

BAB II

TINJAUAN PUSTAKA

2.1. Pengujian Perangkat Lunak

Perangkat lunak telah menjadi komponen esensial dalam kehidupan sehari-hari. Kualitas perangkat lunak yang handal menjadi hal yang penting untuk menjamin kepuasan pengguna dan keberhasilan bisnis. Untuk mencapai hal tersebut, pengujian perangkat lunak menjadi suatu keharusan. Pengujian tidak hanya sekedar menemukan kesalahan, tetapi juga untuk memvalidasi bahwa perangkat lunak telah memenuhi persyaratan yang telah ditetapkan.

Pengujian perangkat lunak adalah serangkaian aktivitas yang dilakukan untuk mengevaluasi apakah suatu perangkat lunak memenuhi persyaratan yang telah ditentukan dan untuk mengidentifikasi kesalahan atau cacat. Pengujian yang efektif dapat membantu mengurangi risiko kegagalan sistem, meningkatkan kepuasan pengguna, dan menjaga reputasi pengembang [10].

Menurut [11], pengujian perangkat lunak menawarkan beberapa keuntungan penting, yaitu mengidentifikasi dan memperbaiki cacat, meningkatkan kualitas perangkat lunak, mencegah kegagalan sistem, meningkatkan kepuasan pengguna, dan menjaga reputasi pengembang.

Sejarah pengujian perangkat lunak telah mengalami perkembangan yang signifikan, dari pendekatan *ad-hoc* pada masa lalu hingga penggunaan metode dan alat yang lebih sistematis dan canggih. Seiring dengan semakin kompleksnya perangkat lunak, kebutuhan akan pengujian yang komprehensif juga semakin meningkat.

Pengujian perangkat lunak pada website SolusiLaundry memiliki tantangan dan kompleksitas tersendiri. Sebagai sistem perangkat lunak yang baru dikembangkan, pengujian terhadap sistem SolusiLaundry penting dilakukan untuk memastikan fungsionalitas dasar seperti pendaftaran pengguna, pemesanan laundry, dan perlu dipertimbangkan aspek-aspek lain seperti kinerja, keamanan, dan kompatibilitas dengan berbagai perangkat dan browser. Pengujian perlu

dilakukan pada saat proses pengembangan sistem informasi baru, dikarenakan sering kali terdapat masalah pada sistem dalam proses pengembangan perangkat lunak, termasuk kesalahan pada sistem.

2.2. Functional Testing

Pengujian perangkat lunak terbagi menjadi dua kategori utama: pengujian fungsional dan non-fungsional. Pengujian fungsional mencakup teknik *white box*, *black box*, dan *gray box testing* [4]. Sementara itu, pengujian non-fungsional meliputi aspek performa, waktu muat (loading time), keamanan, dan aspek teknis lainnya. Pada website SolusiLaundry, pengujian fungsional berperan penting dalam memverifikasi bahwa setiap fitur dan fungsi bekerja sesuai spesifikasi yang telah ditetapkan [12].

Dalam konteks aplikasi website Solusilaundy, pengujian fungsional mencakup verifikasi terhadap berbagai fitur seperti pendaftaran pengguna, pemesanan laundry, pemilihan jenis layanan, metode pembayaran, pelacakan pesanan, hingga fitur layanan pelanggan. Setiap fitur ini harus diuji secara menyeluruh untuk memastikan keakuratan, kelengkapan, dan konsistensi dalam menghasilkan output yang diharapkan.

Pentingnya pengujian fungsional pada aplikasi website solusi laundry tidak dapat diabaikan. Pengujian ini membantu mengidentifikasi cacat atau *bug* yang dapat mengganggu pengalaman pengguna. Selain itu, pengujian fungsional juga memastikan bahwa aplikasi memenuhi kebutuhan bisnis dan memenuhi harapan pengguna.

Penelitian [13], mengemukakan bahwa pengujian fungsional dapat meningkatkan kepercayaan pengguna terhadap kualitas aplikasi. Ketika pengguna merasa yakin bahwa aplikasi berjalan dengan baik, pengguna akan lebih cenderung menggunakan aplikasi tersebut secara berulang dan merekomendasikannya kepada orang lain.

Pengujian fungsional memiliki peran kritis dalam pengembangan website SolusiLaundry. Proses ini tidak hanya mengidentifikasi kesalahan teknis (technical defects), tetapi juga memastikan bahwa aplikasi memenuhi kebutuhan bisnis dan

ekspektasi pengguna. Menurut penelitian [13], implementasi pengujian fungsional yang menyeluruh berkorelasi positif dengan tingkat kepercayaan pengguna terhadap aplikasi.

Ketika pengguna memiliki keyakinan terhadap reliabilitas sistem, mereka cenderung menggunakan aplikasi secara berkelanjutan, memanfaatkan lebih banyak fitur yang tersedia, merekomendasikan layanan kepada pengguna potensial lainnya, memberikan umpan balik yang konstruktif untuk pengembangan sistem.

2.3. White Box Testing

White box testing dikenal sebagai pengujian struktur. *White-box testing* merupakan teknik pengujian perangkat lunak di mana penguji memiliki akses ke struktur internal program, termasuk kode, struktur data, dan algoritma [14]. Pendekatan ini memungkinkan perancangan kasus uji yang lebih komprehensif dibandingkan *black box testing*, karena penguji dapat menganalisis cara kerja internal sistem secara langsung.

Penelitian [15] menjelaskan bahwa pengujian *whitebox* memungkinkan penguji untuk mengidentifikasi kesalahan-kesalahan yang mungkin sulit ditemukan pada pengujian *black box*, seperti kesalahan logika, kesalahan dalam struktur kontrol, dan ketidakefisienan dalam penggunaan sumber daya.

Salah satu teknik *white-box testing* yang umum digunakan adalah basis path. Basis path testing merupakan salah satu teknik utama dalam pengujian *white box* yang digunakan untuk mengidentifikasi dan menguji jalur independen dalam program [16]. Jalur independen didefinisikan sebagai rangkaian eksekusi unik yang tidak memiliki simpul yang berulang. Dalam implementasinya pada website SolusiLaundry, setiap alur proses bisnis mulai dari registrasi pengguna, pemilihan layanan, proses pembayaran, hingga pelacakan status pesanan merepresentasikan jalur yang perlu diuji secara independen untuk memastikan keandalan sistem secara menyeluruh.

2.4. Basis Path

Teknik basis path testing memiliki tahapan dalam proses implementasinya. Pertama, membuat grafik *flowchart*. *Flowchart* adalah representasi visual dari struktur program yang menunjukkan semua jalur yang mungkin diambil oleh program. *Flowchart* ini kemudian digunakan untuk mengidentifikasi semua jalur independent.

Langkah selanjutnya adalah mengubah *flowchart* menjadi *flowgraph notion*. Selanjutnya adalah menghitung nilai kompleksitas siklomatik. Nilai kompleksitas siklomatik adalah metrik yang mengukur kompleksitas struktur program. Nilai yang lebih tinggi menunjukkan program yang lebih kompleks dan membutuhkan lebih banyak jalur uji.

Langkah selanjutnya adalah mengidentifikasi jalur independent. Setelah mendapatkan jalur independent, langkah selanjutnya adalah merancang kasus uji untuk setiap jalur independent. Kasus uji harus memastikan bahwa semua kondisi dalam program dievaluasi dan semua jalur dalam program dieksekusi. Selanjutnya adalah menjalankan kasus uji dan mencatat hasilnya. Terakhir adalah melakukan analisis hasil. Analisis hasil pengujian untuk mengidentifikasi cacat atau bug pada program untuk menentukan apakah memerlukan evaluasi atau perbaikan terhadap fitur yang telah diuji.

Beberapa penelitian terbaru telah menunjukkan efektivitas basis path testing dalam menguji berbagai jenis sistem informasi. Penelitian [17] menggunakan basis path testing untuk menguji sistem pakar. Penelitian tersebut menemukan bahwa basis path testing dapat mengidentifikasi lebih banyak cacat daripada teknik pengujian lainnya, seperti *black-box testing*. Penelitian [18] menggunakan basis path testing untuk menguji sistem web untuk e-commerce. Mereka menemukan bahwa basis path testing dapat mengurangi waktu yang dibutuhkan untuk menguji sistem.

Meskipun penelitian terbaru menunjukkan keefektifan basis path testing dalam menguji berbagai jenis sistem informasi, namun terdapat beberapa kelemahan Penelitian Basis Path sebelumnya. Kelemahan pertama ada pada keterbatasan cakupan, dimana beberapa penelitian sebelumnya hanya fokus pada pengujian fitur-

fitur tertentu dalam sistem. Selanjutnya analisis hasil yang tidak menyeluruh. Beberapa penelitian sebelumnya tidak menganalisis hasil pengujian secara menyeluruh untuk mengetahui kualitas dan kompleksitas sistem. Kurangnya fokus pada kualitas dan kompleksitas. Beberapa penelitian sebelumnya tidak secara eksplisit fokus pada kualitas dan kompleksitas sistem, yang merupakan indikator penting dari keandalan dan kemudahan penggunaan sistem.

Penelitian ini bertujuan untuk mengatasi kelemahan-kelemahan tersebut dengan menguji keseluruhan fitur dalam sistem manajemen laundry SolusiLaundry menggunakan teknik basis path, menganalisis hasil pengujian secara menyeluruh untuk mengetahui kualitas dan kompleksitas sistem, mengidentifikasi bagaimana cacat yang ditemukan memengaruhi kualitas sistem secara keseluruhan.

Penelitian ini diharapkan dapat memberikan kontribusi yang positif dalam pengembangan sistem manajemen laundry SolusiLaundry dan meningkatkan kualitas layanan yang diberikan kepada para penggunanya. Selain itu, dapat memberikan referensi bagi penelitian selanjutnya yang ingin menggunakan teknik basis path untuk menguji sistem informasi lainnya.

2.5. Penelitian Terkait

Pada penelitian sebelumnya [4], [6], [7] dilakukan pengujian dengan teknik basis path. Dalam penelitian tersebut, langkah pertama pengujian adalah membuat flowchart dan flowgraph. Kemudian dilanjutkan dengan menghitung nilai cyclomatic, yang digunakan sebagai indikator jumlah jalur independen yang harus diuji tiap skenarionya.

Dalam [8], [9], peneliti melakukan pengujian menggunakan teknik basis path pada aplikasi. Berdasarkan nilai cyclomatic complexity, terdapat 6 jalur independent yang harus diuji. Pengujian tersebut menghasilkan hasil yang sesuai dengan test case yang diharapkan. Namun demikian, pengujian basis path tersebut hanya dilakukan pada form login yang sederhana sehingga belum bisa mempresentasikan keberhasilan keseluruhan sistem.

Merujuk pada penelitian yang telah disebutkan, dapat diambil celah bahwa pengujian hanya fokus pada pengujian fitur-fitur tertentu dalam sistem. Hal ini

berpotensi mempengaruhi presentasi keakuratan dan keberhasilan keseluruhan kinerja sistem.

Penelitian sebelumnya juga menunjukkan kekurangan dalam menganalisis hasil pengujian. Beberapa penelitian sebelumnya tidak melakukan analisis secara menyeluruh untuk mengetahui kualitas dan kompleksitas sistem [9]. Hal ini dapat menyebabkan kurangnya pemahaman tentang efektivitas pengujian dan potensi masalah yang masih ada dalam sistem.

Penelitian ini bertujuan untuk menguji keseluruhan fitur dalam sistem SolusiLaundry menggunakan pengujian basis path. Dengan demikian, penelitian ini diharapkan mampu memberikan kontribusi pada pemahaman kinerja dari sistem manajemen SolusiLaundry.

