

PERAN QA DALAM MENINGKATKAN PENGALAMAN PENGGUNA APLIKASI DOMPET DUAFI: FOKUS PADA PERFORMA DAN FUNGSIONALITAS

Eko¹, Muhamad Ilham Fadilah², Muhammad Syukron³, Oki Saputra⁴, Aries Saifudin⁵

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

E-mail: ¹pangeranekko@gmail.com, ²ilhamkhaleed05@gmail.com, ³mhmsykr@gmail.com,
⁴okyy.saputra20@gmail.com, ⁵aries.saifudin@unpam.ac.id

Abstract

This research explores the role of Quality Assurance (QA) in enhancing the user experience of the Dompot Duafa application, with a specific focus on performance and functionality aspects. Dompot Duafa, as a technology-based financial application, plays a crucial role in facilitating users in managing their finances. QA is key to ensuring that this application not only functions well but also provides an optimal user experience. The research emphasizes the importance of employing thorough testing methods for performance and functionality to address potential software quality issues. It identifies software quality problems that may arise and describes the consequences that may occur if these issues are not addressed properly. The findings of this research provide insights into how the effective implementation of QA, through meticulous testing methods, can enhance the performance and functionality of Dompot Duafa, thereby delivering a more satisfying user experience.

Keywords : Quality Assurance; Performance; Functionality; Dompot Duafa

Abstrak

Penelitian ini membahas peran *Quality Assurance* (QA) dalam meningkatkan pengalaman pengguna aplikasi Dompot Duafa, dengan fokus khusus pada aspek performa dan fungsionalitas. Dompot Duafa sebagai aplikasi keuangan berbasis teknologi memiliki peran penting dalam memudahkan pengguna dalam mengelola keuangan mereka. QA menjadi kunci untuk memastikan bahwa aplikasi ini tidak hanya berfungsi dengan baik tetapi juga memberikan pengalaman pengguna yang optimal. Metode pengujian performa dan fungsionalitas menjadi fokus utama QA dalam memastikan kualitas aplikasi. Penelitian ini mengidentifikasi masalah kualitas perangkat lunak yang mungkin timbul dan menggambarkan konsekuensi yang mungkin terjadi jika masalah tersebut tidak ditangani dengan baik. Hasil penelitian ini memberikan wawasan tentang bagaimana penerapan QA secara efektif, melalui metode pengujian yang cermat, dapat meningkatkan performa dan fungsionalitas Dompot Duafa, sehingga memberikan pengalaman pengguna yang lebih memuaskan.

Kata Kunci : Kualitas asuransi; Pertunjukan; Kegunaan; Dompot Dhuafa

1. PENDAHULUAN

Pengujian perangkat lunak adalah proses menjalankan dan mengevaluasi sebuah perangkat lunak secara manual maupun otomatis untuk menguji apakah perangkat lunak sudah memenuhi persyaratan atau belum (Clune dan Rood, 2011) dan (Nakagawa dan Maldonado, 2011). Singkat kata, pengujian adalah aktivitas untuk menemukan dan menentukan perbedaan antara hasil yang diharapkan dengan hasil sebenarnya.

Sebuah pengujian perangkat lunak dapat dilakukan setelah perekrutan membangun implementasi dari suatu konsep abstrak perangkat lunak. Pengujian perangkat lunak pada dasarnya adalah bermaksud “membongkar” perangkat lunak yang telah terbangun[1].

Beberapa hal yang membuat pengujian menjadi keharusan adalah:

- a. Menemukan kesalahan/*bug* saat proses pengembangan produk dalam SDLC.
- b. Sebagai acuan dalam meningkatkan kepuasan user melalui produk yang handal / *reliability*.
- c. Produk yang berkualitas menciptakan rasa percaya diri / *confidence*.
- d. Menjamin kinerja suatu produk
- e. Dan memastikan perusahaan pemilik / *product owner* tetap berjalan serta meningkatkan *business value*[2].

Dalam pengembangan perangkat lunak, terutama aplikasi seperti Dompot Dhuafa yang menggunakan *Flutter* untuk bagian antarmuka pengguna dan *Nest JS* untuk bagian pengelolaan data. Terdapat beberapa obyek atau *software* yang perlu diuji. Berikut adalah beberapa obyek yang akan diuji: Antarmuka Pengguna (*UI*) dalam *Flutter*, Fungsionalitas *Front End*, Keamanan, Kinerja dan Kinerja *Back End*.

Kesalahan dalam aplikasi Dompot Dhuafa bisa menyebabkan beberapa kerugian serius, baik bagi yayasan itu

sendiri maupun bagi masyarakat yang menggunakan layanan mereka. Berikut beberapa dampak negatif yang mungkin terjadi jika terdapat kesalahan dalam aplikasi tersebut: Kehilangan Donasi, Pengelolaan Dana yang Tidak Efisien, Ketidakpatuhan Terhadap Regulasi, Kerugian Reputasi dan Ketidakmampuan untuk Membantu yang Membutuhkan.

Untuk menguji aplikasi Dompot Dhuafa beberapa metode pengujian yang umum digunakan dalam pengembangan perangkat lunak. Berikut adalah beberapa metode yang diusulkan: Pengujian Unit, Pengujian Fungsional, Pengujian *UI*/Antarmuka Pengguna dan Pengujian Keamanan.

Berikut adalah rencana penelitian atau pengujian yang dapat diterapkan untuk menguji aplikasi Dompot Dhuafa: Pengumpulan Persyaratan, Perancangan Kasus Uji, Pengembangan Lingkungan Uji, Implementasi Kasus Uji, Eksekusi Kasus Uji, Analisis Hasil Uji, Pemeliharaan dan Pemutakhiran dan Dokumentasi.

2. PENELITIAN YANG TERKAIT

Dalam jurnal ini, penelitian sebelumnya yang sejalan dengan pembahasan tentang penerapan Quality Assurance (QA) dalam pengembangan aplikasi perangkat lunak telah diidentifikasi. Berikut adalah penjelasan singkat mengenai penelitian tersebut dan bagaimana mereka berkaitan dengan penelitian yang dilakukan:

1. Implementasi *Continuous Integration/Continuous Delivery (CI/CD)* pada *Performance Testing DevOps* oleh Jaeni, S., N. A., dan Laksito, A. D. (2022). Penelitian ini

membahas penerapan metode CI/CD dalam pengujian performa di lingkungan DevOps. Hasil penelitian menunjukkan bahwa implementasi CI/CD dapat meningkatkan efisiensi dan efektivitas dalam pengujian performa aplikasi.

2. Peran Penting Testing dan Quality Assurance dalam Siklus Pengembangan Sistem oleh Mantik, H. (2014). Penelitian ini menyoroti pentingnya pengujian dan QA dalam siklus pengembangan sistem. Hasil penelitian mengungkapkan bahwa QA yang efektif dapat meningkatkan kualitas sistem dan mengurangi risiko bug serta masalah performa pada aplikasi.
3. Urgensi Pengujian pada Kemajemukan Perangkat Lunak dalam Multi Perspektif oleh Sulistyanto, H. (2017). Penelitian ini mengkaji pentingnya pengujian perangkat lunak dari berbagai perspektif. Hasil penelitian ini mengidentifikasi bahwa pendekatan pengujian multi-perspektif dapat memberikan cakupan pengujian yang lebih komprehensif.

Dalam penelitian ini, peran QA ditekankan dalam meningkatkan kualitas dan pengalaman pengguna dari aplikasi Dompot Duafa, dengan fokus pada pengujian performa dan fungsionalitas. Penelitian ini mengidentifikasi masalah kualitas perangkat lunak yang mungkin timbul dan menggambarkan konsekuensi yang mungkin terjadi jika masalah tersebut tidak ditangani dengan baik. Implementasi QA yang efektif, melalui metode pengujian yang cermat, dapat meningkatkan performa dan fungsionalitas aplikasi, sehingga

memberikan pengalaman pengguna yang lebih memuaskan.

3. METODOLOGI

Metodologi SDLC dengan model Agile ini dikembangkan untuk menangani permasalahan mengenai permintaan perubahan pada sistem oleh pelanggan, permasalahan mengenai biaya pengembangan yang besar, dan kurangnya efisiensi waktu ketika mengembangkan sebuah proyek. Metode ini dirancang untuk memfasilitasi penyelesaian proyek pengembangan sistem dengan waktu yang cepat[3].

Dalam model Agile, persyaratan dalam pengembangan sistem diuraikan menjadi bagian-bagian kecil yang nantinya dapat dikembangkan secara bertahap, dan akan dikembangkan secara iteratif. Sehingga, setiap bagian inkrementalnya akan dikembangkan melalui iterasi. Iterasi ini bertujuan agar proses pengembangan sistem menjadi lebih mudah dikelola dan dapat diselesaikan dengan waktu yang cepat, serta melibatkan pelanggan untuk memberikan feedback mengenai sistem yang dirancang. Langkah –Langkah atau tahapan pengembangan pada model Agile ini yakni:



Gambar. 1 Metode Agile

Ada beberapa jenis model yang terdapat pada metodologi pengembangan perangkat lunak Agile, yakni:

a) *Adaptive Software Development(ASD)*

ASD merupakan modifikasi dari model extreme programming, dan merupakan model Agile yang paling banyak digunakan. Model pengembangan ini digunakan untuk mengembangkan sistem yang bersifat jangka pendek[4].

b) *Continuous Integration(CI)*

Metode ini mampu melakukan integrasi sistem, menguji coba fungsionalitas sistem, dan memungkinkan penanaman aplikasi pada lingkungan produksi yang bisa dilakukan secara otomatis. CI memberikan beberapa

kelebihan yakni proses test otomatisasi, mempercepat proses *feedback* dan mengurangi konflik *environment* antara *developer* dan tim operasi. Dengan diterapkannya metode CI ini, sistem akan dikembangkan dengan lebih produktif dan efisien[5].

c) *Crystal Methods*

Metode ini merupakan suatu pendekatan pada metodologi Agile yang berfokus pada orang dan interaksi orang – orang tersebut ketika mengerjakan *project* daripada proses dan alat yang digunakan dalam pengembangan, sehingga prioritasnya yakni tim terlibat dengan pengembangan perangkat lunak, bukan proses yang dilakukan selama mengembangkan perangkat lunak tersebut.

d) *Dynamic Systems Development Method*

Metode DSDM menyediakan framework dalam pengembangan maupun pemeliharaan perangkat lunak atau sistem yang memiliki kendala waktu yang sedikit. Metode ini merupakan pendekatan *iterative* dan *incremental* yang mengutamakan keterlibatan user atau pelanggan.

e) *Extreme Programming*

Metode ini juga merupakan yang paling sering digunakan, menggunakan pendekatan yang berorientasi pada objek serta sasaran nya yakni tim dengan skala kecil sampai sedang/medium. Metode ini akan cocok digunakan apabila selama proses pengembangan sistemnya, dihadapkan dengan persyaratan kebutuhan sistem yang tidak didefinisikan dengan jelas atau detail, dan persyaratan kebutuhan sistem nya dapat berubah secara tiba-tiba.

f) *Feature Driven Development*

Metode FDD adalah yang pertama kali dikenalkan dalam metodologi Agile. Acuan pada metode ini adalah pengembangan sistem yang mudah diimplementasikan dan mudah dimengerti,

serta menerapkan teknik problem solving. Sama seperti metodologi Agilelainnya, metode ini berfokus pada kerjasama tim, pengelolaan permasalahan, kompleksitas dari proyek yang dibangun, serta komunikasi.

a. Kanban

Kanban merupakan sebuah metode untuk mendefinisikan, mengelola, dan meningkatkan layanan. Metode ini akan meningkatkan secara bertahap apapun yang dikerjakan. Metode ini memiliki dua fungsi utama yakni sebagai pengendalian produksi dan sarana peningkatan produksi.

b. *Lean Software Development*

Metode ini adalah sebuah pendekatan sistematis yang digunakan untuk mengembangkan sistem dan menghasilkan suatu sistem dengan kualitas tinggi yang kehandalannya telah terjamin, sehingga tidak akan terjadi kegagalan dalam penggunaan perangkat lunak tersebut.

c. *Rational Unified Process*

Metode RUP yakni metode pengembangan sistem informasi yang memiliki sifat berulang. Pada metode ini, ada tiga elemen dasar pada kerangka kerjanya, yakni: peran, kegiatan, serta artefak. Elemen dasar tersebut dipilih sesuai dengan kebutuhan proyek pengembangan sistem yang sedang dikerjakan.

d. *Scrum*

Metode ini bersifat terstruktur dan berfungsi untuk mendukung pengembangan proyek yang kompleks, metode ini mengedepankan kekuatan kolaborasi didalam sebuah tim, produk *incremental*, serta proses dari iterasi selama mengembangkan sistem demi mencapai hasil akhir yang baik.

Test Case

Test case adalah pengujian yang dilakukan berdasarkan beberapa masukan seperti kondisi dan hasil yang telah ditentukan sebelumnya [2]. Hasil dari test case yang telah dilakukan akan dibandingkan dengan hasil yang telah ditentukan sebelumnya. Jika terdapat perbedaan di antara keduanya maka akan dilakukan perbaikan pada kode program. Test case menjadi titik awal untuk melakukan pengujian setelah memasukan nilai-nilai inputan ke sistem. Setelah itu, akan didapat hasil yang definitif dan meninggalkan sistem di beberapa titik akhir atau juga dikenal sebagai postcondition eksekusi [7].

Berikut adalah contoh pembuatan test case berdasarkan metodologi dan hasil pembahasan di atas untuk aplikasi Dompot Duafa dengan fokus pada performa dan fungsionalitas:

1. Test Case untuk performa

1. Skenario: Memuat beranda aplikasi

1. Langkah-langkah:

1. Buka aplikasi Dompot Dhuafa.

2. Amati waktu yang dibutuhkan untuk memuat halaman beranda.

2. Ekspektasi:

Waktu muat tidak melebihi 3 detik pada koneksi jaringan yang stabil.

2. Test Case untuk Fungsionalitas

1. Skenario : Membuat akun di aplikasi Dompot Dhuafa.

1. Langkah-langkah:

1. Buka aplikasi Dompot Dhuafa.

2. Pilih daftar, kemudian masukkan nama,

email, dan password baru anda.

3. Masukkan kode OTP yang masuk ke email anda.

4. Maka anda akan masuk ke halaman home aplikasi Dompot Dhuafa.

2. Ekspektasi:

Akun

2. Skenario : Donasi di aplikasi Dompot Dhuafa

1. Langkah-langkah:

1. Di halaman home, klik icon kemanusiaan atau icon wakaf.

2. Pilih campaign yang ingin anda donasikan.

3. Kemudian klik button 'Donasi Sekarang'.

4. Isi nominal yang ingin di donasikan.

5. Pilih metode pembayaran.

6. Bayar dengan metode pembayaran yang anda pilih.

7. Donasi berhasil.

2. Ekspektasi:

Donasi berhasil dibayar dan donasi tersalurkan ke campaign yang di donasikan.

4. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini memiliki implikasi yang signifikan dalam konteks pengembangan aplikasi perangkat lunak, terutama dalam hal memastikan kualitas dan pengalaman pengguna yang optimal. Dalam era digital yang semakin berkembang pesat, aplikasi perangkat lunak seperti Dompot Duafa memiliki peran yang sangat penting dalam memudahkan pengguna dalam mengelola

keuangan mereka, terutama dalam konteks keuangan yang inklusif dan berkelanjutan.

Dalam konteks ini, Quality Assurance (QA) memainkan peran yang krusial dalam memastikan bahwa aplikasi tersebut tidak hanya berfungsi dengan baik tetapi juga memberikan pengalaman pengguna yang memuaskan. Dengan fokus khusus pada aspek performa dan fungsionalitas, QA bertanggung jawab untuk melakukan pengujian yang cermat guna mengidentifikasi dan memperbaiki potensi masalah dalam aplikasi.

Metode pengujian performa dan fungsionalitas menjadi fokus utama dalam penelitian ini. Dengan menggunakan metode pengujian yang tepat, seperti pengujian unit, pengujian fungsional, pengujian UI/antarmuka pengguna, dan pengujian keamanan, QA dapat memastikan bahwa setiap bagian dari aplikasi Dompot Duafa berjalan sesuai dengan yang diharapkan.

Namun, dalam proses pengujian ini, beberapa masalah kualitas perangkat lunak dapat muncul, seperti bug atau kesalahan fungsional, kinerja yang lambat, atau bahkan masalah keamanan yang dapat membahayakan pengguna. Konsekuensi dari masalah tersebut dapat sangat merugikan, termasuk kehilangan donasi, pengelolaan dana yang tidak efisien, ketidakpatuhan terhadap regulasi, kerugian reputasi, dan ketidakmampuan untuk membantu yang membutuhkan.

Untuk mengatasi potensi masalah ini, penelitian ini memberikan wawasan tentang bagaimana penerapan QA secara efektif dapat meningkatkan performa dan fungsionalitas Dompot Duafa. Melalui penggunaan metode pengujian yang cermat dan sistematis, QA dapat membantu mengidentifikasi, mengisolasi, dan memperbaiki masalah kualitas perangkat lunak dengan cepat dan efisien.

Selain itu, penelitian ini juga memberikan panduan praktis untuk menerapkan metode pengujian yang efektif, termasuk proses pengujian yang meliputi pengumpulan persyaratan, perancangan kasus uji, pengembangan lingkungan uji, implementasi kasus uji, eksekusi kasus uji,

analisis hasil uji, pemeliharaan dan pemutakhiran, serta dokumentasi.

Dengan demikian, penelitian ini tidak hanya memberikan pemahaman yang mendalam tentang pentingnya QA dalam pengembangan aplikasi perangkat lunak, tetapi juga memberikan kontribusi nyata dalam meningkatkan kualitas dan pengalaman pengguna dari Dompot Duafa. Dengan implementasi yang tepat, hasil dari penelitian ini diharapkan dapat memberikan manfaat yang signifikan bagi pengguna dan pemilik aplikasi, serta mendorong perkembangan aplikasi yang lebih baik di masa depan.

5. KESIMPULAN

Kesimpulan dari penelitian ini adalah bahwa Quality Assurance (QA) memiliki peran penting dalam meningkatkan kualitas dan pengalaman pengguna dari aplikasi perangkat lunak, seperti Dompot Duafa. Dengan fokus pada pengujian performa dan fungsionalitas, QA dapat membantu mengidentifikasi dan memperbaiki masalah kualitas perangkat lunak yang mungkin timbul, seperti bug, kinerja yang lambat, atau masalah keamanan.

Metode pengujian yang efektif, seperti pengujian unit, pengujian fungsional, pengujian UI/antarmuka pengguna, dan pengujian keamanan, menjadi kunci dalam memastikan bahwa setiap bagian dari aplikasi berjalan sesuai yang diharapkan. Dengan menggunakan pendekatan yang sistematis dan cermat, QA dapat memastikan bahwa masalah-masalah ini diidentifikasi dan diperbaiki dengan cepat dan efisien.

Dampak dari masalah kualitas perangkat lunak dapat sangat merugikan, termasuk kehilangan donasi, pengelolaan dana yang tidak efisien, ketidakpatuhan terhadap regulasi, kerugian reputasi, dan ketidakmampuan untuk membantu yang membutuhkan. Oleh karena itu, implementasi yang tepat dari QA menjadi kunci dalam memastikan bahwa aplikasi seperti Dompot Duafa dapat memberikan pengalaman pengguna yang optimal dan memenuhi kebutuhan pengguna dengan baik.

Dengan demikian, penelitian ini tidak hanya memberikan pemahaman yang mendalam tentang pentingnya QA dalam pengembangan aplikasi perangkat lunak, tetapi juga memberikan panduan praktis untuk menerapkan metode pengujian yang efektif. Diharapkan bahwa hasil dari penelitian ini dapat memberikan manfaat yang signifikan bagi pengguna dan pemilik aplikasi, serta mendorong perkembangan aplikasi yang lebih baik di masa depan.

Daftar Pustaka

- [1] Arizona, N. D., Yulia, Y., & Jaka, O. (2022). Penerapan Metode ASD (Adaptive Software Development) Dalam Pembuatan Sistem Informasi Akuntansi Kas Masuk Dan Kas Keluar Pada CV.Bela Usaha. *Jurnal Sistem Informasi Akuntansi*, 3(1), 12–21. <https://doi.org/10.31294/justian.v3i1.1149>
- [2] Jaeni, J., S., N. A., & Laksito, A. D. (2022). Implementasi Continuous Integration/Continuous Delivery (Ci/Cd) Pada Performance Testing Devops. *Journal of Information System Management (JOISM)*, 4(1), 62–66. <https://doi.org/10.24076/joism.2022v4i1.887>
- [3] Mantik, H. (2014). Peran Penting *Testing* Dan *Quality Assurance* Dalam Siklus Pengembangan Sistem. *Jurnal Sistem Informasi Universitas Suryadarma*, 4(2), 75–79. <https://doi.org/10.35968/jsi.v4i2.7>
- [4] Prabowo, M. (2020). Metodologi pengembangan sistem informasi. LP2M Press IAIN Salatiga.
- [5] Sulistyanto, H. (2017). Urgensi Pengujian Pada Kemajemukan Perangkat Lunak Dalam Multi Perspektif. *Komuniti : Jurnal Komunikasi Dan Teknologi Informasi*, 6(1), 65–74. <https://journals.ums.ac.id/index.php/komuniti/article/view/2944>