ini adalah *Black-box Testing*. *Black-box Testing* merupakan metode pengujian fungsional yang memeriksa input dan output tanpa memperhatikan struktur atau logika internal dari aplikasi. Kami memilih metode ini karena memungkinkan kami untuk fokus pada fungsionalitas aplikasi tanpa perlu memahami kompleksitas internal dari aplikasi. Dengan menggunakan metode ini, kami dapat memastikan bahwa aplikasi berfungsi dengan baik.

2.2. Pembuatan *Test Case*

a. Identifikasi Fungsionalitas Utama

Kami memulai dengan mengidentifikasi fungsionalitas utama dari aplikasi pembayaran zakat. Fungsionalitas ini mencakup pemilihan jenis zakat, proses pembayaran, dan pemberdayaan masyarakat.

b. Penyusunan *Test Case*

Setelah mengidentifikasi fungsionalitas, kami menyusun berbagai test case yang mencakup skenario uji untuk setiap fungsionalitas. Setiap test case memiliki input yang ditentukan, tindakan yang dilakukan pada aplikasi, dan output yang diharapkan. Data uji dipilih untuk mencakup berbagai kondisi termasuk input valid, tidak valid, batas atas, dan batas bawah.

c. Penyusunan Skrip Uji

Test case yang telah dibuat kemudian diintegrasikan ke dalam skrip uji yang akan dijalankan pada aplikasi. Setiap skrip uji mengacu pada test case yang sesuai untuk memastikan bahwa setiap fungsionalitas diuji sesuai dengan skenario yang telah ditentukan.

2.3 Desain Pengujian

a. Halaman awal

Halaman ini merupakan halaman utama yang berisi terdapat form pembayaran zakat atau infaq di halaman ini pengguna website yang akan melakukan infaq dan zakat diwajibkan terlebih dahulu untuk memilih jenis transaksi beserta dengan data diri dari pengguna website yang akan melakukan pembayaran zakat atau pemberian infaq dan shodaqoh.



JRIIN: Jurnal Riset Informatika dan Inovasi
Volume 1, No. 4, Desember 2023
ISSN 3025-0919 (media online)
Hal 999-999



Gambar 1. Halaman Awal

b. Halaman isi data diri

Halaman ini pemberi zakat infaq dan shodaqoh diwajibkan untuk mengisi data diri lebih lanjut seperti nomer telepon dan juga nominal uang yang akan diberikan.

Gambar 2. Halaman isi data diri

c. Halman pilih jenis pembayaran

Halaman ini pemberi zakat infaq atau shodaqoh di persilahkan untuk memilih metode pembayaran yang akan digunakan, jenis pembayaran di sini termanage semuanya di halaman administrator

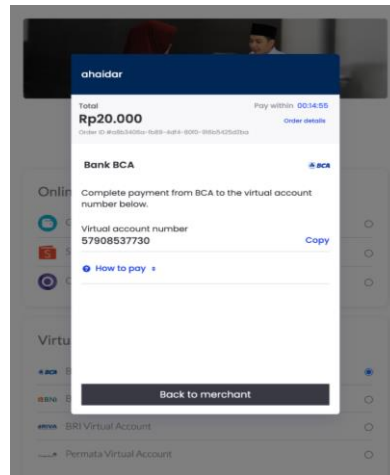
Gambar 3. Halaman pilih jenis pembayaran



JRIIN: Jurnal Riset Informatika dan Inovasi
Volume 1, No. 4, Desember 2023
ISSN 3025-0919 (media online)
Hal 999-999

d. Halaman pembayaran

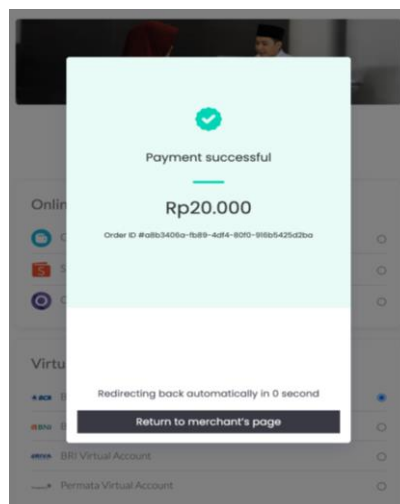
Halaman ini pembayar akan menerima petunjuk mengenai tata cara pembayaran berdasarkan jenis pembayaran yang dipilih, pembayaran di sini sudah terintegrasi dengan payment gateway midtransa sehingga nanti jika pihak masjid ingin menambahkan jenis pembayaran baru yang digunakan untuk mengimplementasikan jenis pembayaran tersebut tidak terlalu susah.



Gambar 4. Halaman pembayaran

e. Halaman sukses pembayaran

Setelah pengguna berhasil membayar tagihan yang diberikan, maka akan muncul suatu halaman yang menginformasikan bahwa pembayaran telah berhasil dikirim ke rekening tujuan.



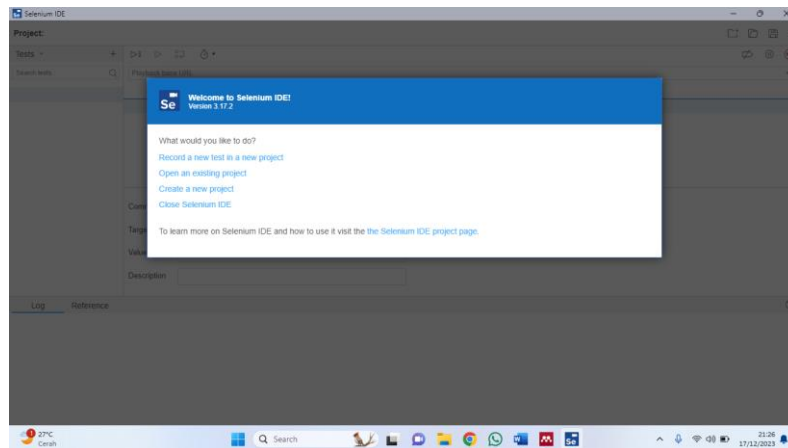
Gambar 5. Halaman sukses pembayaran

3. ANALISA DAN PEMBAHASAN

3.1 Analisa

a. Membuat Proyek Baru

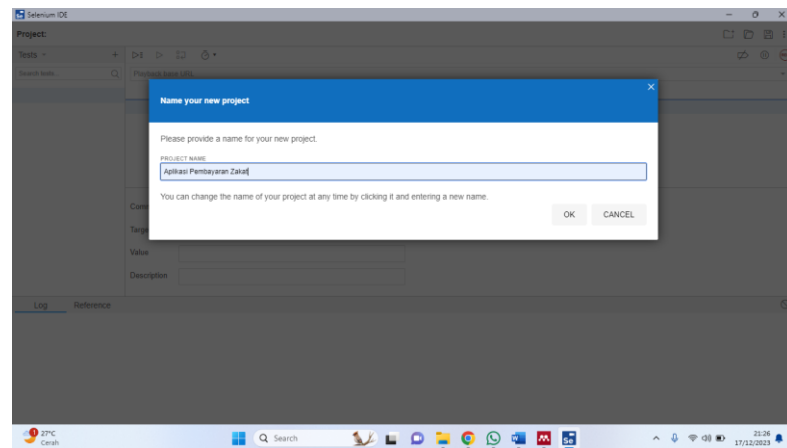
Untuk menjalankan proyek baru pada Selenium IDE pilih opsi record a new test in a new project. Fungsi dari opsi tersebut untuk melakukan test secara otomatis dengan recording dari fitur selenium IDE.



Gambar 7. Memilih project baru

b. Membuat nama proyek pengujian

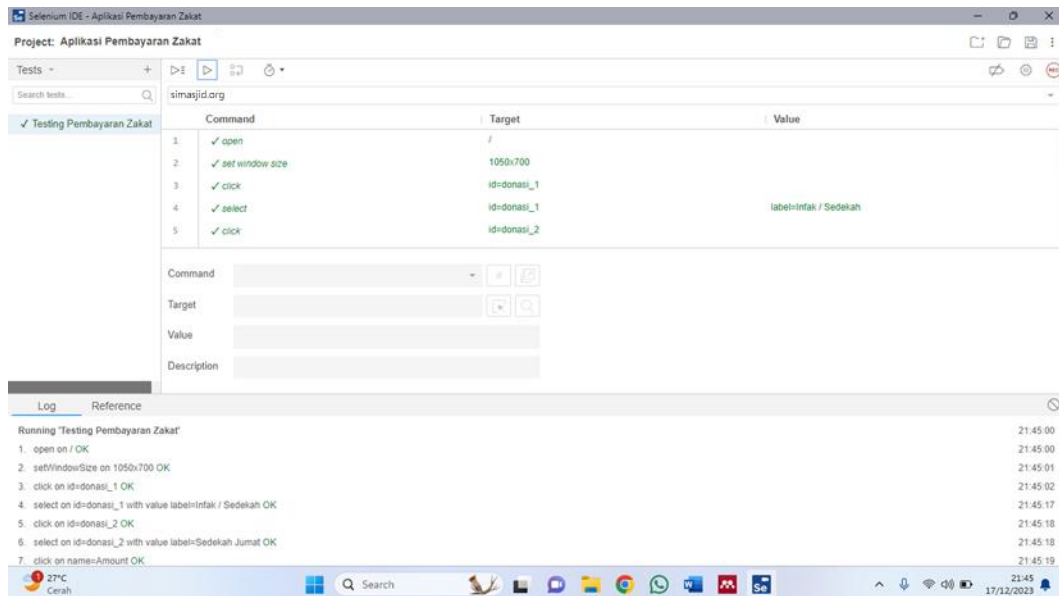
Langkah kedua adalah membuat nama proyek uji yang jelas dan deskriptif. Nama proyek ini berfungsi sebagai panduan utama untuk menjalankan serangkaian pengujian pada aplikasi web tertentu.



Gambar 6. Membuat nama proyek pengujian

c. Menjalankan Section pengujian, dan memulai pembayaran zakat

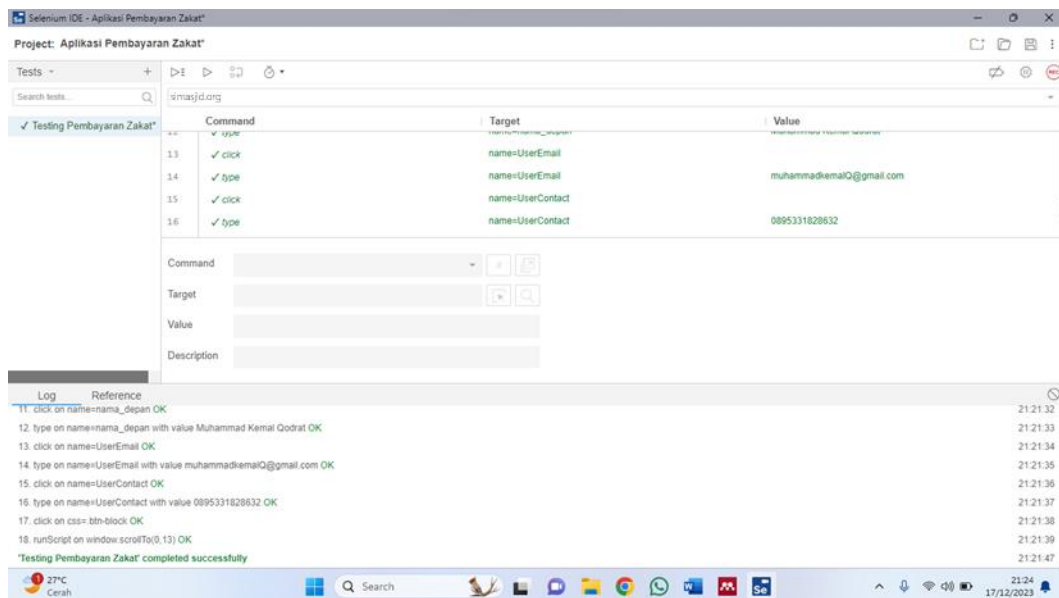
Pada tahap ini, pengujian dilakukan dalam menu pembayaran zakat dimana user mengisi semua kolom yang diperlukan untuk data pembayaran zakat.



Gambar 8. Pengujian terhadap aplikasi

d. Berikut hasil akhir pengujian setelah semuanya berhasil di uji

Pada bagian uji coba dilakukan pada saat user sudah mengisi semua data yang diperlukan dan sudah sesuai, kemudian sudah melakukan pembayaran maka transaksi telah selesai. Dan dalam tahap pengujian ini tidak ada error dalam sistem ini.



Gambar 8. Hasil Pengujian aplikasi

3.2 Pembahasan

Hasil pengujian integrasi memverifikasi bahwa aplikasi pembayaran zakat terintegrasi dengan baik menggunakan sistem keuangan yang terkait. Integrasi yang lancar adalah kunci untuk memastikan bahwa proses transaksi terhubung dengan benar dan data disinkronkan dengan tepat antara aplikasi dan sistem keuangan.



Hasil pengujian performa menunjukkan bahwa aplikasi mampu menanggapi dengan baik bahkan dalam situasi beban pengguna yang tinggi. Kinerja yang baik pada beban tinggi adalah indikator bahwa aplikasi dapat menangani volume transaksi yang besar tanpa mengalami penurunan responsivitas.

Hasil pengujian performa menunjukkan bahwa aplikasi mampu menanggapi dengan baik bahkan dalam situasi beban pengguna yang tinggi. Kinerja yang baik pada beban tinggi adalah indikator bahwa aplikasi dapat menangani volume transaksi yang besar tanpa mengalami penurunan responsivitas.

4. KESIMPULAN

Contain statements answering issues in the previous section and future work of the research.

Penerapan metode black-box testing sebagai pendekatan pengujian pada aplikasi pembayaran zakat telah memberikan hasil yang positif. Berdasarkan hasil pengujian fungsionalitas, keamanan transaksi, integrasi, dan performa, dapat diambil beberapa kesimpulan penting:

a. Fungsionalitas Terjamin:

Metode pengujian fungsionalitas memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi, termasuk proses penghitungan zakat dan pemilihan jenis zakat, yang merupakan inti dari aplikasi pembayaran zakat.

b. Keamanan Terjaga:

Pengujian keamanan transaksi membuktikan bahwa sistem keamanan aplikasi efektif dalam melindungi informasi keuangan pengguna selama proses pembayaran, menjamin transaksi yang aman.

c. Integrasi yang Lancar:

Hasil pengujian integrasi menegaskan bahwa aplikasi terintegrasi dengan baik dengan sistem keuangan terkait, memastikan keterhubungan yang lancar dan sinkronisasi data yang tepat.

d. Performa Optimal:

Pengujian performa menunjukkan bahwa aplikasi mampu menanggapi dengan baik bahkan pada situasi beban pengguna yang tinggi, menegaskan kinerja aplikasi yang optimal.

Penerapan metode black-box testing memiliki manfaat besar dalam memverifikasi dan memvalidasi fungsionalitas aplikasi secara menyeluruh tanpa memperhatikan kompleksitas internal. Namun, perbaikan dan pemeliharaan terus diperlukan untuk memastikan bahwa aplikasi tetap memenuhi standar kualitas dan keamanan seiring dengan perubahan teknologi dan tuntutan pengguna.

Dengan demikian, hasil dan analisis pengujian ini memberikan landasan yang kuat untuk pengembangan selanjutnya, memungkinkan perbaikan yang dibutuhkan dan memastikan aplikasi pembayaran zakat dapat memberikan layanan yang andal dan aman bagi pengguna.



JRIIN: Jurnal Riset Informatika dan Inovasi
Volume 1, No. 4, Desember 2023
ISSN 3025-0919 (media online)
Hal 999-999

REFERENCES

- Cholifah, W. N., Yulianingsih, & Sagita, S. M. (2018). Pengujian Black Box Testing Pada Aplikasi Action & Strategy Berbasis Android Dengan Teknologi Phonegap. *Jurnal String*, 206 - 210.
- Fauzi, M. R., Matahari, & Ramadhani, I. A. (2021). Perancangan Sistem Informasi Pengelolaan Zakat Fitrah Pada Badan Amil Zakat Masjid Miftahul Jannah Kelurahan Malasilen Berbasis J2SE (Java Platform Standard Edition). *Jurnal PETISI (Pendidikan Teknologi Informasi)*, 41 - 47.
- Rosmiyati. (2021). Analisis dan Pengujian Sistem Menggunakan Black Box Testing Equivalence Partitioning. *Jurnal Sains Komputer dan Teknologi Informasi*, 56 - 63.
- Waliyansyah, R. R., Hermawan, G., & Herlambang, B. A. (2022). Sistem Informasi Pengelolaan Zakat Fitrah dan Donasi Pada Masjid Jami' Al Jannah Menggunakan Metode Rule Based Berbasis Android. *Walisongo Journal of Information Technology*, 33 - 42.
- Zulkarnain, Nasution, H., & Irwansyah, M. (2021). Rancang Bangun Aplikasi Pengelolaan Zakat, Infaq. dan Sadaqah Pada Yayasan Al Fatwa. *Jurnal Sistem dan Teknologi informasi*, 62 - 69.
- Hafiz, A. (2021). Analisis Aplikasi Pengolahan Zakat Online Dan Pengelolaan Zakat Berbasis Jaringan Virtual. *Manajemen Dakwah*, 9(April), h.39-40.
- Hidayat, A., & Mukhlisin, M. (2020). Analisis Pertumbuhan Zakat Pada Aplikasi Zakat Online Dompot Dhuafa. *Jurnal Ilmiah Ekonomi Islam*, 6(3), 675. <https://doi.org/10.29040/jiei.v6i3.1435>
- Ramadan, M., Muhammad, R., & Riansyah, F. A. (2021). Aplikasi Pengolahan Zakat Online Berbasis Web Baznas Depok. *Jurnal Manajemen Dakwah*, 8(1), 162–177. <https://doi.org/10.15408/jmd.v8i1.19898>