

Pengujian Black-Box Dengan Teknik State Transition pada Sistem Appoinment Klinik Berbasis Web Citra Medica

David Sugiarto¹, Helmi Azkia², Egia Rosi Subhiyakto³

^{1,2,3}Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

Jl. Imam Bonjol No. 207, Pendrikan Kidul, Semarang Tengah, Kota Semarang, Jawa Tengah 50131

111202214745@mhs.dinus.ac.id . 111202214733@mhs.dinus.ac.id . egia@dsn.dinus.ac.id

ARTICLE INFO

Article history:

Received Agustus 2025

Received in revised from September 2025

Accepted November 2025

Available online Desember 2025

ABSTRACT

Software testing is an essential stage to ensure the quality and reliability of a system before it is widely deployed. One of the commonly used methods is black box testing, which focuses on evaluating the functionality of the software without considering the internal structure of the code. In this study, the black box testing method was applied using the state transition technique to test the web-based clinic appointment system "Citra Medica." The testing process involved mapping user state flows through a transition diagram and creating test scenarios for each state change. The test results showed that all system functions operated according to the specifications, with a 100% success rate achieved in every transition scenario. This method proved effective in detecting logical errors and ensuring system stability during user state transitions.

Keywords: appointment system, black box testing, software testing, state transition, website

1. Introduction

Perkembangan teknologi informasi yang pesat telah mendorong digitalisasi di berbagai sektor, termasuk bidang kesehatan. Klinik modern kini berlomba-lomba menerapkan sistem berbasis web untuk memudahkan pasien dalam melakukan pendaftaran, pemesanan jadwal konsultasi, serta pengelolaan rekam medis. Salah satu solusi digital yang semakin banyak digunakan adalah sistem appointment klinik berbasis web, yang memungkinkan pasien membuat janji temu dengan dokter secara daring. Sistem ini membantu meningkatkan efisiensi pelayanan dan mengurangi kesalahan administratif yang sering terjadi pada proses manual. Namun, seiring meningkatnya kompleksitas sistem berbasis web, diperlukan metode pengujian perangkat lunak yang tepat untuk menjamin kualitas dan fungsionalitas sistem. Software testing merupakan tahap penting dalam Software Development Life Cycle (SDLC) yang bertujuan untuk memastikan sistem bebas dari kesalahan logika dan sesuai dengan kebutuhan pengguna. Salah satu metode yang umum digunakan dalam pengujian fungsional adalah black box testing, di mana pengujian berfokus pada hasil keluaran dari masukan yang diberikan tanpa memperhatikan kode internal program [1]. Metode black box testing memiliki beberapa teknik, antara lain equivalence partitioning, boundary value analysis, decision table testing, dan state transition testing. Teknik terakhir ini memiliki keunggulan dalam menganalisis perilaku sistem yang berubah sesuai dengan status dan kondisi tertentu. State transition testing menguji transisi dari satu keadaan ke keadaan lain berdasarkan peristiwa yang terjadi, sehingga sangat cocok digunakan untuk sistem dengan banyak interaksi status pengguna seperti aplikasi berbasis web dan mobile [2]. Pendekatan serupa juga digunakan dalam penelitian Corazza et al. (2021) yang memanfaatkan teknik tree kernel untuk mendeteksi

near-duplicate states pada pengujian aplikasi web, sehingga meningkatkan efisiensi dalam proses inferensi model otomatis [3].

Teknik state transition telah terbukti efektif dalam berbagai penelitian. Mohialden et al. (2022) juga meninjau berbagai metode pengujian perangkat lunak, termasuk pengujian black-box, white-box, dan grey-box, serta mengklasifikasikan metode testing berdasarkan karakteristiknya [4], pengujian pada aplikasi konseling digital HIV/AIDS (D-Cohiva Apps) menunjukkan tingkat keberhasilan 100% pada seluruh skenario pengujian [5]. Begitu pula penelitian pada sistem inventori alat-alat medis di rumah sakit menghasilkan kesesuaian penuh antara spesifikasi dan fungsi sistem [6]. Hasil serupa juga diperoleh dalam pengujian aplikasi BMI Calculator berbasis Android yang menggunakan metode black box dengan teknik state transition untuk memvalidasi alur logika input dan output pengguna [7]. Selain di bidang kesehatan, pendekatan black box testing juga telah digunakan secara luas pada berbagai jenis sistem. Penerapan metode ini pada fitur TEA di aplikasi QM Mobile Astra Credit Companies membantu memastikan setiap fungsi berjalan sesuai kebutuhan pengguna tanpa perlu melihat kode sumber [8]. Dalam bidang bisnis, metode ini juga diterapkan pada sistem informasi inventaris penjualan berbasis web di Toko Reog Cell untuk menguji keakuratan data stok dan transaksi penjualan [9]. Sementara itu, penerapan konsep state transition juga dapat ditemukan pada berbagai sistem berbasis web dan multimedia. Salah satunya dalam konteks pembelajaran, pendekatan serupa digunakan dalam pengembangan media pembelajaran interaktif berbasis Adobe Animate yang memanfaatkan teknik pengujian untuk memastikan transisi antarhalaman berjalan sesuai skenario pengguna [10]. Saefullah et al. juga menerapkan metode black box testing pada aplikasi buku tamu berbasis web untuk menilai fungsionalitas utama seperti validasi data dan keandalan sistem dari sudut pandang pengguna [11]. Selain itu, Tetteh (2024) menjelaskan bahwa pengujian perangkat lunak mencakup berbagai tingkatan seperti unit, integrasi, sistem, dan penerimaan, yang semuanya berperan penting dalam memastikan keandalan dan keamanan aplikasi [12].

Dari berbagai hasil penelitian tersebut, dapat disimpulkan bahwa black box testing dengan teknik state transition memberikan keunggulan dalam mendeteksi kesalahan logika yang berkaitan dengan alur atau perpindahan status dalam sistem. Teknik ini tidak hanya memastikan fungsionalitas berjalan dengan benar, tetapi juga meningkatkan reliabilitas dan stabilitas sistem, terutama dalam sistem yang memiliki banyak kondisi pengguna. Berdasarkan latar belakang tersebut, penelitian ini berfokus pada penerapan metode black box testing dengan teknik state transition pada Sistem Appointment Klinik Berbasis Web “Citra Medica”. Tujuannya adalah untuk memastikan setiap transisi status pengguna, seperti proses login, pembuatan janji temu, pembatalan, hingga logout, berjalan sesuai spesifikasi yang diharapkan. Melalui pendekatan ini, diharapkan hasil penelitian dapat memberikan kontribusi dalam pengembangan metode pengujian perangkat lunak yang lebih sistematis, efisien, dan relevan untuk diterapkan pada sistem berbasis web di bidang layanan kesehatan.

2. Metode

Metode penelitian digunakan untuk menjelaskan pendekatan yang diterapkan dalam melakukan pengujian terhadap sistem yang dikembangkan. Pada penelitian ini, metode yang digunakan adalah pengujian perangkat lunak (software testing) dengan pendekatan black box testing yang difokuskan pada pengujian fungsi sistem dari sisi pengguna (user perspective). Pengujian dilakukan untuk memastikan bahwa seluruh fitur pada Sistem Appointment Klinik Berbasis Web “Citra Medica” berjalan sesuai dengan kebutuhan pengguna dan spesifikasi sistem. Teknik yang digunakan dalam pengujian ini adalah State Transition Testing, yaitu teknik yang memeriksa kebenaran alur transisi antarstatus dalam sistem berdasarkan input dan aksi pengguna.

Metode ini dipilih karena mampu memberikan gambaran yang jelas mengenai bagaimana sistem merespons setiap tindakan pengguna, sekaligus mendeteksi kemungkinan terjadinya kesalahan logika pada perpindahan antarstatus. Menurut Vitorino et al. (2025), pendekatan pengujian perangkat lunak berbasis otomatisasi seperti adversarial learning dapat memperkuat efektivitas pengujian black box dalam mendeteksi kesalahan dan kerentanan sistem secara lebih adaptif [13]. Berikut ini merupakan uraian dari dua tahapan utama pengujian yang dilakukan pada penelitian ini.

2.1. Pengujian Black Box

Metode black box testing merupakan metode pengujian perangkat lunak yang berfokus pada hasil keluaran (output) dari input yang diberikan tanpa memeriksa kode internal. Pengujian ini bertujuan memastikan sistem bekerja sesuai fungsionalitas yang diharapkan oleh pengguna. Pendekatan ini cocok digunakan pada sistem berbasis web karena pengujian dilakukan berdasarkan interaksi pengguna melalui antarmuka (user interface) tanpa harus mengetahui struktur logika program [14]. Tahapan pengujian black box dalam penelitian ini terdiri dari beberapa langkah sistematis, yaitu:

1. Analisis kebutuhan system
Mengidentifikasi kebutuhan fungsional berdasarkan modul sistem untuk setiap role pengguna.
2. Penentuan fungsi yang diuji
Fitur-fitur yang diuji meliputi:
 - a. Admin: CRUD data dokter, CRUD data pasien (manual register), CRUD data poli, dan CRUD data obat.
 - b. Dokter: CRUD jadwal pemeriksaan, melakukan pemeriksaan pasien, melihat riwayat pasien, dan mengelola profil.
 - c. Pasien: melakukan pendaftaran pemeriksaan (daftar pemeriksaan).
3. Penyusunan data uji (test data)
Menentukan data valid dan tidak valid untuk setiap form input, misalnya data dokter dengan ID duplikat, pasien tanpa poli, atau format data tidak sesuai.
4. Pelaksanaan pengujian (execution)
Menjalankan sistem dan melakukan input sesuai skenario uji, kemudian mencatat hasil keluaran yang diberikan sistem.
5. Perbandingan hasil aktual dan hasil yang diharapkan
Jika hasil aktual sesuai dengan spesifikasi kebutuhan, maka fungsi tersebut dinyatakan valid.
6. Evaluasi hasil pengujian
Hasil yang tidak sesuai akan dianalisis untuk menemukan sumber kesalahan logika atau validasi input.

Pendekatan ini banyak digunakan pada penelitian sistem serupa seperti sistem informasi perpustakaan berbasis web [1], sistem inventori alat medis [6], dan aplikasi e-commerce berbasis web [2]. Dengan metode ini, pengujian sistem Citra Medica dilakukan untuk menilai setiap fungsi antarmuka apakah telah berjalan sesuai kebutuhan masing-masing pengguna.

2.2. Teknik State Transition

State Transition Testing merupakan teknik pengujian yang digunakan untuk menguji perilaku sistem berdasarkan perubahan status (state) akibat aksi atau kejadian tertentu (event). Teknik ini sangat efektif untuk sistem yang memiliki urutan logis proses dan ketergantungan antarstatus pengguna. Dalam konteks sistem Citra Medica, setiap role pengguna memiliki status dan aktivitas berbeda yang dapat berubah sesuai aksi yang dilakukan. Pengujian dilakukan untuk memastikan setiap transisi antarstatus berlangsung sesuai logika sistem dan tidak terjadi kesalahan alur.

1. Admin

Admin bertanggung jawab atas pengelolaan data dasar sistem. Status utama dalam aktivitas admin adalah sebagai berikut:

Tabel 2. 1 Tabel State Transition Admin

Transisi	Dari	Aksi	Tujuan
T1	Login	Klik Login dengan Email dan Password Jika Benar	Sistem Akan Masuk Ke Menu Panel Admin.
T2	Login	Klik Login dengan Email dan Password Jika Salah	Sistem Akan Mengembalikan Ke Halaman Login
T3	Dashboard	Klik Dashboard	Sistem Akan Masuk Ke Menu Dashboard Admin
T4	Kelola Dokter	Klik Kelola Dokter	Sistem Akan Masuk Ke Menu Data Dokter
T5	Menambahkan Dokter	Input Data Dokter Dengan Ketentuan	Data Dokter Tersimpan Di Sistem
T6	Menambahkan Dokter	Mengosongkan salah satu field (misalnya nama dokter) lalu klik simpan	Validasi input muncul
T7	Mengubah Data Dokter	Mengedit salah satu field data dokter (misalnya nomor telepon) lalu klik simpan	Data dokter berhasil diperbarui
T8	Menghapus Data Dokter	Klik tombol hapus pada salah satu data dokter dan konfirmasi	Data dokter terhapus dari sistem
T9	Kelola Pasien	Klik Kelola Pasien	Sistem Akan Masuk Ke Menu Data Pasien
T10	Menambah Pasien	Input Data Pasien Dengan Ketentuan	Data Pasien Tersimpan Di Sistem
T11	Menambah Pasien	Mengosongkan salah satu field (misalnya nama pasien) lalu klik simpan	Validasi input muncul
T12	Mengubah Data Pasien	Mengedit salah satu field data pasien (misalnya nomor telepon) lalu klik simpan	Data pasien berhasil diperbarui

T13	Menghapus Data Pasien	Klik tombol hapus pada salah satu data pasien dan konfirmasi	Data pasien terhapus dari sistem
T14	Kelola Poli	Klik Kelola Poli	Sistem akan masuk ke menu data poli
T15	Menambahkan Poli	Input Data Poli	Input Data Poli
T16	Mengubah Data Poli	Dengan Ketentuan Mengedit Nama Poli	Dengan Ketentuan Data Poli Berhasil Di Perbarui
T17	Menghapus Poli	Klik tombol hapus pada salah satu data poli dan konfirmasi	Data poli terhapus dari sistem
T18	Kelola Obat	Klik Kelola Obat	Sistem akan masuk ke menu data kelola obat
T19	Menambah Data Obat	Data obat tersimpan di sistem	
T20	Menambah Data Obat	Mengosongkan salah satu field (misalnya harga obat) lalu klik simpan	Validasi input muncul
T21	Mengubah Data Obat	Mengedit salah satu field data obat (misalnya harga obat) lalu klik simpan	Data obat berhasil di perbarui
T22	Menghapus Obat	Klik tombol hapus pada salah satu data obat dan konfirmasi	Data obat terhapus dari sistem
T23	Logout	Klik Logout	Keluar dari sistem

2. Dokter

Dokter menggunakan sistem untuk mengatur jadwal dan memeriksa pasien. Status utama yang diuji meliputi:

Tabel 2. 2 Tabel State Transition Dokter

Transisi	Dari	Aksi	Tujuan
T1	Login	Klik Login dengan Email dan Password Jika Benar	Sistem Akan Masuk Ke Menu Panel Dokter
T2	Login	Klik Login dengan Email dan Password Jika Salah	Sistem Akan Mengembalikan Ke Halaman Login
T3	Dashboard	Klik Dashboard	Sistem Akan Masuk Ke Menu Dashboard Dokter
T4	Jadwal Periksa	Klik jadwal periksa	Sistem Akan Masuk Ke Menu Data Jadwal Periksa

T5	Menambahkan Jadwal Periksa	Input Data Jadwal Periksa Dengan Ketentuan	Data Jadwal Periksa Tersimpan Di Sistem
T6	Menambahkan Jadwal Periksa	Mengosongkan salah satu field (misalnya hari) lalu klik simpan	Validasi input muncul
T7	Mengubah Data Jadwal Periksa/Menonaktifkan Jadwal Periksa	Mengedit salah satu field data dokter (misalnya nomor jadwal) dan menonaktifkan jadwal periksa lalu klik simpan	Data jadwal periksa berhasil diperbarui
T8	Mengaktifkan Jadwal Periksa	Klik tombol aktifkan pada salah satu data dokter dan konfirmasi	Data periksa berhasil diaktifkan
T9	Periksa	Klik Periksa	Sistem Akan Masuk Ke Menu Periksa
T10	Periksa Pasien	Klik Periksa Pasien	Menampilkan Form Pemeriksaan Pasien
T11	Form Pemeriksaan	Mengisi Field Catatan Dokter dan Obat	Selesai Diperiksa
T12	Riwayat Pasien	Klik Riwayat Pasien	Sistem akan masuk ke menu data riwayat pasien
T13	Detail Riwayat Periksa	Klik tombol Riwayat Periksa	Sistem akan menampilkan data riwayat pasien
T14	Logout	Klik Logout	Keluar dari sistem

3. Pasien

Pasien merupakan pengguna akhir yang melakukan pendaftaran untuk periksa. Status utamanya meliputi:

Tabel 2. 3 Tabel State Transition Pasien

Transisi	Dari	Aksi	Tujuan
T1	Login	Klik Register	Sistem masuk ke menu registrasi
T2	Register	Klik Register dengan mengisi field data Jika Benar	Sistem masuk ke halaman login
T3	Register	Klik Register dengan mengisi field data Jika Benar	Validasi input muncul
T4	Login	Klik Login dengan Email dan Password Jika Benar	Sistem Akan Masuk Ke Menu Panel Pasien

T5	Login	Klik Login dengan Email dan Password Jika Salah	Sistem Akan Mengembalikan Ke Halaman Login
T6	Dashboard	Klik Dashboard	Sistem Akan Masuk Ke Panel Pasien
T7	Periksa	Klik Periksa	Sistem Akan Masuk Ke Menu Data Periksa
T8	Daftar Periksa	Input Data Dengan Ketentuan	Data Perikas Tersimpan Di Sistem
T9	Lihat	Melihat Detail serta nomer antrean	Muncul data detail pemeriksaan(status)
T10	Logout	Klik Logot	Keluar dari sistem

Pengujian dilakukan dengan menjalankan setiap skenario transisi di atas dan mencatat hasil aktual untuk dibandingkan dengan hasil yang diharapkan. Apabila seluruh transisi berjalan sesuai logika sistem, maka fungsi tersebut dinyatakan valid. Metode state transition terbukti efektif untuk menguji sistem yang memiliki alur status berlapis seperti ini. Beberapa penelitian sebelumnya menunjukkan keberhasilan teknik ini pada aplikasi konseling kesehatan digital [5], dan sistem pembelajaran interaktif berbasis web [10]. Dengan penerapan teknik ini, pengujian sistem appointment klinik “Citra Medica” dapat memastikan bahwa setiap transisi status antarrole (Admin, Dokter, dan Pasien) berjalan konsisten, tanpa kesalahan, dan memenuhi kebutuhan fungsional yang telah ditetapkan.

3. Hasil dan Pembahasan

Bagian ini menjelaskan hasil dari proses pengujian perangkat lunak yang telah dilakukan terhadap Sistem Appointment Klinik Berbasis Web “Citra Medica” menggunakan metode black box testing dengan teknik state transition. Pengujian dilakukan untuk memastikan setiap fungsi berjalan sesuai spesifikasi, serta transisi antarstatus dari masing-masing role pengguna berlangsung konsisten dan bebas kesalahan logika. Pengujian dilakukan pada tiga peran utama, yaitu Admin, Dokter, dan Pasien, sesuai urutan logis penggunaan sistem. Setiap hasil pengujian disajikan dalam bentuk tabel yang berisi kolom Transition, Skenario, Output, dan Kesimpulan.

3.1. Pengujian Role Admin

Role Admin bertanggung jawab mengelola seluruh data utama dalam sistem seperti dokter, pasien, poli, dan obat. Pengujian dilakukan terhadap setiap aktivitas CRUD (Create, Read, Update, Delete) untuk memastikan sistem merespons setiap aksi dengan benar.

Tabel 3. 1 Tabel Pengujian Role Admin

Transisi	Skenario	Output	Kesimpulan
Test1	Login dengan email dan password benar	Sistem berhasil masuk ke menu panel admin	Berhasil
Test2	Login dengan email dan password salah	Sistem menampilkan pesan error dan kembali ke halaman login	Validasi berhasil
Test3	Klik Dashboard di menu utama	Sistem menampilkan halaman dashboard admin	Berhasil
Test4	Klik Kelola Dokter	Sistem masuk ke	Berhasil

Test5	Input data dokter sesuai ketentuan	menu data dokter Data dokter tersimpan di sistem	Berhasil
Test6	Kosongkan salah satu field (misalnya nama dokter) lalu klik simpan	Muncul validasi input (field wajib diisi)	Validasi berhasil
Test7	Edit salah satu data dokter lalu klik simpan	Data dokter berhasil diperbarui	Berhasil
Test8	Klik tombol hapus pada data dokter dan konfirmasi	Data dokter terhapus dari sistem	Berhasil
Test9	Klik Kelola Pasien	Sistem masuk ke menu data pasien	Berhasil
Test10	Input data pasien sesuai ketentuan	Data pasien tersimpan di sistem	Berhasil
Test11	Kosongkan salah satu field (misalnya nama pasien) lalu klik simpan	Muncul validasi input (field wajib diisi)	Validasi berhasil
Test12	Edit salah satu data pasien lalu klik simpan	Data pasien berhasil diperbarui	Berhasil
Test13	Klik tombol hapus pada data pasien dan konfirmasi	Data pasien terhapus dari sistem	Berhasil
Test14	Klik Kelola Poli	Sistem masuk ke menu data poli	Berhasil
Test15	Input data poli sesuai ketentuan	Data poli tersimpan di sistem	Berhasil
Test16	Edit nama poli lalu klik simpan	Data poli berhasil diperbarui	Berhasil
Test17	Klik tombol hapus pada data poli dan konfirmasi	Data poli terhapus dari sistem	Berhasil
Test18	Klik Kelola Obat	Sistem masuk ke menu data obat	Berhasil
Test19	Input data obat sesuai ketentuan	Data obat tersimpan di sistem	Berhasil
Test20	Kosongkan salah satu field (misalnya harga obat) lalu klik simpan	Muncul validasi input (field wajib diisi)	Validasi Berhasil
Test21	Edit salah satu field data obat lalu klik simpan	Data obat berhasil diperbarui	Berhasil
Test22	Klik tombol hapus pada salah satu data obat dan konfirmasi	Data obat terhapus dari sistem	Berhasil
Test23	Klik Logout	Sistem keluar dan kembali ke halaman login	Berhasil

3.2. Pengujian Role Dokter

Role Dokter digunakan untuk mengelola jadwal praktik dan melakukan pemeriksaan pasien. Pengujian dilakukan terhadap fungsi login, manajemen jadwal, pemeriksaan pasien, riwayat pasien, serta logout.

Tabel 3. 2 Tabel Pengujian Role Dokter

Transisi	Skenario	Output	Kesimpulan
Test1	Login dengan email dan password benar	Sistem berhasil masuk ke menu panel dokter	Berhasil
Test2	Login dengan email dan password salah	Sistem menampilkan pesan error dan kembali ke halaman login	Validasi Berhasil
Test3	Klik Dashboard di menu utama	Sistem menampilkan halaman dashboard dokter	Berhasil
Test4	Klik Jadwal Periksa	Sistem masuk ke menu data jadwal periksa	Berhasil
Test5	Input data jadwal periksa sesuai ketentuan	Data jadwal periksa tersimpan di sistem	Berhasil
Test6	Kosongkan salah satu field (misalnya hari) lalu klik simpan	Muncul validasi input (field wajib diisi)	Validasi Berhasil
Test7	Edit salah satu field data jadwal (misalnya nomor jadwal) dan menonaktifkan jadwal periksa, lalu klik simpan	Data jadwal periksa berhasil diperbarui dan status jadwal menjadi nonaktif	Berhasil
Test8	Klik tombol aktifkan pada salah satu jadwal dan konfirmasi	Jadwal periksa berhasil diaktifkan	Berhasil
Test9	Klik menu Periksa	Sistem masuk ke menu periksa pasien	Berhasil
Test10	Klik Periksa Pasien pada daftar pasien	Sistem menampilkan form pemeriksaan pasien	Berhasil
Test11	Isi field Catatan Dokter dan Obat, lalu klik simpan	Status pasien menjadi "Selesai Diperiksa"	Berhasil
Test12	Klik Riwayat Pasien	Sistem masuk ke menu data riwayat pasien	Berhasil
Test13	Klik tombol Riwayat Periksa	Sistem menampilkan detail riwayat pemeriksaan pasien	Berhasil
Test14	Klik Logout	Sistem keluar dan kembali ke halaman login	Berhasil

Hasil pengujian pada role Dokter menunjukkan bahwa seluruh fungsi berjalan sesuai kebutuhan. Proses pembuatan dan pengelolaan jadwal berjalan baik, serta sistem dapat memperbarui status pasien dari Belum Diperiksa menjadi Sudah Diperiksa secara otomatis. Hal ini menunjukkan keberhasilan penerapan teknik state transition testing dalam memvalidasi perubahan status pemeriksaan pasien.

3.3. Pengujian Role Pasien

Role Pasien merupakan pengguna akhir yang berinteraksi langsung dengan sistem untuk melakukan pendaftaran periksa dan melihat status pemeriksaannya.

Tabel 3. 3 Tabel Pengujian Role Pasien

Transisi	Skenario	Output	Kesimpulan
Test1	Klik Register pada halaman login	Sistem masuk ke menu registrasi	Berhasil
Test2	Isi semua field data registrasi dengan benar lalu klik Register	Sistem menyimpan data dan masuk ke halaman login	Berhasil
Test3	Kosongkan salah satu field (misalnya nama atau email) lalu klik Register	Muncul validasi input (field wajib diisi)	Validasi Berhasil
Test4	Login dengan email dan password benar	Sistem berhasil masuk ke menu panel pasien	Berhasil
Test5	Login dengan email dan password salah	Sistem menampilkan pesan error dan kembali ke halaman login	Validasi Berhasil
Test6	Klik Dashboard di menu utama	Sistem menampilkan halaman panel pasien	Berhasil
Test7	Klik menu Periksa	Sistem masuk ke menu data periksa	Berhasil
Test8	Input data pemeriksaan sesuai ketentuan	Data periksa tersimpan di sistem	Berhasil
Test9	Klik Lihat untuk melihat detail dan nomor antrean	Sistem menampilkan data detail pemeriksaan serta status	Berhasil
Test10	Klik Logout	Sistem keluar dan kembali ke halaman login	Berhasil

Hasil pengujian role Pasien menunjukkan bahwa sistem berfungsi sesuai kebutuhan pengguna akhir. Proses pendaftaran, login, dan daftar periksa bekerja dengan benar, serta status pasien berubah secara otomatis setelah pemeriksaan oleh dokter. Validasi input juga bekerja sesuai yang diharapkan, membantu menjaga integritas data pasien.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan menggunakan metode black box testing dengan teknik state transition pada Sistem Appointment Klinik Berbasis Web “Citra Medica”, dapat disimpulkan bahwa proses pengujian berjalan dengan baik dan menghasilkan sistem yang sesuai dengan kebutuhan fungsional pengguna. Pengujian black box terbukti efektif dalam menilai kinerja sistem dari sisi pengguna tanpa memerlukan pengetahuan tentang kode program internal. Setiap modul diuji berdasarkan kesesuaian antara masukan dan keluaran yang dihasilkan, sehingga seluruh fungsi sistem dapat diverifikasi dengan jelas. Teknik state transition yang digunakan dalam penelitian ini berhasil memastikan konsistensi dan keakuratan transisi status antar role pengguna, yaitu admin, dokter, dan pasien. Setiap perubahan status seperti login, pengelolaan data, pembuatan jadwal, pemeriksaan pasien, dan pendaftaran periksa telah berfungsi sesuai dengan logika sistem yang dirancang. Hasil pengujian menunjukkan bahwa seluruh fungsi utama sistem berjalan dengan baik, meliputi pengelolaan data dokter, pasien, poli, dan obat pada role admin; pengaturan jadwal dan pemeriksaan pasien pada role dokter; serta pendaftaran dan pelacakan status pemeriksaan pada role pasien. Semua skenario pengujian menghasilkan keluaran yang sesuai dengan ekspektasi, tanpa ditemukan kesalahan logika yang signifikan, dengan tingkat keberhasilan pengujian mencapai 100%. Hal ini menunjukkan bahwa mekanisme validasi, alur transisi status, serta pengendalian kesalahan pada sistem telah berfungsi dengan optimal. Dengan demikian, sistem appointment klinik “Citra Medica” dinyatakan siap untuk digunakan dalam mendukung proses layanan klinik secara digital, efisien, dan terintegrasi.

4.2. Saran

Meskipun hasil pengujian menunjukkan bahwa sistem telah berjalan optimal, terdapat beberapa hal yang dapat dijadikan masukan untuk pengembangan di masa mendatang. Disarankan agar sistem ini dikembangkan lebih lanjut dengan menambahkan fitur notifikasi otomatis seperti email atau pesan singkat untuk mengingatkan jadwal periksa pasien dan dokter secara real-time. Selain itu, sistem dapat diintegrasikan dengan Electronic Medical Record (EMR) agar data pemeriksaan pasien dapat tersimpan secara terpusat dan mudah diakses oleh pihak klinik. Penelitian berikutnya juga diharapkan dapat melakukan pengujian tambahan menggunakan metode white box testing atau security testing guna memeriksa keandalan sistem dari sisi kode dan keamanan data pengguna. Capote (2024) juga menegaskan bahwa pemilihan antara metode black box dan white box harus mempertimbangkan konteks proyek serta risiko pengujian untuk mencapai hasil yang optimal[15]. Antarmuka sistem juga perlu disempurnakan agar lebih responsif dan ramah pengguna pada berbagai perangkat, terutama versi mobile. Terakhir, pengujian performa dengan jumlah pengguna dan data yang lebih besar perlu dilakukan untuk mengukur kemampuan sistem dalam menangani beban akses tinggi secara simultan. Dengan berbagai saran pengembangan tersebut, diharapkan sistem appointment klinik berbasis web “Citra Medica” dapat terus ditingkatkan sehingga menjadi solusi digital yang andal, aman, dan efisien dalam meningkatkan mutu pelayanan klinik di Indonesia.

References

- [1] M. Mintarsih, “Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation,” *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, hlm. 33–35, Feb 2023, doi: 10.47233/jteksis.v5i1.727.
- [2] A. Salim dan R. Rusdiansyah, “Implementasi Black Box Testing pada website E-commerce Shopee menggunakan State Transition Testing,” *Jurnal Informatika dan Bisnis*, vol. 13, no. 2, hlm. 161–170, Jan 2025, doi: 10.46806/jib.v13i2.1321.

- [3] A. Corazza, S. Di Martino, A. Peron, dan L. L. L. Starace, "Web application testing: Using tree kernels to detect near-duplicate states in automated model inference," dalam *International Symposium on Empirical Software Engineering and Measurement*, IEEE Computer Society, Okt 2021. doi: 10.1145/3475716.3484187.
- [4] Y. M. Mohialden, N. M. Hussien, dan S. A. Hameed, "Review of Software Testing Methods," *Journal La Multiapp*, vol. 3, no. 3, hlm. 104–112, Jun 2022, doi: 10.37899/journallamultiapp.v3i3.648.
- [5] D. Maryanti, A. R. Pangesti, dan T. Suprihatiningsih, "Black Box Testing for HIV AIDS Digital Counseling Website (D-Cohiva Apps) with State Transition Technique," *Jurnal Penelitian Pendidikan IPA*, vol. 9, no. SpecialIssue, hlm. 822–827, Des 2023, doi: 10.29303/jppipa.v9ispecialissue.6087.
- [6] "218-Article Text-4382-1-10-20240308".
- [7] M. Rizky, A. Agustin, T. Cahya Herdiyani, A. Waluyo, dan J. Riyanto, "Pengujian Black Box menggunakan Teknik State Transition Testing pada Aplikasi BMI Calculator Berbasis Android," 2021. [Daring]. Tersedia pada: <http://pijarpemikiran.com/index.php/Scientia>
- [8] M. Ansfridus, "SEMINAR NASIONAL AMIKOM SURAKARTA (SEMNAS) 2023 Penerapan Black Box Testing Pada Fitur TEA Dalam Aplikasi QM Mobile Astra Credit Companies".
- [9] M. Raafi, R. 1✉, H. Yuana, dan M. F. Rahmat, "Perancangan Sistem Informasi Inventaris Penjualan Berbasis Web Di Toko Konter Handphone Reog Cell," *INNOVATIVE: Journal Of Social Science Research*, vol. 4, hlm. 16101–16110.
- [10] M. Kholil, S. Nur Agung Wibowo, P. Audio dan Video, dan A. Komunitas Negeri Putra Sang Fajar Blitar, "Pengembangan Media Pembelajaran Interaktif Menggunakan Adobe Animate Berbasis Video Development Of Interactive Learning Media Using Adobe Animate Based On Video," *Journal of Electrical, Electronic, Mechanical, Informatic and Social Applied Science Jurnal EEMISAS*, vol. 2, no. 1, hlm. 32–37, 2023.
- [11] Y. Salih dan R. Saefullah, "Black Box Testing on Website-Based Guestbook Registration Applications," *International Journal of Mathematics, Statistics, and Computing*, vol. 2, no. 2, hlm. 44–49, 2024.
- [12] S. G. Tetteh dan S. Gbli Tetteh, "Software Testing Techniques and Levels in Software Development," *Article in International Journal of Advancements in Computing Technology*, vol. 2, hlm. 10–19, 2024, doi: 10.56472/25838628/IJACT-V2I1P102.
- [13] J. Vitorino, T. Dias, T. Fonseca, E. Maia, dan I. Praça, "Constrained adversarial learning for automated software testing: a literature review," 1 Juni 2025, *Springer Nature*. doi: 10.1007/s42452-025-07073-3.
- [14] A. Roman, "Black-Box Testing Techniques," dalam *A Study Guide to the ISTQB® Foundation Level 2018 Syllabus*, Springer International Publishing, 2018, hlm. 25–60. doi: 10.1007/978-3-319-98740-8_4.
- [15] T. D. Capote, "A COMPARATIVE STUDY OF BLACK BOX AND WHITE BOX TESTING TECHNIQUES IN MODERN SOFTWARE DEVELOPMENT," vol. 5, no. 1, hlm. 1–7, [Daring]. Tersedia pada: https://iaeme.com/Home/journal/FET1ID:FET_05_01_001Availableonlineathttps://iaeme.com/Home/issue/FET?Volume=5&Issue=1