

PENGUJIAN PERANGKAT LUNAK PADA WEBSITE KA'CAKE: IMPLEMENTASI UNIT TESTING, INTEGRATION TESTING, SYSTEM TESTING, DAN VALIDATION TESTING UNTUK MENJAMIN KUALITAS DAN KEANDALAN SISTEM

Muhammad Amin Rahim Hidayat, Nadia Nura Izzati, Andika Rizky Putra Prirhatama,
Yudhistira Patria Wijaya, Susanti Kurmilasari

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Palangka Raya
Kampus Tunjung Nyaho Jl. Yos Sudarso Palangka Raya 73112, Indonesia
muhammadamin6909@mhs.eng.upr.ac.id

ABSTRAK

Penjaminan mutu perangkat lunak merupakan tahapan penting dalam pengembangan aplikasi untuk memastikan sistem berfungsi dengan baik, aman, dan sesuai kebutuhan pengguna. Penelitian ini membahas penerapan pengujian perangkat lunak pada website restoran Ka'Cake dengan metode White Box dan Black Box Testing. Pengujian dilakukan pada berbagai tahap, yaitu Unit Testing, Integration Testing, System Testing, dan Validation Testing. Hasil pengujian menunjukkan bahwa sistem mampu memvalidasi input dan menyimpan data dengan baik. Tingkat keberhasilan pengujian mencapai 100% untuk semua skenario pada Unit Testing, Integration Testing, System Testing, dan Validation Testing. Meskipun demikian, masih ditemukan beberapa perbedaan respons sistem terhadap input tidak valid pada tahap System Testing. Artikel ini juga menyajikan perbandingan dan hubungan antar jenis pengujian yang digunakan sebagai dasar evaluasi mutu perangkat lunak secara menyeluruh.

Kata kunci : *Software Quality Assurance, Ka'Cake, White Box Testing, Black Box Testing, Unit Testing*

1. PENDAHULUAN

Pengujian perangkat lunak (software testing) merupakan tahapan penting dalam siklus pengembangan perangkat lunak untuk menjamin bahwa sistem yang dibangun berfungsi sebagaimana mestinya, bebas dari kesalahan, dan memenuhi kebutuhan pengguna akhir. Dalam proses pengembangan aplikasi, pengujian bertujuan untuk mengidentifikasi bug atau error sejak tahap awal agar tidak berdampak pada fungsionalitas sistem saat dirilis. Pengujian perangkat lunak memiliki peranan penting dalam suatu sistem informasi, dengan pengujian ini dapat diketahui galat atau error pada sistem.[1]

Website Ka'Cake merupakan platform digital yang menyediakan layanan reservasi restoran, informasi menu, dan interaksi pengguna melalui chatbot. Namun, sebelum dilakukan pengujian, ditemukan beberapa permasalahan seperti validasi input yang tidak konsisten, pesan kesalahan yang kurang informatif, serta potensi kegagalan penyimpanan data yang tidak valid ke dalam database. Permasalahan-permasalahan ini berdampak pada pengalaman pengguna serta keandalan sistem, terutama dalam proses reservasi yang menjadi fitur utama.

Dalam penelitian ini, dilakukan proses pengujian secara menyeluruh pada aplikasi Ka'Cake untuk memastikan sistem dapat berjalan sesuai spesifikasi yang telah dirancang. Pengujian perangkat lunak tidak hanya bertujuan untuk menemukan bug, tetapi juga untuk meningkatkan kualitas perangkat lunak secara menyeluruh. Proses ini mencakup pengujian fungsionalitas, kinerja, keandalan, serta aspek usability yang sangat penting dalam konteks aplikasi

berbasis web. Selain itu, dalam dunia industri perangkat lunak modern, pengujian secara berjenjang atau berlapis telah menjadi praktik terbaik untuk menjamin stabilitas dan performa aplikasi.

2. TINJAUAN PUSTAKA

2.1. *Software Quality Assurance I (SQA)*

Pengujian perangkat lunak merupakan tahapan penting dalam proses pengembangan *software* yang ditujukan untuk menjamin mutu, performa, dan keamanan produk *software* yang sedang dikembangkan. Pengujian perangkat lunak memiliki cakupan yang lebih luas dari sekadar mencari bug, yaitu bertujuan untuk meningkatkan kualitas *software* secara menyeluruh.

Berbagai keuntungan besar dari penerapan SQA termasuk kualitas produk akhir yang lebih baik, kepuasan pengguna yang lebih tinggi, dan penurunan kemungkinan kegagalan aplikasi setelah peluncuran.[2]

2.2. *Black Box Testing*

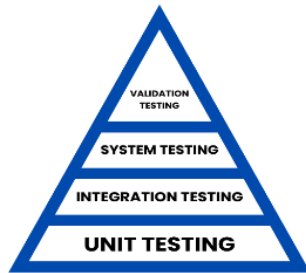
Metode Black Box Testing adalah teknik pengujian yang ditujukan untuk mengidentifikasi kesalahan dalam sistem aplikasi, seperti kekeliruan pada fungsi sistem dan hilangnya menu aplikasi. [3]

2.3. *White Box Testing*

White Box adalah metode pengujian perangkat lunak yang dilakukan dengan menganalisis serta mengevaluasi struktur internal dan kode programnya. White Box Testing menguji suatu aplikasi atau perangkat lunak dengan langsung menganalisis kode programnya tanpa memperhatikan tampilan atau User Interface (UI). [4]

2.4. Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan proses yang dilakukan untuk mengevaluasi fungsionalitas dari suatu aplikasi atau sistem dengan tujuan menemukan kesalahan dan memastikan perangkat lunak berjalan dengan spesifikasi yang telah ditentukan. Ada pun tujuan pengujian perangkat lunak itu sendiri yaitu untuk menemukan kesalahan yang menyebabkan perangkat lunak yang telah dibangun gagal.



Gambar 1. Sistem pengujian mutu perangkat lunak

2.5. Unit Testing

Unit testing merupakan pengujian pada tingkat unit atau modul terkecil dari perangkat lunak untuk memastikan bahwa setiap bagian berfungsi sebagaimana mestinya. elah dikembangkan sudah sesuai yang diharapkan atau tidak. Biasanya, pengujian ini dilakukan secara otomatis menggunakan framework seperti PHP Unit, JUnit, PyTest, Mocha dan lain-lain.

Unit testing ini sangatlah efektif dalam pengembangan, karena akan mempermudah tim dalam memperbaiki desain kode program dan mengurangi waktu saat melakukan debugging.[6]

2.6. Integration Testing

Pengujian integrasi bertujuan untuk menguji bagaimana modul atau komponen dalam sistem dapat berinteraksi dengan baik satu sama lain. Pengujian ini penting untuk mendeteksi kesalahan yang terjadi akibat integrasi antar modul yang sebelumnya diuji secara individual dalam unit testing.

Perbedaan antara unit test dan integration test terletak pada subjek yang diuji, dimana unit test biasanya berfokus pada satu bagian tertentu di tiap tesnya, sedangkan integration test biasanya menguji beberapa komponen sekaligus dan memastikan tiap komponen tersebut berfungsi dengan baik.[7]

2.7. System Testing

Pengujian sistem merupakan pengujian secara menyeluruh terhadap perangkat lunak guna memastikan bahwa semua fitur telah berfungsi sesuai dengan spesifikasi yang telah ditetapkan.[8] System testing mencakup :

- a. Pengujian fungsional: Memastikan fitur bekerja sesuai dengan kebutuhan.

- b. Pengujian non-fungsional: Termasuk performa, keamanan, dan kompatibilitas.

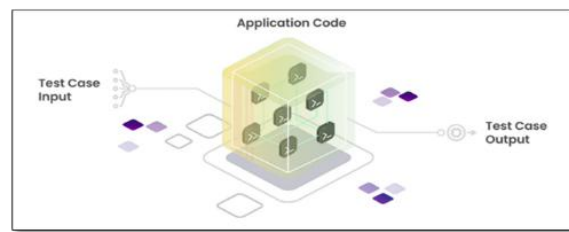
2.8. Validation Testing

Validation testing dilakukan untuk memastikan apakah perangkat lunak memenuhi kebutuhan pengguna dan dapat digunakan dengan baik di lingkungan sebenarnya. Pengujian ini biasanya dilakukan oleh pengguna akhir dengan skenario dunia nyata sebelum perangkat lunak dirilis secara resmi.

2.9. White Box Testing

White Box Testing, atau pengujian white box, merupakan metode pengujian yang dilakukan dengan menganalisis kode program untuk mendeteksi adanya kesalahan.[9]

Dalam pengujian ini, penguji harus memiliki pemahaman tentang kode yang diuji, sehingga dapat mengidentifikasi jalur eksekusi, cabang logika, dan kemungkinan kesalahan seperti apakah masukan dan keluaran yang sesuai dengan spesifikasi yang dibutuhkan.[10]



Gambar 2. Skema white box testing

2.10. Black Box Testing

Black-box Testing adalah metode pengujian perangkat lunak yang menilai fungsionalitas sistem tanpa memperhatikan struktur internal atau kode sumbernya. Teknik perancangan data untuk pengujian yang dimana spesifikasi dalam perangkat lunak menjadi indikator utama dalam pengujian.[11]



Gambar 3. Skema black box testing

3. METODE PENELITIAN

3.1. Pemilihan Objek Penelitian

Website Ka'Cake dipilih karena memiliki struktur modular dan fitur beragam seperti halaman home, menu, reservasi meja, tim, chatbot, dan dashboard admin. Kompleksitas fitur tersebut memungkinkan pengujian dilakukan secara bertingkat, mulai dari unit testing hingga pengujian sistem secara keseluruhan, dan memberikan kesempatan untuk

fokus pada validasi input, interaksi antar modul, serta penyimpanan data di backend.

3.2. Pendekatan Pengujian

Untuk menjamin mutu perangkat lunak, digunakan dua pendekatan utama. White Box Testing diterapkan pada Unit dan Integration Testing untuk mengulas logika dan struktur internal kode, sedangkan Black Box Testing digunakan pada System dan Validation Testing untuk menilai output berdasarkan input tanpa meneliti kode sumber.

3.3. Identifikasi Pengguna

- Bahasa Pemrograman: PHP 8.2.4
- Framework Unit Test: PHPUnit 11.5.10
- Server: XAMPP (Apache, MySQL)
- Tools tambahan: GuzzleHttp untuk pengujian API endpoint
- Sistem Operasi: Windows 10/11
- Browser: Google Chrome versi 120, Microsoft Edge versi 133.0.3065.82

Lingkungan pengujian disesuaikan dengan lingkungan yang kemungkinan besar digunakan dalam produksi untuk memastikan hasil pengujian seakurat mungkin.

3.4. Alur Tahapan Penelitian

Alur penelitian dilakukan secara bertahap dari awal hingga hasil akhir, dengan poin-poin pembahasan sebagai berikut:

- Studi literatur dan Identifikasi Masalah
Mengkaji penelitian-penelitian terdahulu mengenai Software Quality Assurance (SQA), White Box, dan Black Box Testing. Mengidentifikasi permasalahan di lokasi studi kasus, seperti validasi input yang tidak konsisten dan pesan error yang kurang informatif.
- Pemilihan Objek dan Persiapan
Website Ka'Cake ditetapkan sebagai objek penelitian dan lingkungan pengujian disusun sesuai spesifikasi produksi guna memastikan evaluasi kinerja sistem yang valid dan representatif.
- Perancangan Kasus Uji
Pengujian dilakukan secara bertahap. Unit Testing memvalidasi input seperti nama, email, jumlah, dan waktu reservasi. Integration Testing menguji interaksi antar modul, misalnya, alur data formulir ke database. System Testing mensimulasikan penggunaan nyata untuk mengevaluasi seluruh fungsi sistem, sedangkan Validation Testing melalui UAT menilai kemudahan penggunaan dan respons sistem.
- Pelaksanaan Pengujian dan Pengumpulan Data
Setiap tahapan pengujian dilaksanakan secara berturut-turut (Unit → Integration → System → Validation), dan hasilnya dicatat melalui log otomatis, tabel, yang membandingkan nilai aktual dengan ekspektasi.

e. Analisis Hasil

Data pengujian dianalisis untuk menghitung persentase keberhasilan tiap skenario dan mengidentifikasi titik-titik error, kemudian laporan akhir disusun untuk merangkum hasil pengujian.

4. HASIL DAN PEMBAHASAN

Pengujian perangkat lunak dilakukan secara bertahap untuk memastikan bahwa setiap komponen dalam aplikasi Ka'Cake berfungsi sesuai dengan spesifikasi yang ditetapkan. Pengujian ini mencakup unit testing, pengujian integrasi, pengujian sistem, serta uji penerimaan pengguna (*User Acceptance Testing* - UAT).

4.1. Studi Literatur dan Identifikasi Masalah

Pada tahap awal, dilakukan kajian terhadap penelitian-penelitian terdahulu yang berkaitan dengan *Software Quality Assurance* (SQA), *White Box Testing*, dan *Black Box Testing*. Hasil kajian menunjukkan pentingnya pengujian berlapis untuk menjamin kualitas perangkat lunak. Berdasarkan analisis awal pada website Ka'Cake, ditemukan beberapa permasalahan seperti:

- Validasi input yang belum konsisten (contohnya: nama kosong tetap bisa dikirim),
- Pesan error yang kurang informatif atau tidak muncul sama sekali,
- Potensi penyimpanan data tidak valid ke database.

Masalah-masalah ini menjadi dasar utama untuk dilakukan pengujian berjenjang demi memastikan kualitas dan keandalan sistem.

4.2. Pemilihan Objek dan Persiapan

Website Ka'Cake dipilih sebagai objek penelitian karena kompleksitas fiturnya yang mencakup formulir reservasi, halaman menu, chatbot, dan dashboard admin. Lingkungan pengujian disusun menyerupai kondisi produksi:

- PHP 8.2.4,
- PHPUnit 11.5.10,
- XAMPP, dan
- Windows 10/11.
- Pengujian juga dilakukan dengan tambahan tools seperti GuzzleHttp untuk simulasi request API.

4.3. Perancangan Kasus Uji

Kasus uji dirancang berdasarkan metode:

- White Box Testing* pada Unit dan *Integration Testing*, untuk menganalisis struktur dan logika kode,
- Black Box Testing* pada System dan *Validation Testing*, untuk menilai fungsi sistem dari sisi pengguna.

Unit Testing fokus pada validasi input dalam formulir reservasi, seperti:

- Nama tidak boleh kosong,
- Format email harus valid,
- Jumlah orang tidak boleh nol,
- Tanggal dan waktu reservasi valid.

Integration Testing menguji aliran data antar modul, khususnya interaksi antara form dan database. *System Testing* mensimulasikan penggunaan penuh sistem, termasuk pengiriman data melalui API dan validasi respons sistem terhadap input yang valid maupun tidak valid.

Validation Testing dilakukan oleh pengguna akhir melalui UAT (*User Acceptance Testing*), menilai pengalaman pengguna terhadap kemudahan penggunaan dan kejelasan *feedback* sistem.

4.4. Unit Testing

Unit testing dilakukan untuk memastikan bahwa setiap komponen dalam aplikasi Ka'Cake berfungsi sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini difokuskan pada Formulir Reservasi (Reservation Form) guna *memverifikasi* validasi input dan respons sistem terhadap berbagai skenario yang diuji.

White Box Testing digunakan dalam *Unit Testing* aplikasi Ka'Cake untuk memastikan validasi input pada Formulir Reservasi berjalan sesuai spesifikasi. Pengujian ini dilakukan dengan PHPUnit 11.5.10 pada lingkungan Windows 10/11, PHP 8.2.4, dan server XAMPP. Fokus utama *adalah* memastikan aturan seperti nama tidak boleh kosong, format email valid, jumlah orang tidak negatif, serta waktu reservasi sesuai ketentuan. Dengan lima skenario pengujian yang berhasil 100% lulus, Unit Testing membuktikan bahwa sistem mampu menangani berbagai input dengan benar dan sesuai ekspektasi. Dapat dilihat pada

Tabel 1 merupakan informasi umum mengenai *unit testing* yang dilakukan.

Tabel 1. Informasi umum

Nama	Ka'Cake
Modul yang Diuji	Formulir Reservasi (Reservation Form)
Tanggal Pengujian	27 Februari 2025
Alat yang Digunakan	PHPUnit 11.5.10
Lingkungan Pengujian	Sistem Operasi: Windows 10/11
PHP Version	8.2.4
Server	XAMPP

Pada Tabel 2. merupakan hasil pengujian menunjukkan bahwa semua skenario validasi telah berhasil dijalankan dengan baik, memastikan bahwa form reservasi dapat mengelola input pengguna secara akurat dan mencegah kesalahan dalam proses reservasi

Tabel 2. Unit testing tujuan pengujian

Skenario Validasi	Hasil
Validasi nama tidak kosong	OK
Validasi format email	OK
Validasi jumlah orang tidak negatif atau kosong	OK
Validasi tanggal dan waktu reservasi	OK

Skenario pengujian dilakukan untuk mengevaluasi apakah formulir reservasi dalam aplikasi Ka'Cake mampu memproses dan memvalidasi input dengan benar. Setiap skenario diuji menggunakan berbagai jenis input untuk memastikan sistem memberikan respons yang sesuai. Berikut Tabel 3. adalah hasil pengujian berdasarkan skenario yang telah diuji:

Tabel 3. Unit testing skenario dan hasil pengujian

Skenario	Input Uji	Ekspektasi	Hasil Aktual	Status
Nama tidak boleh kosong	"" (kosong)	Error: "Nama wajib diisi"	Error ditampilkan	Lulus
Format email valid	john@example.com,	Diterima,	Diterima,	Lulus
Format email salah	"johnexample.com"	Error: "Email tidak valid"	Error ditampilkan	Lulus
Jumlah orang minimal	"0"	Error: "Jumlah orang minimal 1"	Error ditampilkan	Lulus
Waktu reservasi harus valid	"2025-03-01 19:00"	Diterima	Diterima	Lulus

4.5. Integration testing

Pengujian integrasi dilakukan untuk memastikan bahwa modul dalam aplikasi Ka'Cake dapat bekerja secara harmonis dan berkomunikasi dengan baik satu sama lain. Pengujian ini difokuskan pada Formulir Reservasi (Reservation Form) guna memastikan bahwa data yang dimasukkan oleh pengguna tidak hanya diverifikasi dengan benar tetapi juga berhasil dikirim dan disimpan di database.

Pengujian integrasi pada Formulir Reservasi aplikasi Ka'Cake menggunakan *White Box Testing* karena menguji struktur internal kode dan interaksi

modul dengan database MySQL. Dengan PHPUnit 11.5.10, pengujian memastikan validasi input, penyimpanan data, serta penanganan error sesuai aturan kode. Proses ini menilai aliran data dan logika program, bukan sekadar output tampilan. Hasil pengujian menunjukkan bahwa sistem bekerja sesuai harapan, menjadikan metode *White Box Testing* tepat dalam memastikan integrasi dan validasi sistem berjalan dengan baik. Tabel 4. Berikut merupakan skenario dan hasil pengujian integrasi yang telah dilakukan:

Tabel 4. *Intergrasion Testing*

Skenario	Input Uji	Ekspektasi	Hasil Aktual	Status
Pengisian form valid & pengiriman ke database	Nama: Jane Doe, Email: jane@example.com, Waktu: 2025-03-01 19:00, Jumlah: 2 Request: Near window	Data tersimpan di database	Data ditemukan dalam database	✓ Lulus
Input email tidak valid	Nama: John Doe, Email: invalid-email, Waktu: 2025-03-01 18:00, Jumlah: 2	Sistem menolak input dengan error "Email tidak valid"	Error ditampilkan	✓ Lulus
Input tanpa waktu	Nama: John Doe, Email: john@example.com, Waktu: (kosong), Jumlah: 2	Sistem menolak input dengan error "Waktu harus diisi"	Error ditampilkan	✓ Lulus
Input tanpa nama	Nama: (kosong), Email: john@example.com, Waktu: 2025-03-01 18:00, Jumlah: 2	Sistem menolak input dengan error "Nama harus diisi"	Error ditampilkan	✓ Lulus

4.6. System testing

System Testing dilakukan untuk mengevaluasi keandalan dan fungsionalitas Sistem Reservasi Bakery secara menyeluruh. Pengujian ini menggunakan metode System Testing dengan PHPUnit dan GuzzleHttp untuk memastikan sistem dapat menerima input pengguna, memvalidasi data, serta menyimpan informasi reservasi ke dalam database dengan benar. Lingkungan pengujian mencakup PHP 8.2.4, PHPUnit 11.5.10, server XAMPP, serta database MySQL (bakery3). Beberapa skenario diuji, termasuk pengiriman data reservasi yang valid, validasi email, serta pengujian terhadap input yang tidak lengkap. Hasil pengujian menunjukkan bahwa sistem telah berfungsi sesuai dengan harapan, dengan semua skenario yang diuji berhasil memberikan respons yang tepat. Berdasarkan evaluasi ini, sistem dinyatakan lulus uji, dengan rekomendasi untuk menambahkan pengujian terhadap serangan keamanan seperti SQL Injection dan XSS, serta pengujian performa di bawah beban tinggi.

Pengujian sistem reservasi menggunakan metode *Black Box Testing* dilakukan untuk memastikan bahwa sistem dapat menangani berbagai skenario input dengan benar tanpa melihat struktur internal kode. Dengan menggunakan PHPUnit dan GuzzleHttp, pengujian mencakup skenario seperti pengiriman data reservasi yang valid, input dengan email tidak valid,

data tanpa waktu reservasi, serta data tanpa nama pelanggan. Hasil pengujian menunjukkan bahwa sistem berhasil memvalidasi dan merespons setiap skenario sesuai harapan, menolak input yang tidak lengkap, serta menyimpan data yang valid ke dalam database. Dengan semua uji coba lulus tanpa error, sistem dianggap memenuhi fungsionalitas yang diharapkan, namun masih direkomendasikan untuk menambahkan pengujian edge case seperti SQL Injection dan XSS serta menguji performa sistem di bawah beban tinggi.

Pada pengujian dilakukan dengan beberapa skenario untuk mengevaluasi kemampuan sistem dalam menangani berbagai kondisi input. Pengujian mencakup pengiriman data reservasi yang valid serta beberapa skenario dengan data tidak lengkap atau tidak sesuai. Hasil pengujian menunjukkan bahwa sistem mampu menyimpan data reservasi dengan benar ketika input valid, serta memberikan respons error yang sesuai saat terjadi kesalahan input, seperti format email yang tidak valid, waktu reservasi yang kosong, atau nama pelanggan yang tidak diisi. Semua skenario yang diuji berhasil memberikan hasil yang diharapkan, menandakan bahwa sistem memiliki mekanisme validasi data yang baik. Berikut ini merupakan skenario pengujian dan hasil yang sudah dilakukan:

Tabel 5. Skenario pengujian dan hasil

Skenario Pengujian	Data Uji	Ekspektasi	Hasil
Mengirim data reservasi yang valid	Nama: Jane Doe, Email: jane@gmail.com , Waktu: 03-15-2025 09:00 PM, Jumlah Orang: 2 Persons, Request: Near window	Data tersimpan di database, API merespon dengan success: true	✓ LULUS
Mengirim data dengan email tidak valid	Nama: Jane Doe, Email: invalid-email, Waktu: 03-15-2025 09:00 PM, Jumlah Orang: 2 Persons, Request: Near window	API merespon dengan success: false dan message: Email tidak valid	✓ LULUS
Mengirim data tanpa waktu reservasi	Nama: Jane Doe, Email: jane@gmail.com , Waktu: (kosong), Jumlah Orang: 2 Persons, Request: Near window	API merespon dengan success: false dan message: Data tidak lengkap	✓ LULUS
Mengirim data tanpa nama pelanggan	Nama: (kosong), Email: jane@example.com , Waktu: 03-15-2025 09:00 PM, Jumlah Orang: 2 Persons, Request: Near window	API merespon dengan success: false dan message: Data tidak lengkap	✓ LULUS

4.7. Validation testing

Pengujian yang dilakukan ini termasuk dalam kategori *Black Box Testing*, khususnya User Acceptance Testing (UAT), yang bertujuan untuk memastikan bahwa formulir reservasi dalam aplikasi Ka'Cake berfungsi sesuai dengan kebutuhan pengguna tanpa melihat atau menganalisis kode sumbernya. Fokus utama dari pengujian ini adalah memvalidasi input pengguna, memastikan formulir reservasi dapat diakses dan digunakan dengan benar, mengevaluasi apakah sistem memberikan notifikasi dan pesan error yang sesuai, serta menilai pengalaman pengguna dalam mengisi formulir. Berbagai skenario diuji dengan memberikan input tertentu dan mencocokkan hasil aktual dengan ekspektasi yang telah ditentukan,

sehingga metode ini hanya mengevaluasi output berdasarkan input tanpa mempertimbangkan bagaimana sistem bekerja secara internal. Pengujian ini dilakukan dari sudut pandang pengguna akhir untuk memastikan bahwa fitur formulir reservasi bekerja dengan baik sebelum implementasi penuh dilakukan. Melalui pendekatan *Black Box Testing*, pengujian ini membantu mengidentifikasi potensi perbaikan dalam sistem tanpa memerlukan pemahaman teknis tentang kode sumber aplikasi, sehingga lebih berfokus pada pengalaman pengguna dan kesesuaian dengan kebutuhan bisnis.

Pengujian dilakukan dengan berbagai skenario dan input untuk memastikan sistem memberikan respons yang sesuai. Berikut adalah hasil pengujian:

Tabel 6. Skenario dan Hasil Pengujian

Skenario	Langkah Uji	Hasil yang diharapkan	Hasil Aktual	Status
Melakukan reservasi meja	Isi formulir dengan data valid dan klik "Book Now"	Pesan sukses muncul dan reservasi tersimpan	Pesan sukses ditampilkan dan data ditemukan dalam database	✓ Lulus
Cek validasi input	Coba kirim form dengan data kosong atau email salah	Muncul peringatan kesalahan	Error ditampilkan	✓ Lulus
Konfirmasi reservasi	Setelah submit, cek tampilan halaman	Pengguna mendapat pesan konfirmasi	Pesan konfirmasi ditampilkan	✓ Lulus
Keamanan data	Periksa apakah data pribadi bisa diakses oleh pihak lain	Data tetap aman	Data aman hanya bisa diakses admin restoran Tampilan rapi dan navigasi mudah di perangkat PC maupun mobile	✓ Lulus
Responsivitas	Uji sistem di berbagai perangkat (PC dan mobile)	Tampilan tetap rapi dan navigasi mudah.	Tampilan	✓ Lulus

4.8. Analisis Hasil

Dari data yang dikumpulkan, berikut ringkasan hasil analisis:

- Unit Testing*: semua validasi berjalan sesuai ekspektasi (100% lulus),
- Integration Testing*: komunikasi form–database berjalan sempurna (100% lulus),
- System Testing*: sistem berfungsi dengan baik namun masih ada potensi perbaikan validasi pada *edge case*,
- Validation Testing*: pengguna menyatakan sistem responsif dan mudah digunakan, meskipun ada saran untuk meningkatkan kejelasan pesan kesalahan.

Keseluruhan pengujian menunjukkan bahwa sistem telah layak digunakan, namun dengan catatan agar dilakukan pengujian tambahan seperti uji keamanan (*SQL Injection/XSS*) dan uji performa di bawah beban tinggi untuk lebih menjamin stabilitas.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian, aplikasi Ka'Cake memiliki kualitas yang baik dalam aspek validasi input, integrasi sistem, dan fungsionalitas utama. Unit Testing menunjukkan bahwa formulir reservasi dapat

menangani berbagai skenario input dengan benar, memastikan data sesuai dengan ketentuan yang ditetapkan. Integration Testing membuktikan bahwa komunikasi antara formulir dan *database* berjalan dengan baik, dengan data valid tersimpan dan input tidak valid ditolak sesuai aturan. System Testing mengonfirmasi bahwa sistem dapat beroperasi sesuai fungsinya, meskipun masih diperlukan perbaikan dalam menangani input tidak valid untuk meningkatkan ketahanan sistem. Validation Testing melalui UAT juga menunjukkan bahwa formulir reservasi telah memenuhi standar penggunaan dan validasi input yang dibutuhkan pengguna. Dengan hasil ini, aplikasi Ka'Cake dinilai telah memenuhi kebutuhan pengguna dalam melakukan reservasi, meskipun masih dapat ditingkatkan dengan perbaikan pada validasi error untuk pengalaman yang lebih optimal.

DAFTAR PUSTAKA

- [1] D. Ahrizal, M. K. Miftah, R. Kurniawan, and T. Zaelani, "Pengujian perangkat lunak sistem informasi peminjaman PlayStation dengan teknik Boundary Value Analysis menggunakan metode Black Box Testing," *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 1, pp. 73-77, Mar. 2020. [Online]. Available:

- <http://openjournal.unpam.ac.id/index.php/informatika>
- [2] B. Respiar, A. Fernanda, Taryadi, F. Maulana, and A. H. Anshor, "Peran penting Software Quality Assurance dalam pengembangan aplikasi," *Journal Scientific of Mandalika*, vol. 5, no. 12, pp. 535-540, Dec. 2024. [Online]. Available: <http://ojs.cahayamandalika.com/index.php/jomla>
- [3] N. M. D. Febriyanti, A. A. K. O. Sudana, and I. N. Piarsa, "Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen," *JITTER - Jurnal Ilmiah Teknologi dan Komputer*, vol. 2, no. 3, pp. 1-10, Dec. 2021.
- [4] I. M. A. O. Gunawan, G. Indrawan, and Sariyasa, "Pengembangan Sistem Informasi Kemajuan Akademik Menggunakan Model Incremental Berbasis Evaluasi Usability dan White Box Testing," *SINTECH Journal*, vol. 4, no. 1, Apr. 2021. [Online]. Available: <https://s.id/sintechjournal>.
- [5] A. N. Hasibuan and T. Dirgahayu, "Pengujian dengan Unit Testing dan Test Case pada Proyek Pengembangan Modul Manajemen Pengguna," *Proceedings of Informatics Research Conference*, Universitas Islam Indonesia, Yogyakarta, 2024.
- [6] R. Setiawan and R. Risal, "Pengembangan Unit Testing dan Integration Testing REST API Pengelola Data Bootcamp PT Mitra Integrasi Informatika," *Jurnal Strategi*, vol. 6, no. 2, Nov. 2024.
- [7] A. Murtado, E. Firmansyah, M. R. Fadillah, P. Maula, and A. Saifudin, "Survei Strategi Pengujian Software Menggunakan Metode Systematic Literature Review," *Jurnal BATIRSI*, vol. 8, no. 1, Jul. 2024. [Online]. Available: <https://jurnalbatirsi.com>
- [8] H. Gusdevi, B. D. R. Utama, and A. A. Rizaldi, "Pengujian White Box pada Aplikasi Pinjam Dulu Seratus Berbasis Website," *NARATIF: Jurnal Ilmiah Nasional Riset Aplikasi dan Teknik Informatika*, vol. 6, no. 2, Dec. 2024.
- [9] C. T. Pratala, E. M. Asyer, I. Prayudi, and A. Saifudin, "Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path," **Jurnal Informatika Universitas Pamulang**, vol. 5, no. 2, pp. 111-119, Jun. 2020, doi: 10.32493/informatika.v5i2.4713.
- [10] H. Hendra and R. P. Kristianto, "Pengujian Aplikasi Game Puzzle Indonesia Berbasis Android dengan Teknik Black-Box Testing," *INFOTECH: Journal of Technology Information*, vol. 10, no. 1, Jun. 2024, doi: 10.37365/jti.v10i1.224